A CRITICAL ANALYSIS OF SCHOOL COMPUTER STUDIES SYLLABUSES IN SOUTH AFRICA WITH REFERENCE TO UNIVERSITY COMPUTER SCIENCE CURRICULA.

THESIS

Submitted in partial fulfilment of

the requirements for the Degree of

MASTER OF EDUCATION

of Rhodes University

by

R. ROETS

December 1991

Abstract

This thesis analyses the existing and proposed Computer Studies syllabuses in South African schools. Thereafter it compares the South African syllabuses with syllabuses in the United Kingdom. An attempt is made to assess the objectives of the design of the new S.A. syllabus. University Computer Science syllabuses are examined in order to gauge the purported overlap between what is taught at schools and universities. Opinions are obtained on the problems which apparently exist in offering or teaching the subject at schools and universities by conducting surveys on syllabus designers and university Computer Science departments. Finally recommendations are made for future Computer Studies syllabuses.

A critical analysis of school Computer Studies syllabuses in South Africa with reference to university computer science curricula.

Table of Contents

Acknowledgements

I wish to express my gratitude to the following people for making the production of this work possible:

- My supervisor, Mrs Cecille Marsh, for suggestions and constructive criticism.
- Professor Terry Marsh for his help in formulating this research.
- My husband, Peet Roets, for translating questionnaires into Afrikaans.
- Respondents of the questionnaires for their patience.

Thank you to the above. Any errors remaining are entirely my own doing.

Glossary of terms

To clarify nomenclature:

In South Africa 'Computer Studies' implies the study of the computer as a matriculation subject, whereas 'Computer Literacy' implies a broader, less technical approach to Information Technology (IT) with emphasis on application software. In the U.S.A., the term 'Computer Studies' implies both the above, whereas in the U.K. the term 'Computer Science' and 'Computer Studies' are used: Information Technology at GCSE is examined through 'Computer Studies' a.k.a. 'Computer Science', and 'Information Technology'. In Europe the term 'Informatics' is used for the study of the computer. The South African terms will be used unless otherwise stated.

List of tables

List of figures

List of Appendices

Chapter 1.

Introduction

The issue of computers in schools has been much discussed, criticised and researched. There have been many areas of concern and interest. The main debate has centred around the issue of whether there should be computers in schools at all. If the answer to this question was 'yes', then in what guise? The effect of the introduction of computers in schools on teaching, learning and thinking has been investigated, as well as the problems associated with their introduction.

Questions have been raised such as whether the subject Computer Studies should exist, what it should comprise and what emphasis should be placed on the role of programming. These concerns are mirrored in journal articles with titles such as "The rise and fall of computer studies (A subject with a death wish)" (Selwood 1988); "Computer Science education is very much your business" (Mead 1988); "Today's development for tomorrow's world - educating for the information society" (Longworth 1982); "Computers in education: Rear-vision thinking or golden opportunities for the future" (Kennett 1990); and "Does computer studies have a future?" (Ruthven 1981), amongst others. The contents of the courses at school and at university have been changing, as has the relationship between the courses. What schools and universities teach and what the workplace requires is another factor to be considered. Titles such as "Does today's industry need qualified computer scientists?" (Steiner 1991) mirror the present concerns.

In South Africa, Computer Studies in the Department of Education and Culture (DEC):House of Assembly schools is a relatively new subject as it was introduced in schools in the early seventies. Computer Studies on the standard grade has been offered in the Cape since 1977. Higher grade Computer Studies has been available since 1986, and there have been two syllabuses; the last one came into force in 1985. The DEC has drafted a new syllabus for both standard and higher grade to come into effect in 1993. In 1988 1056 pupils in all provinces took the subject for matriculation as opposed to 573 in 1985. (Wilkens 1990 pers.comm.). The Cape Education Department (CED) is considering introducing Computer Literacy in an evolutionary manner. The Orange Free State is offering Computer Studies as a sixth subject in three selected schools from 1990. In the Transvaal it has been offered as a sixth subject in selected schools from 1990 onwards. In Natal it can only be taken as a seventh subject. However, Natal schools offer a form of Computer Literacy for standard 6 and standard 7.

The Department of Education of the House of Delegates offers Computer Studies in standards 8 to 10 as a matric option (at 54 high schools at present) with the first matriculation having taken place in 1988. The use of computers as a training aid in schools (Computer assisted Instruction or CAI) is viewed as a priority by the Department of Education and Training. In schools under the control of the Department of Education of the House of Representatives, computers are used for CAI. (Fallick & Pistorius 1990:56).

Universities have been offering Computer Science since the early seventies, initially as part of other courses, for example Applied Mathematics. Later, courses such as Business Information Systems (BIS) and Software Engineering (SE) were introduced. Universities have had some influence in framing the school syllabuses. At least 4 university lecturers served on the committee for syllabus design on the 1985 syllabus. (Riordan 1991 pers.comm.). The courses at both schools and universities have been changing for this ever evolving topic.

The purpose of this study is to analyze what schools and universities are offering in South Africa in terms of computer science education, particularly with reference to the purported overlap between what is taught at schools and universities. The study will examine the following issues in detail:

1. What is being taught at schools and universities in the field of computer science and computer related subjects.

2. What problems and issues are raised by the introduction of Computer Studies into schools and what relationship exists with related subjects such as Computer Literacy.

3. What the objectives of the designers of the school syllabuses are.

4. What influence is exerted by universities on the design of the proposed new school syllabus.

5. What future Computer Studies syllabuses should contain for schools in South Africa.

The study examines existing literature in order to highlight issues which have arisen in the past concerning the topic. An analysis is carried out on the contents of present and proposed South African syllabuses. Because of South Africa's historic educational association with Britain, it was thought instructive to examine what is taught at schools in the United Kingdom (U.K.), and to draw comparisons between the syllabuses of the two countries. The study analyses a survey to elicit the motivation and opinions of the designers of the proposed new syllabus. South African University syllabuses are analyzed to examine common areas. Universities were also canvassed for their opinions on the situation at schools as regards Computer Studies The study examines these findings and then attempts to suggest what future Computer Studies syllabuses in South Africa should comprise.

Chapter 2

Literature survey.

Outline of the chapter

2.1 Why Computers in Schools?

There are three educational thrusts for computing education in schools: First as a subject in its own right, i.e. Computer Studies for academically able pupils. Secondly in the form of Computer Literacy, i.e. the more general application-based approach for all pupils. Thirdly there is the cross-curricular approach in which computers are used as tools in the teaching of other subjects (for Computer Aided Instruction or CAI and Computer Aided Learning or CAL).

The main area of agreement seems to be that the cross-curricular use of the computers will be of the greatest benefit for pupils. A variety of writers hold this opinion. Tucker (1985:22) states "the key to the future use of the computer in education . . . lies in its use as a tool to get work done ". North states (1990:13) "through its integration with traditional subjects, it has the potential to enhance the quality of learning ". Most writers realise that there may be difficulties in putting this into practice e.g. Downes (Downes 1990:430). In Britain the new national curriculum introduces Information Technology (IT) with attainment targets in many other subjects (Anderson 1990). Despite this agreement, the subject Computer Studies remains in schools. Hodson (1990:70) noted that in New Zealand the study of the computer in its own right at school level was persisting despite promising developments in using the computer in CAL.

Computers and Computers Studies have been introduced into schools for a variety of reasons. According to Hawkridge (Hawkridge 1990:1-6), there were underlying rationales for introducing computers and computer education into schools.

Briefly, these were the following:

1. The <u>social rationale:</u> all children should be aware and unafraid of computers, i.e. parents do not want their children to lag behind others. ("The greatest benefit to the greatest

number", or "free and equal access for all children to computers"). (Hawkridge 1990:2).

2. The <u>vocational rationale:</u> children should be able to operate computers i.e schools should provide some vocational training.

3. This ties in very closely with the <u>IT rationale</u> that the country needs people with IT skills and schools should consider this in their syllabuses.

4. The <u>pedagogic rationale:</u> teaching may be improved through teaching by computers using CAL or CAI.

5. It also may be cost-effective to replace (or aid) teachers with CAI, i.e. the <u>cost effective rationale</u>.

6. The <u>catalytic rationale:</u> computers will change the way in which schools operate, introducing more flexibility.

7. Finally the <u>special needs rationale:</u> computers can act uniquely as an aid for children with disabilities, or "special needs".

Hawkridge stated that it was very necessary for any country's Ministry of Education to realise which rationale it was backing. The implications in terms of costs and resources for each rationale should be considered.

It is proposed that the social, vocational and IT rationale, as well as the catalytic rationale will be the rationales found to underlie the offering of Computer Studies in DEC schools in S.A. This proposal is examined further in chapter 4. Hawkridge stated that most of these rationales, despite their surface acceptability, did not bear too close a scrutiny. This chapter will consider his and various other writers' arguments as applied to the rationales for Computer Studies in schools in South Africa.

2.1.1 A broad education for all

One generally accepted basic aim of education is to prepare pupils for life. Eisner (1981:9) stated that in America "there was a time when the concept of a balanced curriculum was much discussed . . . children needed opportunities to work in a variety of fields of study ". We want our children to have computer knowledge since this will equip them better for the technological future they will face. What Stonier would call "education for life " when he wrote "it is not possible to understand the world if we do not understand the impact science and technology have on all aspects of society." (Stonier 1979:34). Evans wrote "Basic computer 'skills' are the fundamental requirements for the society of the future" (Evans 1986:81). This <u>social rationale</u> results in parental pressure on schools to introduce some form of computer education. (Tucker 1985). If the social rationale is to be taken to its liberal conclusion, we should then be providing computer education for all pupils. In South Africa this is of course not happening, the more so in that even in DEC schools there is often a selection of pupils to take Computer Studies. Wellington raised a problem with Computer Studies in schools in connection with the social rationale: "an excessive concentration on computing courses for a minority severely restricted the aim of IT awareness and familiarity for all, and the use of CAL in other subjects." (Wellington 1990).

2.1.2 Equipping pupils for the workplace

In South Africa we are being assailed constantly by the successful examples of Israel and the so-called 'Pacific Rim' countries that have achieved national success through government funding of vocational training at schools and government funding of purely vocational schools. This is discussed both by Hawkridge (1990:5) and Rautenbach (1979 and 1980).

Lorton and Muscat suggested that school leavers should be computer literate to enable them to slot in at what they call the lowest level of the data processing skills base, thus adding to their career choices. This level includes card-punch operators, computer operator trainees, word processing operators, librarians, programming trainees, and input/output clerks. (Lorton and Muscat 1981:23). Evans stated that "children leaving school in five years' time will need skills which are being only touched upon lightly in our present curriculum" (Evans 1986:63).

Hawkridge viewed this argument, which he called the <u>vocational rationale</u>, with suspicion. He questioned whether schools were the correct place to teach vocational training. This opinion is of course widely held. Similarly, he felt the <u>IT rationale</u> suffered from the same drawback as the vocational rationale. The latter rationale also opens the possibility of IT firms introducing computers in schools, seeing these as easy markets for their goods, irrespective of the needs of the schools. We have seen an extension of this problem in South Africa: Braam Fourie , the Director-general of Department of Education and Training (DET) awarded an IT educational contract of R4,8 million to the company Learning Technologies in which his son had an interest. ("Inquiry into DET deal," 1988). Both Lees and Longworth support Hawkridge's opinion on these rationales. Longworth (1981:18) stated that the aim in British education had not been to provide "industry fodder", but to educate the "whole man". Lees (1981:38) wrote that "general education in schools should not aim to meet the specialised requirements of particular occupations".

2.1.3 Better learning

The <u>catalytic rationale</u> suggests that computers will encourage different or better ways of learning:

2.1.3.1 Computers may encourage co-operative learning

One of the advantages put forward for teaching computing, whether through the Computer Literacy courses or through Computer Studies, is that it allows for co-operative learning. This can facilitate learning for some pupils and has some psychological benefits such as improved confidence and better social interaction for the pupils. Evans (Evans 1986:65) stated that the employment of the computer was "often declared . . . the single best generator of group-coordination ever devised" leading to "social interaction and corporate effort". By contrast, Wood believed that gains by co-operative learning is overrated except in laboratory settings. (Wood 1988:214).

2.1.3.2 Computers can affect problem-solving skills

It has been stated that the programming of computers (generally in the traditional Computer Studies course, but also in general computer literacy) has a role to play in the improvement of thought processes. This has created much controversy and research. (Chen 1985:48-50). Lieberman quoted research by Paisley & Chen, which affirmed that the "process of writing computer programs, with its requirements for procedural, analytic, abstract thinking, improves children's problem-solving abilities". She also held that writing programs could accelerate cognitive development. (Lieberman 1985:73-74).

Pea stated "Systematic developmental research documenting what children are learning as they learn to program is necessary, rather than existing anecdotes . . . After a year's experience of programming, children . . . did not display greater planning skills than a matched group who did not do programming ". (Pea 1987:155-156). A research project suggested that the transfer of problem solving skills to areas other than programming was not guaranteed. (Pea, Kurland & Hawkins 1985).

Cathcart found evidence of greater divergent thinking as well as other cognitive gains in students after a LOGO course. (Cathcart 1990:235). Griffin suggests that computers may be useful in helping pupils move from concrete operational to formal operational thinking (as proposed by Piaget). (Griffin 1988:444).

This issue is by no means resolved.

2.1.3.3 Mathematics and Computer Studies

The subject of Computer Studies has traditionally been associated with Mathematics. There is a reasonable base of evidence to support the premise that writing programs to solve a specific mathematical problem contributes to the learning of mathematical concepts. (Johnson and Anderson 1985:691). Learning to program aids in problem solving and therefore is a useful approach in Mathematics to problem solving according to Forcheri, Furinghetti & Molfino (1990:332).

Should the subject, instead of being a subject in its own right, be part of the Mathematics syllabus? Although the subject originally was closely linked to Mathematics, Cockcroft (1982:116) stressed that the working group for the Microelectronic Education Programme (MEP) was "unanimous that computer studies should not be regarded as part of Mathematics, but should ideally exist within a separate department." This was to prevent Computer Studies from appearing too mathematical, and the social and application aspects from being ignored. Lees disliked computing linked to Mathematics, as this led to too technical an approach. The "use of the system rather than internal workings of the computer" should be stressed, he stated.

2.2 Problems with Computer Studies in schools

The subject is relatively new, based on technology which is, to use the common cliché, changing all the time. There have been problems associated with offering Computer Studies in schools.

2.2.1 Elitism

The problem of elitism is allied particularly to the restricted nature of the subject of Computer Studies in many schools. Ruthven (1981:5) saw a problem in that Computer Studies reached a few pupils only. In South Africa access to computers in state schools is limited mainly to Department of Education and Culture (DEC) schools. In DEC schools in most provinces only selected pupils may take Computer Studies. Parents of school-going children are concerned that the subject as taught at school is elitist in that schools select children to take Computer Studies based on their Mathematics and English results (Wheals 1990 pers. comm.).

As previously stated, Hawkridge felt that this situation violated the social rationale.

2.2.2 Teacher training

Teachers qualified to teach computing or implement IT have been notoriously lacking. This is partly owing to the salaries that people qualified in Computer Science can obtain in industry compared to education. It is also due to the changing aspect of the technology itself, which makes keeping teachers up-to-date a major retraining problem. Maddison quoted an article by K. Hennesey, lecturer at the University of Manchester Institute of

Science and Technology about A-level Computer Science: "We find that . . . their teachers are by and large, amateurs . . . It is extremely rare to find competent, qualified, experienced computer professionals in the classroom. Normally he is out getting two or three times as much money in industry." (Maddison 1983:76). Terry expressed concern over the issue that Computer Studies could not be taught adequately at school level owing to the limited number of teachers who are trained to teach Computer Studies (Terry 1990 pers.comm.). However, MacLachlan's view is that there are plenty of qualified Computer Studies teachers, but that they have no intention of teaching Computer Studies. (MacLachlan 1991 pers.comm.). It is a subject which is daunting and demanding to teach, because of the technical component and the one-to-one nature of the practical aspects of the subject.

2.2.3 Fitting the subject into the curriculum

If Computer Studies is introduced in the normal syllabus at schools, it needs to fit in with many other possible subject choices. This can cause problems with timetables that are already overfull. (Ruthven 1981:5).

2.2.4 Special problems in South Africa

The costs of hardware and software are not negligible, and there are new technological developments all the time. The DEC (House of Assembly) has stated that in principle every school would be allocated five computers. (Heunis 1990 pers.comm.). This is not happening in practice in DEC schools, and far less so in any other schools. Natal is somewhat unique in the equipment which the schools have: each Natal provincial secondary school (DEC) has a minimum of 6 Apples funded by the province. These are being replaced with IBM-compatible Personal Computers (PC's). At present central funds available for education are being channelled into providing more basic needs such as schools and teachers for the mass of disadvantaged children. Schools that wish to acquire hardware will probably have to provide the funds themselves in the future. (However, there has been a new move to supply DEC schools with computers. Whether this includes DET schools is not known - see Cape Education Department: Computers in Schools and Colleges Project 1991).

2.3 Historical perspective on computing in British schools

2.3.1 Student numbers

Courses in Computer Studies have been offered in Britain since 1968 (Hennessy & Croft 1980:33). In Northern Ireland Computer Studies is offered in 97% of the schools and is the 12th most popular subject (following history) (Anderson 1991:24)

The following Table (2.1) shows the trends in student numbers in the U.K. These figures are obtained from McGee (1980:69), Rich (1986:16), Hennesey (1983:76), and Selwood (1988:2).

Table 2.1 - Number of scholars in schools in the U.K.

1974:	6887	O level and CSE Computer Studies
1975:	6000	1000 O/GCE 5000 CSE levels
1977:	23073	O, CSE & A level
1978:	26419	O, A levels
1981:	52000	CSE or GCE level
1983:	102182	O, CSE & A levels
1985:	130500	62500 GCE 58000 CSE levels

2.3.2 Computer availability:

By 1982 there were 4000 schools who had ordered computer equipment. (Hawkridge 1987:22). In primary schools the ratio of computers to pupils was 1:107 in 1985, and 1:69 in 1989. In secondary schools these ratios were 1:60 in 1985 and 1:28 in 1989. (Wellington 1990:62).

2.3.3 Curricula in schools

Initially there was no national curriculum - curricula were devised by regional examination boards and principals and staff. The head teacher was responsible for the curriculum and answered to a board of governors and the Local Education Authority. University entrance requirements largely determined the syllabus. These university boards could suggest new curricular offerings but the onus was on the schools to agree to them or not. (Gwyn 1982:41). The position was made more difficult because the examining bodies offered examinations at O level (GCE), CSE, and A level. (Now there are the GCSE and A levels). Informatics or Computer Studies was introduced in schools as early as 1960's, usually by an enthusiastic teacher. This allowed for many experiments in curricula. However, problems arose because there was no national co-ordination and, for example, hardware standardisation suffered. Since funding was largely local, there was disparity between the technology available in schools. Lack of standardisation caused teacher training to be out of step with what was happening at schools. (Gwyn 1982:41).

2.3.4 Examining bodies

There has been an attempt to centralise control of the curriculum. A new system of examinations based on national criteria (GCSE) was announced in 1984 to take effect in 1988 replacing existing O and CSE level examinations. For GCE there are 7 Examination boards. For the GCSE (comprising GCE O level, CSE & joint O/GCE level examinations) there are 5 groups. (Education Authorities Directory 1990). See Appendix B, tables 1 and 2 for details of the boards and groups. The GCE and CSE boards have responsibility for maintaining standards at A, B, C levels and D to G levels respectively.

2.3.5 Funding

2.3.5.1 The Microelectronic Education Programme (MEP)

The Microelectronic Education Programme (MEP) was formed in 1980 to co-ordinate funding and development of software for schools. (Wellington 1990:57-58). Another body that was set up to co-ordinate efforts was the Chelsea Software Library. This body was funded by the MEP to revise and republish many programs for computer aided learning (CAL) for micros that had previously been developed. (Masterton 1982:22).

2.3.5.2 Funding and state support

See Table 2.2 below for details of government and parastatal support for computer hardware and software in schools. The sources of funding were the Department of Trade and Industry (DTI), the Department of Education and Science (DES), and the Microelectronics Support Unit (MeSU) and the Technical and Vocational Education Initiative (TVEI).

Table 2.2 - Funding in Schools
(Source: Wellington 1990:57-58)

1980:	DES	MEP	23,000,000
	DTI	Software	3,5000,000
1981:	DTI	Micros in Schools	16,000,000
1983:		TVEI	An unknown amount
1986:	DTI	Modems	1,000,000
1986:		MeSU	3,000,000
1988:	DES	Education support for IT	19,000,000

Appendix B Table 3 gives details of government and parastatal support for the 'computers in education' thrust.

2.4 The subject Computer Studies in Western countries

2.4.1 The subject in the curriculum

Longworth (1981:17) questioned whether there was a case for retaining a separate subject called 'Computer Studies' since computer education was too technical.

Ruthven (1981:5) stated that there were problems with the introduction of Computer Science or Computer Studies as an extra subject at schools. It had to compete with many other new subjects such as electronics, sociology, drama, economics etc. He disliked the superficial quality of the content of Computer Studies courses and said the course was too technical. He felt that the subject should concentrate on the interaction between man and machine; it should reexamine the idea of intelligence, the nature of work, and political and economic forces on technological advances.

Hawkridge stated that if the subject was seen as a necessary part of the school syllabus, then the extra time required by the subject should be increased and the equipment should be available. The conflict with other established and new subjects had to be considered. "As such the course would require more time than simply literacy, and therefore there is conflict with other subjects for time.". Ruthven echoed this problem by stating that Computer Studies would have to vie with other subjects in an already overcrowded syllabus.

2.4.2 Contents of the subject

Should the subject incline more to Computer Literacy as defined in the Glossary? Schiffman suggested that we should examine what society required from school leavers to decide what "skills, values, and attitudes students will need when they leave the educational system". In terms of this analysis, she postulated that what she termed "Computer Literacy" (U.S.A) should consist mainly of the use of application software with a de-emphasis on programming. (Schiffman 1985:678). In South Africa, the doubt has been raised as to whether professional educators are "capable of making unbiased, well-informed decisions on syllabuses, objectives, exams . . . if they have had limited real world experience ?" (Roets 1991 pers.comm.). This was echoed by Dodds. (Dodds 1991 pers.comm.).

Should Computer Studies comprise programming? If so, which computer language? Although Fritz and Schiffman held that programming should be de-emphasized, Fritz stated that programming still had a place in a Computer Studies course. She maintained that in schools pupils often learnt about topics that they would not use directly later. She said that the objective of teaching programming was to "understand a problem and describe it as a programmable algorithm ". She did maintain, however, that unwanted side-effects in teaching programming existed. Pupils who studied programming thought they knew more than they really did. Some critics held that high level languages were becoming outdated and should not be taught; fourth-generation languages should be taught. Fritz countered this argument by stating that the latter provided no standards and could therefore not be taught in schools. The former provided a valuable aid in teaching computer concepts. (Fritz 1985:707).

Languages taught in schools have been in the main Basic, Pascal and LOGO. Prolog has been put forward as a language that "forces pupils . . . to order information consistently, to express themselves precisely and to make explicit logical connections between different events . . ." (Dean, Briggs & Nichol 1987:175). Pea wrote "we also need to rethink, in ways suggested by Midian Kurland, the educational goals that programming is meant to fulfil." (Pea 1987:156).

Another effect of teaching programming at schools is the necessary demand on computer resources. Piddock (1987:166) quoted Whitwell in stating that because of the heavy programming load at the 16+ level in the schools in the UK, the computers could not be used fully in a cross-curricular fashion, i.e. there was a conflict for the resources.

2.4.3 The trends in Computer Studies in the UK

2.4.3.1 Pre-1980

Computer Studies began as a study of the computer and the stress was on 'how they worked'. The subject was divided into hardware sections without much integration of these sections. Teaching methods followed the traditions of teaching in Mathematics, i.e. all answers were either right or wrong and no discussion was encouraged. The practical element of Computer Studies consisted of programming, whether this was practically feasible or not.

Most of the programming exercises were related to mathematical problems. (Griffin 1988:443).

2.4.3.2 1980-1985

In 1980 McGee had the following comments to make on the various syllabuses of Computer Studies at both O and CSE level: He stated that initially computers were used for scientific work and had therefore been associated with Mathematics and Science. (McGee 1980:75). He found that many questions were to do with Mathematics and not much with computers in examination papers. He felt that educators had been slow in following the trend of the use of computers in other areas. He noted the extension to the use of packages in some syllabuses, and appeared to favour the approach of the London O-level syllabus. This syllabus attempted to expand the curriculum into the use of packages and set up objectives for the course. (McGee 1980:74).

Hennesy & Croft stated that programming with (for example) Basic should not take up more than 10% of the total course. They viewed program entry as the main bottlenecks in the practical aspects of the course, exacerbated by the limited amount of hardware available. (Hennesy and Croft 1980:34).

Longworth noted problems associated with the teaching of Computer Studies:

> To teach about the machine as a piece of technological wonder is one aspect of the current preoccupation with trivia . . . similar arguments apply to the study of logic gates, careers in computers . . . ,

i.e. he felt the subject matter was "over technical and over intellectual". Furthermore he stated:

> indeed there is some evidence that such a study has an inhibiting effect on the future learning curves in entrants to the Data Processing profession.

He noted the preponderance of mathematicians in the examining bodies, the lack of appropriate pre- and in-service training for teachers, and lack of adequate resources and teaching material. He stated that the need was not to teach pupils about the technology, but about the impact of computers, the expanding use of computers in the future, and their envisaged social impact. (Longworth 1981:17-19).

Ruthven criticised the technical aspect of the Computer Studies courses that stem from what he called "the pursuit of academic respectability". As already mentioned, he expounded on what the subject should stress: the relationship between technology and man, with technology challenging the way we think about intelligence and creativity. (Ruthven 1981:5).

The London O level Board recognised that the course should stress the use of the computer rather than its more technical aspects in the new 1980 syllabus. Lees, with the British Computer Society, felt that all pupils should have some understanding of computers. Some should be able to program, but the first two years of secondary education should provide computer awareness to all pupils. This could then be followed by a course in programming as an option available to all. (Lees 1981:38-42).

Wellington stated that there were criticisms of some topics in the course such as the history of computing, binary representation, and the use of Basic as a programming language. The fact that the course attracted more males than females was also seen to be a problem. (Wellington (1990:58).

The overall criticisms of syllabuses prior to 1986 were that the subject was far too technical, and that the course was divorced from the use of computers in the real world. (Griffin 1988:443-444).

2.4.3.3 Post 1985:The GCSE and A level

The new GCSE syllabus of 1985 answered the main criticisms of the previous syllabuses. There was a move to a more application-based approach in the subject of Computer Studies. There was an increase in emphasis on the "study of applications of computers and the development of practical activities into more useful tasks". (Griffin 1988:443-444). Rich mentioned that industry representatives opined that the A level syllabus should deal more with applications than it did. They also felt that the syllabus could never be up-to-date, and that the topic was not appropriate at school level. (Rich 1986:19).

Neal discussed the new GCSE syllabus, noting that the emphasis was on problem solving rather than on programming. The use of packages was seen as important.

> . . . the designers have taken the opportunity . . . to move the subject firmly towards the study of how computers are used and away from a study of how they work. . . . Computer Studies is an excellent vehicle to use for integrating a curriculum that remains subject orientated. (Neal 1986:3).

McGee wrote that the GCSE Computer Studies demanded that teachers lay greater stress on analysing and investigating problems, and that an application approach should be used with an emphasis on real world applications. (McGee 1986:5-6).

Piddock mentioned the problems attendant on Computer Studies in schools, i.e. cost of hardware and software, lack of books and resources and teacher shortage. (Piddock 1987:167).

The aim of the GCSE syllabus as developed after 1984 was to have an even greater systems applications approach and human bias, with a major emphasis on the use of the computer in problem solving, i.e. the "practical nature of the subject, and the applications of computers". (Griffin 1988:444). Griffin assessed the theory underlying this move. She wrote that if one accepted the Piagetian view of development (with Shayer's adjustment for age), most of the teaching ought to be aimed at the concrete operational stage at the 14+ level. Learning would best take place when new experiences were related to existing knowledge. She maintained that the application approach helped learning in this manner. She mentioned Bruner's spiral curriculum where pupils are "exposed to subject matter which ensures both overlap with previous material and a steady progression in the complexity of the material", i.e. by revisiting concepts at different levels. Again the application approach of Computer Studies would fit this type of learning. She stated that the aim in Computer Studies was to

> educate students to be aware of the relevance of computers in the 'real world' and of the vital importance of computer technology to modern society. ...The specific vocational aspect is a relatively low priority in considering GCSE courses. . . . Computer Studies is concerned with much more than computers. It is also concerned with the organisation into which they are placed and the people who will work with and otherwise be affected by them.

Griffin states that computing has been taught at 16+ level for over 20 years and has steadily gained in popularity. Computer Studies has now moved away from its mathematical beginnings and is mainly concerned with problem solving through the use of computers. (Griffin 1988:443).

Selwood (1988:3) discussed what the application approach should comprise, but said that textbooks were not available yet to support this approach. A textbook by Kennewell et al. is now available which supports this approach. (Kennewell et al. 1989).

2.4.3.4 IT in the curriculum

With the new national curriculum, Design and Technology is to be part of the core curriculum up to 16+. Information technology is subsumed in this, with the target of pupils being able to use technology and assess its effect on society. This will result in the following: "Computer Studies or computer science as a specialist subject at GCSE is likely to find less room in the school timetable and become integrated into design and technology teaching " (Education Broadsheet 1990). The final level of the IT National Curriculum consists of what was 'Computer Studies' formerly.

2.5 Computer Studies at schools as related to university courses in S.A. and elsewhere

In the U.K., the syllabuses were originally closely controlled by the university entrance exams. This implied that there could be a natural progression from what was taught at school to what was taught at university. With the new National Curriculum, the influence of the universities has waned. The situation there is now closer to that in South Africa where, theoretically, the universities might be totally excluded from the design of the school syllabuses. This could lead to problems at university level in teaching the related subjects.

2.5.1 Overlap

What is taught at school echoes some of what is taught at first year level at universities. This would appear wasteful of resources, and may cause some problems at university level.

Terry mentioned some of these problems viz. teaching difficulties are caused because some students start their first year of Computer Science at university having matriculated with Computer Studies, while the rest are normally ignorant of the subject. (Terry 1990 pers.comm.). This is a problem very similar to the one experienced in Accountancy. (Rowlands 1988).

Rich (1986:16) stated that Computer Science departments at universities in the UK preferred students to have A levels in subjects other than Computer Studies. He quoted statements from universities and polytechnics stating that there were two problems with students taking Computer Studies at schools: first year courses bored such students, and students who had been badly taught needed to be retaught. He stated that these problems should be redressed.

Maddison (1983: 76) quoted an article by K. Hennesey, lecturer at the University of Manchester Institute of Science and Technology about A-level Computer Science who stated more critically:

> We find that, in general, entrants with A-level computer science have to unlearn a great deal of irrelevant, out-of-date and incorrect information and skills . . . Furthermore, they learn the most appalling programming habits because their teachers are by and large, amateurs . . ..

This opinion is echoed by Longworth (1981).

2.5.2 Drop-out rate

From discussion, the drop-out rate of pupils starting Computer Studies at school in South Africa is too high in certain areas. (Ross 1990 pers.comm.). This may affect the numbers taking Computer Science at university by demotivating prospective students. There is a problem of falling numbers of students taking Computer Science at universities. This would seem to be the position in the U.K. A report in the Guardian quoted by Durndell, Macleod and Siann (1987:167) stated that the current "number of university applicants making a course in computer studies their first choice has fallen by 11.5%." Similar fears were expressed in the States. Leith (Leith 1990) referred to the fall in numbers of Computer Science applications in colleges in the States. From discussion with Standard 8 pupils considering dropping out of Computer Studies, the following reason appears to be a factor in their decision: Among the pupils who take Computer Studies at schools are some who are already 'computer literate', and who have started programming at a fairly advanced level. These pupils would like to expand their programming skills i.e. become more proficient at programming in perhaps more trendy languages like C. They are more interested in the practical rather than theoretical aspects of computing. (Van Niekerk pers.comm.).

2.5.3 Content of university Computer Studies courses

Is what is taught at universities in Computer Science and related courses above criticism? Reeves and Bussom, as early as 1979, questioned whether what was taught at university was relevant to what was needed by industry in Management Information Systems (MIS) courses, since much of it was too technical. (Reeves & Bussom 1979:21). Barron criticised the subject as taught at universities as being too introspective and of ignoring applications. He saw this as the result of the "superfluous nature of computer science as a discipline". (Barron 1990:1). Skok stated that the subject Computer Science at university had moved into a more business orientated framework. "However, most computer science degrees retain some mathematical topics for historical rather than industry related reasons." (Skok 1988:27). King (King 1986) stated that industry needs were better met by business graduates who could undergo computer training than by computer science graduates. ". . . IS (Information Systems - author's note) directors say that they expect to hire fewer and fewer with technical (i.e. Computer Science - author's note) degrees." (Davis 1989:66), and "the technical/scientific background of new employees who are being recruited is largely irrelevant." (Steiner 1991:2); while Visser stated "industry requires qualified computer scientists . . . in order to cope successfully with future computing technology". (Visser 1991:2).

2.6 Conclusion

Some of these issues seem to have caused more concern in Britain and the U.S.A., whereas others have been raised more frequently in South Africa. How valid are the opinions expressed? Very little South African literature exists on the subject of computer science/studies in the school and university curricula. The subject is still young in South Africa. The research described in the following chapters will, it is hoped, cast some light on the existing state of affairs.

Chapter 3

An analysis of the trends in Computer Studies syllabuses

Outline of Chapter.

3.1 Introduction

This chapter first examines the trends in Computer Studies school syllabuses. It was decided to limit the analysis to the Higher Grade syllabuses as these syllabuses would have the most impact on university computer science syllabuses. The syllabuses of the CED are used as examples of the trends in South Africa, since the four provincial syllabuses were all based on the same core syllabus. The trends are compared to trends in O-level and GCSE syllabuses in the United Kingdom (U.K.). The purpose is to examine the differences between the two countries' school syllabuses, and to examine whether the changing trends in the U.K. have any relevance for S.A.

Then various South African university first year Computer Science syllabuses are examined for commonality, and compared with two selected United Kingdom syllabuses.

3.2 S.A. Syllabus 1985

This was the first Higher Grade syllabus implemented in South Africa.

3.2.1 Aims

The aims as expressed in the syllabus can be grouped into categories:

(a) A knowledge of the computer, where it can be used, and its the social implications.

(b) A knowledge of data collection and provision of information.

(c) Problem solving abilities, developing logical and critical faculties, developing skills in manipulating symbols, with emphasis on the development of algorithms.

(d) Wider aims namely "To stimulate interest in further computer study".

(e) Introduction to software packages.
(see Appendix B:1 Computer Studies Syllabus for Higher Grade 1985)

3.2.2 Assessment

The practical content counts for 30% of the final mark, while the examination counts for 70%.

3.2.3 Content

The syllabus is divided up into contents for Standards 8, 9 and 10. The contents for the three standards are combined and are described in detail to form a base for comparison with the new syllabus. The contents are:

1. Algorithm design including "binary selection construction", "loop control" (Std 8); evaluation of structure in algorithm design, and modular design (Std 9); problem analysis using specification languages or flowcharts; trace tables for debugging (Std 8).

2. A model of the computer (Std 8); the control cycle and properties of secondary storage (std 9); microcomputer hardware and data communications (Std 10).

3. Data representation including binary conversion, binary addition, overflow errors, and character representation (Std 8); data representation including conversion to binary and hexadecimal, internal representation of negative numbers, floating point numbers, arithmetic and errors (Std 9); addition and subtraction in hexadecimal and conversion between decimal, binary and hexadecimal (Std 10); data representation including

arrays and lists, as well as stacks, queues and trees, and storage of data in files (Std 10).

4. An introduction to logical operators (Std 8); the logical data type and de Morgan's Laws (Std 9); logical operations and truth tables extended from the work done in 8 and 9 (Std 10).

5. An introduction to systems software and application packages (Std 8); system software including operating systems, assemblers and compilers, data communications (Std 10).

6. Programming - introducing a language, and debugging techniques. LOGO is suggested as a programming language (Std 8). Another programming language is introduced, as is the use of application packages such as wordprocessing and spreadsheets, and data base systems (Std 9). Programming includes elementary file handling (Std 10). File access and databases are dealt with (Std 10).

7. Application packages, and 4th generation languages (4GL's) are added (Std 10).

8. The social impact of computers (Std 8, 9 & 10).

3.3 Proposed S.A. Syllabus 1993

The proposed new syllabus does not deviate greatly from the old in the aims:

3.3.1 Aims

The general aims are identical with those of the previous syllabus. In the introduction, however, it specifically stated that the course should provide the pupils with "the foundation upon which to build further studies at a tertiary (university) level" or provide them with "valuable insights into the computer field" if they do not continue with further study of the subject. (Appendix B: 2 "Computer Studies Syllabus for Higher Grade 1990"). It also stated that the emphasis would be on more theoretical aspects of the subject.

3.3.2 Assessment

The evaluation guidelines are that the theoretical aspects should not count for more than 75% in each year, the practical aspects should not count for more than 40%, and in the examination the algorithmic or practical content should count between 40% and 60% of the paper.

3.3.3 Contents

The proposed syllabus is designed differently in that (i) topics are required to be presented in the "spiral didactic model", and (ii) the course is divided up into modules of which some are compulsory and others optional.

The following five are the compulsory modules:

1. Algorithm Design in this syllabus is similar to the previous syllabus (including the previous problem analysis, but leaving out stepwise refinement and problem specification). Recursion is added.

2. The Programming module is similar to the previous syllabus but with more clearly defined contents in terms of modular programming, debugging, file handling. There is emphasis on input validation and file handling. Programming in a second language is no longer a requirement.

3. The third module is a module on "Computer Architecture" that is similar in description to the hardware contents in the previous syllabus.

4. Data Structures is a fourth module and is similar to the previous syllabus. User-defined data types and searching, sorting and list manipulation techniques are added in.

5. Systems Software is the final compulsory module whose contents are similar to those of the previous syllabus.

The optional modules then follow.

1. First there is Development of Computers and Social Implications. The development is more clearly defined and in more detail.

2. Computer Logic is another optional topic that has the same contents as before, but also includes elementary logic circuits.

3. Data Communications has been expanded markedly, and is almost totally new.

4. End-user Computing (EUC) has been expanded from the previous syllabus, to include 4GL's, utility and graphic packages.

5. Modern Computing is another optional module that "must be seen as an open-ended module" including 4GL's, AI, Robotics, Expert Systems and CASE and Fifth Generation computing. This is totally new.

6. Software Engineering is also new and includes the System Development Life Cycle (SDLC), Feasibility Studies, Logical and Physical Systems Design, Project Management, Documentation, Testing, Implementation, Security and Privacy, and Maintainability and Portability of Systems.

The compulsory modules make up 75% of the course in time, whereas the optional modules varying between 5%, 10% and 15% make up the remaining 25%. The compulsory practical component is 50% (hours) of the course, whereas there is a further 15% possible in "End-user Computing".

3.4 A comparison with UK Syllabuses

This section highlights the differences in the proposed matriculation Computer Studies syllabus and two selected GCSE syllabuses, namely those of the London and East Anglia Group (L.E.A.G.) and the Joint Matriculation Board (JMB). It is recognised that the standard of matriculation and GCSE may not be the same, since the matriculation level is normally accepted to be of a higher standard than O-levels (or now GCSE), but not as advanced as the A-level. The Alternative Ordinary Level was probably of an equivalent nature to the S.A. matric, but this level has now been withdrawn.

3.4.1 Aims

The aims of the L.E.A.G. and JMB are identical, based on the aims as stated in the National Criteria. (General Certificate of Secondary Education, Computer Studies, JMB & London East Anglian Group, 1992). Many of these aims correspond to those of the S.A. syllabus. For example the two aims of the U.K. syllabuses: "to foster an awareness of what characterises information, information processing and computing systems; to develop reasoning, judgment and persistence in applying, creatively, information technology to problems" correspond to the aims in the S.A. syllabus. The U.K. aims differ in emphasis from the aims of the S.A. proposed matriculation syllabus in that they aim to "foster an *interest and enjoyment* of the use of computers; develop *confidence in the practical skills* in the use of computers; . . . offer a broad and balanced perspective of a *range of applications* to provide an understanding of their capabilities and limitations; to develop an awareness of ethical, social, economic and political consequences of the use of computers for individuals, organisations and society *through the study of meaningful applications; to develop the ability to communicate and interpret* information and concepts relevant to computing using appropriate means." (the writer's italics). The emphasis is on interest for the pupils, the study of the subject by applications, and the ability to interpret and communicate information.

The S.A syllabus's aims place greater emphasis on the development of the pupils' *problem solving ability*, the pupils' *logical and critical thinking* and their skills in the *manipulation of symbols* (the writer's italics).

3.4.2 Difference in structure

The first significant difference between the latest S.A. and U.K. syllabuses is that there are optional modules in the S.A. syllabus. In the U.K. syllabuses there is choice in the implementation of the project, but not in the theoretical topics.

3.4.3 Areas unique to the U.K. syllabuses

What both U.K. syllabuses emphasize is the introduction to the pupils of a wide range of applications. The S.A. syllabus makes this an optional topic. The L.E.A.G syllabus has developed a module called "Information Processing" that looks at systems analysis and design: the description, input and output requirements, encoding of information, data capture, input validation, data verification, required files, processing requirements and file access, backup procedures and people involved, all <u>*for a particular application*</u>. Also required is that pupils should be able to describe a system and its security requirements with system flowcharts. Also included are the various possibilities for acquiring the computer hardware and software capabilities to implement the system (e.g. in-house development versus packages). Another topic "Effects of the Computer" assesses the effects of the particular application on the users and clients. The JMB syllabus does not stress to the same extent the teaching of concepts with respect to a particular application.

3.4.3.1 System Design

Both U.K. syllabuses stress the broader aspect of program or system design in that they require the pupils' to "explain the need for encoding information for computer processing" (JMB page 3) and to "recognise when a problem has characteristics making it suitable for the application of computer techniques"; to "derive the information requirements of a system"; to "specify a system precisely in terms of the output needed, the input data and any necessary files or interfaces". System flowcharting is also required. Also required by the JMB syllabus is to "interpret information about systems presented in a variety of ways". (JMB syllabus page 1). All topics above may be covered in the module on Software Engineering in the S.A. syllabus, but this is an optional module.

3.4.3.2 Stored program concept

The stored program concept, representation of instructions, and levels of programming languages are contained in both the U.K. syllabuses, but not in the S.A. syllabus. The L.E.A.G syllabus requires pupils to be able to describe and distinguish between the purpose of utility and application packages and to evaluate design criteria for good applications packages. The use and evaluation of packages to solve problems is also required by the JMB syllabus.

3.4.3.3 Practical work

The practical project in the U.K. syllabuses requires the candidate to design a system and to implement it by either writing programs or by using pre-written software packages or a combination of the two. It also requires a project report. The S.A. syllabus requires the system or program to be implemented by the pupils' own programs only. The JMB syllabus stresses that "it is essential that all candidates can use a data handling package, can prepare and edit documents using a word processor and can manipulate information on a spreadsheet". (JMB syllabus page 3). Some experience with graphic packages and query languages is recommended. This is an optional module in the S.A. syllabus.

Two other topics that are not in the S.A. syllabus but are in the U.K. syllabuses is the issue of backup and recovery, and the practical requirement to develop a control system. (JMB page 4).

3.4.4 Areas unique to the proposed S.A. syllabus

3.4.4.1 Minor inclusions in the compulsory topics

Contained in the S.A. syllabus is the "evaluation of the structure and efficiency of an algorithm" (Appendix B:2). This may be included in the UK syllabuses, but is not explicitly stated. It also contains requirements for explicit programming techniques such as variable naming and indentation, and modular programming features such as global variables. It still contains the requirements for other numbering systems e.g. hexadecimal, and conversions between them. This has been dropped from the U.K. syllabuses. Floating point representation is included in the S.A. syllabus only, as are also data types, and data structures such as arrays, stacks, queues and trees, and list manipulation techniques. These are the main inclusions in the compulsory modules. In the optional module on Software

Engineering there are inclusions unique to the S.A. syllabus, particularly the "project management, maintainability of systems, and portability of systems". Depending on how it is interpreted, feasibility studies could be seen as common to both syllabuses.

3.4.4.2 Major differences

The rest of the optional modules cover a far wider area than the U.K. syllabuses. The S.A syllabus has an optional topic that encompasses the historical development of computers, and discussions of A.I. and computer viruses. The two U.K. syllabuses ignore these totally. Another module, the one on Computer Logic, is almost totally excluded from the U.K. syllabuses. The Data Communications module is also excluded from the U.K. syllabuses with only a passing reference to Bulletin Boards. The topic on Modern Computing is unique to the S.A. syllabus.

3.5 Summary of school syllabus comparisons

The S.A. school syllabus has topics that have been discarded in the U.K. syllabuses e.g. numbering systems and computer logic. It emphasizes newer technical aspects of computing in the optional topics. It enforces computer programming whereas the U.K. syllabuses allow a variety of software solutions. The U.K. syllabuses emphasize the system aspect of computing, and the introduction of concepts through applications. The U.K syllabuses require a study of the ethical and social aspects of computing, whereas in the S.A syllabus this is optional. The U.K. syllabuses also place greater stress on the evaluation and use of common packages. This again is optional in the S.A. syllabus. The South African syllabus allows imaginatively for future changes in the subject. The designers recognise the fact that the syllabus, in such a fast-changing subject, lags behind reality, and have tried to compensate for this.

3.6 Commonality and differences in university first year syllabuses in Computer Science

3.6.1 Introduction

The universities that were likely to be fed by DEC school matriculants were asked for their detailed first year Computer Science, Business Information Systems or Software Engineering syllabuses. The aim was to establish the commonality in first year syllabuses, and to highlight any overlap that might occur with the Computer Studies school syllabus. 10 of the 12 universities replied with descriptions of their Computer Science 1 syllabuses. Some of these descriptions were detailed whereas others were very brief. Four departments supplied their BIS and Applied Computing syllabuses. It was decided to concentrate on the Computer Science syllabuses, since these departments represented the greatest number of respondents to the university questionnaire. (See chapter 5).

3.6.2 Contents of first year S.A. Computer Science syllabuses

Although the syllabuses for Computer Science that were received from the South African universities differed in the amount of detail, there were areas of commonality, as would be expected.

(See Appendix C: Tables 1 and 2 for greater detail).

The areas of commonality were the following:

1. All the first year syllabuses dealt with algorithm concepts, design of algorithms, analysis of algorithms and computation concepts in varying degrees of depth.

2. 60% of the universities mentioned Pascal as their standard programming language, two mentioned Modula 2 and the rest did not specify. 40% of the responding universities itemised programming and debugging techniques as part of the syllabus.

3. Hardware and communications i.e. computer architecture, was mentioned by almost all the universities with some defining more depth in this area than others.

4. Binary numbering systems and boolean algebra were mentioned by 60%. Integers, floating point and strings were also itemised by several.

5. Data structures and types was very heavily emphasized by all. This included records and files, lists, stacks, trees and queues, and arrays and sets.

6. 33% of the respondents mentioned application packages to varying degrees of intensity. These included spreadsheets, wordprocessing and database management packages.

7. There were some items mentioned in turn by the respondents which were common to at least 25% of the S.A. universities: searching and sorting techniques; recursion; problem solving; system software including operating systems (OS) and database management systems; history of computer developments; and systems analysis.

8. Thereafter there were divergences with different departments teaching some of the following: Artificial Intelligence concepts, Object-Orientated concepts, logic circuits and circuit design, linear grammar, microprogramming, parallel processing, and social issues.

3.6.3 Comparison with a first year curriculum in Computer Science in the U.K.

The situation at United Kingdom Universities is somewhat different in that a degree can be taken solely in Computer Science. This means that in such a degree eight or more first year Computer Science courses can be taken. These courses vary in length from 10 lectures to 30 lectures with an average of 22 lectures spread over 11 weeks. A combined degree such as Computer Science with either Mathematics or Electronics is perhaps closer to our degrees, and here four to six first year Computer Science courses are offered. The Computer Science courses for such a degree will be examined.

The university chosen for comparative purposes was King's College London, since this would be a natural progression for students who had written the L.E.A.G GCSE course, (which was examined earlier in this chapter). Warwick University was also briefly examined since it would possibly be a natural continuation for the JMB schools. This comparison was undertaken to establish whether the same problems of overlap occurred between the school and university Computer Science syllabuses as reportedly existed in South Africa.

3.6.3.1 King's College syllabus

For a course in Computer Science and Mathematics in King's College, four Computer Science courses were offered. These were: a course in "Introduction to Computer Systems", two courses in structured programming, and a course in low level programming.

The contents of these courses were the following, with an estimate of the proportion of S.A. universities offering the same topics in brackets:

Contents of courses	Proportion of S.A. universities offering the same topics.
Introduction:	
Control cycle	(Most)
Data representation	(Most)

Radices and conversion	(Some)
Conversion between real and strings	(Some)
Binary addition and subtraction	(Some)
Two's complement	(Unstated)
Floating point numbers	(Most)
Error detection	(Unstated)
Wordprocessing and spreadsheets	(Many)
Hardware (Architecture)	(All)
OS in detail	(Many,but more cursorily)
Communications	(Most)

<u>Structured Programming 1</u>

The use of Pascal	(Most)
Data types: record,arrays,files	(All)
System Development Lifecycle (SDLC)	(Some)
Structured program design	(Some)
Advanced 8088 features	(Unstated)
Software quality	(Unstated as such)

<u>Structured Programming 2</u>

Data structures: stacks, queues, lists, trees	(All)
Recursion	(Many)
Sorting: internal, external	(Many)
Searching	(Many)
Multidimensional access	(Not stated as such)

<u>Low level programming</u>

A simplified microprocessor system	(One)
Assembly language programming	(Two)
Interfacing	

It was not possible to be more exact about the number of S.A. universities teaching the above topics, since the detail provided in some syllabus descriptions were insufficient and assumptions were drawn from the descriptions.

Four other courses were offered by King's College which could be chosen for different combination degrees with Computer Science.

One course was called Basic Computing Theory including sets, functions, techniques of proof, sequences, graphs, computability, probability and statistics. Another course offered dealt with System Analysis including data flow diagrams, data entity relationships, and comparative analysis of methodologies. Introduction to Formal Methods i.e. mathematical specification of software systems including basic mathematical notation of sets, relations, functions and sequences and schemas was included in the third course. Another course on Logic Design was offered including boolean algebra, gates, registers, and memory characteristics. (Note - these courses are not reflected in the tables in the Appendix).

3.6.3.2 Warwick University Computer Science

At Warwick university, for a similar Mathematics and Computer Science course, there is a choice of 4 first year courses. Although the course on programming is similar to the King's College course, the Warwick University choices for this degree reflect a more theoretical approach, "intended for mathematically able students interested in the theoretical basis of Computer Science" (Warwick University 1990:5), with the following stated contents in the topics that differ from the King's College choices:

- A series of programming exercises such as recursive programs; pattern matching; path-finding in graphs;Use of software tools; command language programs.

- Programming concepts including first order logic, predicate transformers, guarded command languages.
- Discrete mathematics including elementary graph theory, boolean algebra and lattices, and finish state automata.

3.6.3.3 University comparisons

The King's College choices were more like the S.A. syllabuses than those of Warwick university. Some of the South African syllabuses offered additional newer topics in their first year. One in particular offered a more mathematical topic in propositional calculus and linear grammars. Database theory was ignored in the King's College first year topics. South African universities diversely offered Artificial Intelligence, Parallel Programming and Object Orientated design topics. End-user Computing (in the form of spreadsheets etc.) was mentioned, but very cursorily, in the U.K. syllabuses. The 'softer' topics such as history and the social implications of IT were ignored by the King's College syllabus but were included in some of the South African university topics.

The major difference between the U.K. and S.A. syllabuses was firstly in the area of choice. The U.K. syllabuses allowed for greater flexibility by offering a choice of first year modules. This allowed for greater specialisation earlier on. However, this study does not take into account topics offered in both countries at second and third year levels. The other major difference was the somewhat more technical approach at the U.K. universities with the option of for example studying assembler for 24 lectures (or logic design for 10 lectures and 15 hours laboratory work at Warwick) at first year level.

3.7 The progression from school to university syllabuses

In comparing the progression from school to university, the latest proposed school syllabuses in S.A. and of the L.E.A.G. are compared to a combination of the South African university modules, and also the King's College Computer Science syllabus. See Appendix C: Table 3 for a condensed comparison between S.A. and U.K. schools and universities. This table uses as a base the summarised university first year syllabus for Computer Science.

The following would appear to be the case:

(i) Almost all the topics in the S.A. combined university syllabus are also dealt with at school level.

(ii) The S.A. combined university syllabuses have many areas of commonality with the King's College syllabus. Algorithm design, computation concepts, program development, boolean algebra, introduction to systems software and languages, modern trends, social issues are largely absent from the modules chosen to reflect the Mathematics and Computer Science degree in the King's College syllabus.

(iii) The areas of commonality (or overlap) between the U.K. L.E.A.G syllabus and King's College syllabus appear not very high - presumably this can be attributed to the application approach of the school syllabus, and the de-emphasis of programming.

It merits further research to find out what proportion of U.K. students have Computer Studies as a school leaving subject.

(vi) There are similarities between the two countries' school syllabuses: the main difference appears in the more theoretical approach of the S.A. syllabus, and the emphasis on the application approach and the use of computers in the U.K..

3.8 Conclusion

The next chapter will examine the motivation of the designers of the proposed S.A. syllabus for the changes it reflects, and their underlying assumptions about Computer Studies. Thereafter, the university lecturers' opinions about the subject of Computer Studies at school and its relationship with first year Computer Science at universities will be examined.

Chapter 4

The role of Computer Studies in S.A. from the point of view of the designers of the syllabus

Outline of the chapter.

4.1 Introduction.

4.2 Aim of the research.

4.3 Description of the research.

- 4.3.1 The population.
- 4.3.2 The survey.
- 4.3.3 Contents and structure of the questionnaire.
- 4.3.4 Grouping of the questions in the first section.
- 4.3.5 Limitations in the design of the questionnaire.

4.4 Findings from the designers' questionnaire.

- 4.4.1 Scaling of the responses.
- 4.4.2 Response from Section 1 of the questionnaire: why Computer Studies at schools?
  - 4.4.2.1 Ranked reasons for offering Computer Studies at schools.
  - 4.4.2.2 Agreement on less important reasons.
  - 4.4.2.3 Contradictions.
  - 4.4.2.4 Other reasons for offering Computer Studies.
  - 4.4.2.5 Rationales for offering Computer Studies.

- 4.4.3 Response from Section 2 of the questionnaire: The aims and improvements of the new syllabus.
  - 4.4.3.1 Problems with the old syllabus.
  - 4.4.3.2 Improvements in the new syllabus.
  - 4.4.3.3 Possible problems with the new syllabus.

- 4.4.4 Response from Section 3 of the questionnaire: Problems with Computer Studies in schools.
  - 4.4.4.1 Problems with the teaching of Computer Studies.
  - 4.4.4.2 Problems experienced in offering Computer Studies.
  - 4.4.4.3 Other problems and comments.
  - 4.4.4.4 Limitations of this part of the survey.

4.5 Conclusions from the survey.

4.1 Introduction

For this study, the descriptive research method was used. According to Gay (Gay 1987: 189), such a method is used in a high percentage of educational research. In particular, a 'self-report' survey was used to try to establish what the designers of the new syllabus's underlying rationale(s) were in creating the new syllabus, and what their opinions were of the difficulties involved in offering such a subject.

4.2 Aim of the research

In particular, the aim of this part of the research was to discover the opinions of the designers of the new Computer Studies syllabus on the following:

(a) The role of the subject as part of the school curriculum.

(b) The shortcomings of the old syllabus; and how the new syllabus would meet these shortcomings.

(c) The criticisms voiced of the subject Computer Studies as offered and taught at school.

(d) The relationship between the subject and the first year of university Computer Science or Applied Computer courses.

The opinions are examined in the light of the trends in the S.A. and the U.K. syllabuses as discussed in Chapter 3.

4.3 Description of the research

4.3.1 The population

It was initially assumed that the designers of the syllabus were the Computer Studies Committees of the four provinces, and the Curriculum Committees and Syllabus Committees of the DEC. The Executive Director of Computer Studies for the Cape Educational Department provided a list of names consisting of the Computer Studies Committee of the CED, the Curriculum Committee of the DEC, and the Joint Syllabus Committee (which represents all examining bodies in the country). It was decided to confine the research to this group, because the Cape Education Department would be represented, and the persons serving on the higher committees would reflect the opinions of the committee members of the other three provinces.

4.3.2 The survey

In descriptive research, "data can be collected either by direct observation or by personal communication with the respondent". (Rowlands 1988:117). Personal communication was chosen as the appropriate manner of conducting the survey. Churchill (1983:174) categorises personal communication research as follows:

(a) Methods of Administration:

Personal Interview.
Telephone Interview.
Mail Questionnaire.

(b) Degree of structure:
i.e. whether the survey allows for fixed responses, semi-fixed responses or free format responses.

(c) Degree of disguise:

This refers to whether the purpose of the questionnaire or of individual questions is hidden from the respondents, and to what degree, i.e. whether there is a 'hidden agenda'.

The structured-undisguised questionnaires are common. They are reliable because the fixed responses clarify the frame of reference and the questions. They can be used to "collect data on attitudes, intentions . . . and behavior". (Churchill 1983:178).

It was decided to administer a mail questionnaire because of the advantages of this type of observation: the advantages for the researcher are simplicity of administration and analysis. For the respondent, depending on the nature of the questionnaire, it allows for anonymous responses, and thus gives greater freedom in answering. It takes less time and may therefore predispose the respondent to give a better quality response. The expenses in time and travel for both researcher and respondent are minimised.

There are disadvantages to a mail questionnaire. One of the most widely quoted is that mail questionnaires offer poor control, both in getting a response and in ensuring the quality of the response. The latter is particularly true in the case of a structured questionnaire. Loss of validity may occur because:

(i) Fixed responses may force the responder to voice an opinion even if he/she has none.

(ii) If the response category does not reflect any of the respondent's views.

(iii) If the questions introduce bias, or are 'leading questions'. (Churchill 1983:178).

4.3.3 Contents and structure of the questionnaire

Twenty questionnaires were sent. Of these half were English and the other Afrikaans. Of the fifteen replies received, twelve questionnaires were completed. (The others pleaded involvement at a higher level in syllabus design). The questionnaire is appended in both the English and Afrikaans versions (Questionnaire 1 & 2 in Appendix D).

The questionnaire was sent out in either English or Afrikaans, depending on the respondent's title and address. The questionnaires did not require identification. Respondents could be anonymous, thus encouraging a personal rather than a committee response and greater freedom of expression.

The questionnaire was divided up into three sections. The aim of the first section was to gain the respondent's opinion about the importance of and the reasons for offering Computer Studies in schools in South Africa. The format was structured in that fixed responses were provided. However, to attempt to overcome the problem mentioned above i.e. to allow for deviations from the fixed responses and for possible bias in the answers provided, comments were asked for. There was no degree of disguise in respect of this aim. Underlying the questions, however, was an attempt to guide the questions in terms of the rationales, as described by Hawkridge. Here there was a degree of disguise, but only in so far as the word 'rationale' was not explicitly used. Certain questions were chosen to reflect the rationales as posed by Hawkridge in Chapter 2. (See Appendix D for the individual choices).

The second section of the questionnaire was totally undisguised in its aim and was unstructured i.e. it comprised a free format structure by which to elicit personal opinions on the merits and disadvantages of the syllabuses.

The third section of the questionnaire's structure was similar to the first comprising fixed responses with free comment. The aim of this part of the questionnaire was to gauge the opinions of the respondents on problems and criticisms that have been raised about the offering and teaching of Computer Studies as a seventh subject in South Africa. Here again, there was no disguise.

4.3.4 Grouping of the questions in the first section

The questions in the first section were formulated to extract underlying rationales as mentioned above. The table 4.1 below gives the questions in abbreviated form grouped by rationales they were chosen to reflect.

Table 4.1 Questions grouped by rationales

Rationales	Q.no	Question/Statement
Social	1a	Schools should prepare (all)students for a future requiring the use of computers
	1b	Parental pressure for computing in schools
Catalytic	1h	Different type of learning would be encouraged
	1f	Exploratory learning would be encouraged
	1g	Encouraged cooperative learning
	1i	Problem solving skill would be enhanced
Pedagogic	1e	Provided further stimulus for gifted children
	1d	Positive affect on acquiring mathematical skills
Vocational (& IT)	1j	Computer skills are required by the economy and schools should provide the training
Social & Vocational	1c	Pupils should learn to operate computers to enter the workplace with basic skills.
Syllabus aim	1k	Good basis for Computer Science at university

4.3.5 Limitations in the design of the questionnaire

A pilot study of the questionnaire was not carried out, because of the small size of the population targeted. It was felt that the respondents' willingness to reply would be exhausted by administering a pilot survey, and the population size would be decreased unacceptably. It was hoped that the comments asked for would allow any inadequacies in the design of the questionnaire to be high-lighted and obviated. Some comments did in fact raise criticisms of the third part of the questionnaire. Respondents felt that the questions should have stated "offering" of Computer Studies as opposed to the "teaching" of Computer Studies. One respondent also expressed the reservation that some questions were 'leading questions' thus lessening the validity by introducing bias. All expressed opinions on the stated problems, but some found difficulty in ranking what they termed 'unlike' categories. The above difficulties were valid. However, the comments that were asked for allowed for criticism, opinions and clarification of the answers given.

4.4 Findings from the designers' questionnaire

4.4.1 Scaling of the responses

The responses to individual statements were scaled from 2 to -2 and aggregated to give an agreement rating on a scale from a maximum of 24 to a minimum of -24. For example, question 1a (whether schools should prepare pupils to deal with computers as part of their required life-skills) had the following responses:

Agree Totally	Agree	Unsure	Disagree	Disagree totally
7	1	2	0	2

The scaling factor assigned was:

2	1	0	-1	-2

This allowed for an agreement rating to be calculated by multiplying the response with the scaling factor e.g.

$$(7 \times 2) + (1 \times 1) + (2 \times 0) + (0 \times -1) + (2 \times -2) = 11$$

It was then possible to identify those statements that evoked the greatest agreement or disagreement.

In the second part of this section of the questionnaire, the respondents were asked to rank these eleven statements in order of importance. Their responses were averaged, and a ranking from 1 to 11 was assigned to the statements. See Appendix D for a summary of the results.

It was presumed that the importance of the statements (to the designers' group) would be shown in the rankings, and that the agreement rating would reflect consensus about the statement. It was also assumed that the agreement rating and ranking would correspond i.e. statements with low agreement ratings (or very negative) agreement ratings would merit low rankings. This has been found to be the case in this part of the questionnaire.

On examination it was decided that statements whose agreements ratings had an absolute value of less than 6 and a ranking of greater than 6, were of less importance: They were deemed to have low validity. The remaining (viz. 1a, 1h, 1i, 1f, 1k, ij, 1c and 1e) are shown with their rankings and agreement ratings in Table 4.2.

4.4.2 Response from Section 1 of the questionnaire: why Computer Studies at schools?

4.4.2.1 Ranked reasons for offering Computer Studies at schools

Table 4.2 Results from section 1 of the questionnaire

Rationales	Rank	Agreement rating	Qno	Question/Statement
Social	1	11	1a	Schools should prepare (all)students for a future requiring the use of computers
Social & Vocational	2	5	1c	Pupils should learn to operate computers to enter the workplace with basic skills.
Catalytic	3	14	1h	Different type of learning would be encouraged
	4	9	1i	Problem solving skill would be enhanced
	5	10	1f	Exploratory learning would be encouraged
Syllabus Aims	6	8	1k	Good basis for Computer Science at university
Pedagogic	9	12	1e	Provided further stimulus for gifted children

See appendix D for a fuller summary of the findings and calculations.

1. The contention that to prepare children for life, schools should prepare them to deal with computers, was ranked (on aggregate of the rankings) as the most important reason for offering the subject. It had a high agreement rating: a rating of 11, which was the third highest agreement rating.

2. The second ranking reason will be discussed 4.4.2.3. since it contains a contradiction.

3. The third reason by rank for offering Computer Studies at schools was the statement that learning with computers required less memorising of facts and more information-handling and problem-solving. This was agreed upon (with the highest agreement rating of 14), and was seen as an important reason (overall ranking 3) for introducing computers into schools.

4. Following in order of ranking (overall ranking 4) was the statement that the problem solving skills achieved in this subject would be transferable to other areas. The agreement rating on this statement was 9.

5. Thereafter the assertion that computers allowed for exploratory learning was ranked 5th overall with an agreement rating of 10. One comment was that the time constraints mitigated this advantage.

6. That Computer Studies at school was a good basis for continuing with Computer Science at university was next most important by rank (rank 6), with an agreement rating of 8. This statement was formulated to assess the importance to syllabus designers of this aim as stated in the introduction to the new syllabus.

7. This statement was followed in ranking (rank 9) by the opinion (of the statements that were assumed to have a high validity) that the subject would provide the gifted pupil with a further challenge. This had an agreement rating of 12.

4.4.2.2 Agreement on less important reasons

Of the statements which were deemed less valid according to the criterion in 4.4.1, the statement that caused some disagreement was that Computer Studies had a beneficial effect on mathematical skills. (A (dis)agreement rating of -2). This tallied with the lowest ranking given i.e 11th. It was therefore seen as neither very valid nor of much importance.

The assertion that pressure was put on schools by parents to provide the pupils with Computer Studies evoked mixed reactions: an agreement rating of 5, and a low ranking (8th). This contrasts with the argument raised in America that parental pressure was the cause for the introduction of computers into upper middle class schools. Wellington mentions the sociological pressure of parents to introduce IT into education (Wellington J.J. 1990:61) as does Tucker (1985). Perhaps this is not so much a contradiction as it appears: the DEC does not see this as a motivation for introducing computers and computer education, but individual and independent schools (private schools) may. One respondent felt that parental pressure was for Computer Literacy, rather than Computer Studies.

Similarly the contention that computers encourage pupils to learn by working together did not have a much support: an agreement rating of 5 and a ranking of 10.

4.4.2.3 Contradictions

The statement that evoked the most disagreement (a rating of -7) was the one based on the IT and vocational rationale: there is a shortage of people with computer skills in South Africa and schools should help to fill this gap. Strangely enough, this statement had a medium ranking (a ranking of 8). This is difficult to explain.

Also difficult to explain is the rating and ranking of the statement that children should learn to operate computers to be able to enter the work place with basic skills. It did not have a high agreement rating (5). However, it was ranked as the second most important reason for offering the subject. This statement could have been seen to reflect both the social and the vocational rationale.

A possible reason for the mixed response to these statements could be that there was an in-built ambiguity in the statements. This could have been because the statements reflected two rationales. Perhaps a more likely explanation is that most respondents do not see the role of schools as providing vocational training, but acknowledge that the country requires such skills from school leavers and this need is not met elsewhere.

4.4.2.4 Other reasons for offering Computer Studies

One respondent felt very strongly that there was only one reason for offering Computer Studies, and that was that the subject was interesting and stimulating in its own right. However, this statement could hold for many other subjects not offered at schools (such as Russian language and literature or Astronomy). Another writer stated that the subject was introduced because the facilities and teachers were available.

A respondent echoed one reason given above, that problem solving and data processing using computers affected many other subjects. He or she felt that this made an understanding of computers essential.

A very different reason given, but unfortunately not expanded upon, was that Computer Studies allowed for the development of moral values such as truthfulness and modesty. (Certainly programming is a humbling experience!).

One respondent stated that in his/her experience, many students took Computer Studies to provide a background knowledge to help them in their future careers. These pupils had no intention of studying the subject any further.

4.4.2.5 Rationales for offering Computer Studies

The results suggest that the social rationale has the most support for offering Computer Studies. The vocational rationale had some support. The catalytic rationale ranked third in importance. The IT rationale also had some support, with the pedagogic rationale lagging in having a much lower average ranking.

4.4.3 Response from Section 2 of the questionnaire: The aims and improvements of the new syllabus

The second part of the questionnaire was an attempt to elicit the particular motives for changing the syllabus. This section consisted of an open-ended question allowing for a free-format answer. (See Appendix D:5 for a summary of the responses).

4.4.3.1 Problems with the old syllabus

There appeared to be four main criticisms of the old syllabus.

First it was thought to be out-of-date. It had an ill-defined structure. Then it was seen as too rigid. Fourthly, its scope for flexibility was criticised (this could be part of being too rigid.). The emphasis on programming was also criticised. The problem with technology changes outstripping the syllabus was mentioned.

4.4.3.2 Improvements in the new syllabus

There was a strong feeling that the new syllabus had better structure and balance. The new syllabus would cater for new developments by removing outdated sections (these were not defined by the respondents), and by including newer developments. A less widely held view was that the new syllabus was more job-market related. Another view was that the new syllabus could appeal to a wider group of pupils because of the possibility of making it more application-based. The views most strongly expressed were that by providing optional modules, the new syllabus was far more flexible. This allowed for growth and change in the contents of the subject, without further syllabus changes. Also widely mentioned was the change in emphasis of programming: first that one language only would be required and that this language was not specified (i.e. allowing for fourth generation languages); and secondly that the syllabus showed a move towards application software and development of algorithms as opposed to programming.

4.4.3.3 Possible problems with the new syllabus

There were only two points raised about possible drawbacks of the proposed syllabus viz.:

(i) There might not be a sufficient number of teachers capable of teaching the optional modules.

(ii) Problems might be experienced in choosing the optional modules (this was not expanded upon).

4.4.4 Response from Section 3 of the questionnaire: Problems with Computer Studies in schools

The third section of the questionnaire attempted to elicit opinions on criticisms and difficulties that have been raised about the offering and teaching of Computer Studies in the past. The questions were grouped according to whether they reflected the difficulties of teaching the subject or offering the subject generally. The rankings and agreement ratings for these statements were calculated and analyzed in the same manner as for the first section of the questionnaire (See 4.4.1 above).

4.4.4.1 Problems with the teaching of Computer Studies

Table 4.3 Results from section 3a of the questionnaire

Q.no.	Question/Statement	Ranking	Agreement rating
3a	Lack of qualified teachers	1	20
3b	Retraining	2	7
3h	Extracurricular nature	3	7

There was a significant agreement on the validity of the three statement regarding problems with teaching Computer Studies.

The lack of enough qualified teachers was seen as a major problem. The agreement rating of this was the highest at 20, and it was also ranked the most important of the three. The fact that teachers required constant retraining was ranked second in importance with an agreement rating of 7. The difficulties experienced by pupils by virtue of the extra-curricular nature of the subject was also viewed as very important. This could be a reason for the drop out rate. This had a ranking of three and an agreement rating of 7.

The teachings problems were seen as more significant than the problems relating to the offering of the subject in schools, since overall these had the top rankings and mostly higher agreement ratings than the remaining statements reflecting the offering of Computer Studies.

4.4.4.2 Problems experienced in offering Computer Studies

Table 4.4 Results from Section 3b of the questionnaire.

Q.no.	Question/Statement	Ranking	Agreement Rating
3d	Computer literacy for all	1	10
3c	Selection a problem	2	0
3f	Costs are a problem	3	-11
3g	Outdated hardware leads to ongoing costs	4	-3
3i	Overlap with universities	5	5
3j	Unnecessary headstart	6	-2
3k	Demotivating for students	7	5
3e	Would displace an educationally sounder subject	8	-6

Ranking 1 in importance was the statement that it would be better to expose all pupils to computer literacy than to offer Computer Studies to a few. The agreement rating was 10 and the rank 1. This signified major agreement with the statement. Many comments reinforced this view.

The rest of the statements merited lower agreement ratings, and some contradictory importance rankings.

Of the statements with non-negative agreement ratings, the opinion that by selecting pupils the subject gained an elitist aura was ranked second in overall importance. However, it had an agreement rating of 0 i.e. people were evenly divided in their opinions for and against this statement.

The following two statements had agreement ratings of 5 and low rankings of 5 and 7 regarding their importance. The first was that there existed an overlap with universities' first year syllabuses. The second was that students found it demotivating to repeat work in the first year at University.

There was disagreement with the statement that Computer Studies as a sixth subject would displace an 'educationally broader' subject. The (dis)agreement rating was -6, and it was seen as the least important by rank (8th) of the proposed criticisms. A (dis)agreement rating of -2 was given to the statement that it was unnecessary to give these selected pupils a further advantage (a head start). This statement was also ranked as largely unimportant at 6th position on the scale.

However, the greatest disagreement was expressed for the statement that money spent on computer equipment could be better spent elsewhere. Despite this consensus, the costs were nevertheless seen as a problem, with a ranking of 3. A contradictory ranking was also given to the contention that changing trends in hardware leads to an ongoing expense. The (dis)agreement rating for this was -3, but it had a concern ranking of 4. This apparent contradiction was explained by one respondent in that he/she said that the costs were a major problem especially if allied to the proposal to introduce computers to a wider base of pupils.

4.4.4.3 Other problems and comments

The respondents offered general comments which cast further light on their opinions:

1) One main area of concern was the strain on pupils who take Computer Studies as a seventh extra-curricular subject. This was mentioned by several respondents.

2) The second opinion that recurred was that computer literacy should be offered to all pupils, but that the costs of hardware and software could be a limiting factor.

3) The opinion was also raised that universities underestimated the level attained in Computer Studies at schools. Some areas were thus repeated unnecessarily at universities. Universities, it was felt, should adapt to the changing circumstances at school.

4) There were a few other diverse comments. One questioned the role of programming. Another suggested providing the subject Computer Literacy for all, while maintaining Computer Studies informally for those wishing to pursue the topic.

4.4.4.4 Limitations of this part of the survey

The agreement ratings for the latter part of the survey were lower than those of section 1, and there were contradictions in the rankings. It is felt that these occurred when the questions were ambiguous. Respondents felt that rankings were difficult to assign in the third section of the questionnaire: "difficult to rank unalike statements". However, the comments highlighted the respondents' opinions on these topics despite the quoted difficulties.

4.5 Conclusions from the survey

The questionnaire did not elicit much support from the designers group for some of the issues found in the literature. There was a deal of support for Computer Literacy to be offered rather than or as well as Computer Studies i.e support for the social rationale. The group was obviously aware of and concerned about the difficulties in teaching the subject at schools. The main problems with the present method of offering the subject at school were seen to be elitism and costs of technology.

How do these opinions compare with what the university lecturers feel about the composition of the school course, the problems experienced at schools, and what effects Computer Studies has on university curricula? The next chapter will examine the difficulties faced by first year Computer Science lecturers in S.A. in teaching their subject to Computer Studies matriculants, and what their opinion is of the Computer Studies school syllabus which could or should feed students into their courses.

Chapter 5

University educators' opinions of the S.A. Computer Studies curriculum and the relationship between the school and university subject.

Outline of the chapter

5.1 Introduction.

5.2 Involvement of universities in syllabus design.

5.3 Problems with syllabus content overlap.

5.4 Role of the subject as part of the school curriculum.

5.4.1 Costs and rationalisation.
5.4.2 Advantage for pupils in taking Computer Studies at school.
5.4.3 Computer Literacy.
5.4.4 Perceived effect of Computer Studies on pupils.
5.4.5 Quality of teaching at school.

5.5 Preparation for university.

5.6 Syllabus evaluation.

5.6.1 Aspects to be changed in the school syllabus.

5.7 General comments.

5.8 Finally on a positive note.

5.9 Limitations of the survey.

5.10 Conclusion.

5.1 This survey

A survey was conducted to examine the participation in and opinion of the university lecturers of the design of the Computer Studies school syllabus and the role of Computer Studies in schools. It was decided to target the S.A. universities most likely to be fed by DEC school matriculants. A mailed questionnaire was selected, for the same reasons as given in Chapter 4. The format was structured with fixed responses required. After each response, a free format comment area was provided. Several copies of the questionnaire (see Appendix E) were sent in either English or Afrikaans to the Computer Science, Business Information System (BIS), or Software Engineering (SEng) departments of each of the 12 relevant universities. Included were copies of the proposed new Computer Studies syllabus. These were addressed to first year lecturers and department heads of these departments. Twenty six replies in total from all but one university (U.C.T.) were received, mostly from Computer Science departments.

5.2 Involvement of universities in syllabus design

Of the 26 respondents, 30% i.e. 8 had seen the proposed new syllabus, and of these, 3 had been involved for comment and one had participated in drawing up the syllabus.

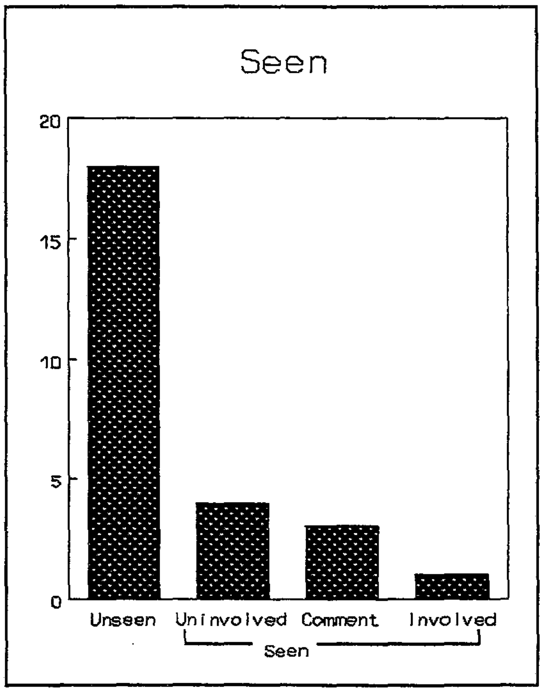

Figure 5.1: Question 4

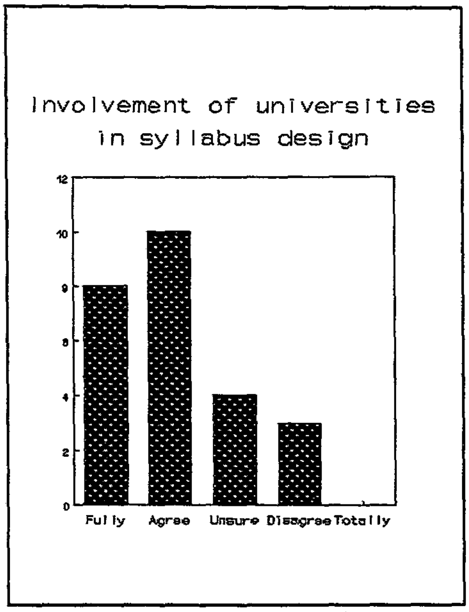

Figure 5.2: Question 7

On the role of universities in drawing up the school syllabus, 32% felt that it was vital for universities to be more involved, 40% felt it would be advantageous, 16% were unsure, and 12% felt it would not be a good idea (one respondent did not answer this question). The general feeling was that there should be closer co-operation between education departments and universities since what was taught at schools had a great effect on what the universities offered. This did not mean that the universities should have the final say, necessarily, but that both parties should be aware of what was taught at both types of institutions. More interaction could prevent overlapping/disjoint curricula and syllabus designers and university departments should co-ordinate in respect of the more modern aspects of the school syllabus. By contrast, some felt that by involving more people, no consensus would ever be reached.

5.3 Problems with syllabus content overlap

There were very disparate views on the overlap of the matriculation syllabus and the first year Computer Science, Software Engineering or Business Information Systems course, and whether this was wasteful of teaching resources. Almost 50% agreed and another almost 50% disagreed with the statement that the overlap was wasteful of resources.

Most respondents felt that the overlap was unavoidable and only serious if it affected the greater proportion of first year Computer Science students, which it did not. Many felt that the universities should adapt to the situation by, for example, introducing bridging courses (presumably similar to 'Special'

French, or perhaps an introductory non-credit course). A respondent wrote that problems arose because of poor communications between schools and universities.

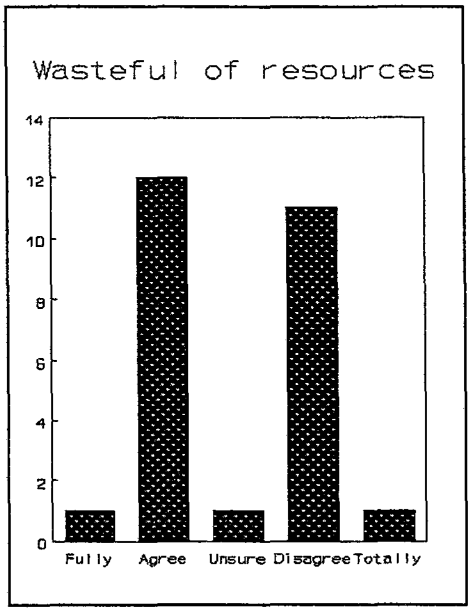

Figure 5.3: Question 9a

On teaching difficulties with two 'levels' of students, and on increasing the gap between the intellectual have and have-nots: 32% expressed the opinion that this was very much a problem, 48% replied that it was somewhat of a problem, 16% said it was no problem, and 4% (i.e. 1 respondent) felt that it was not a problem at all.

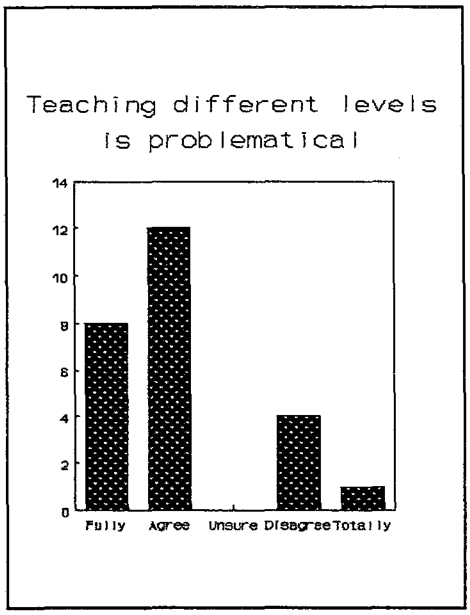

Figure 5.4: Question 9b

Respondents felt that generally it was a problem (especially for educationally disadvantaged students), but that bridging courses could possibly solve this (as above). One opinion was that the students who had not taken Computer Studies at school were often "the best" in the first year course at university (presumably because they had not acquired any bad habits in Computer Studies at school). Another opinion was that it was possible to keep the advanced pupils busy with self-initiated and perhaps advanced work until the gap closed. Parity was usually achieved by students in the second semester. This has been found to be the case in the similar case of Accounting at first year level at university (Rowlands 1988).

5.4 Role of the subject as part of the school curriculum

5.4.1 Costs and rationalisation

There was a great deal of disagreement with the contention that the subject should be addressed only at tertiary level because of the costs of technology. 27% disagreed totally, 50% disagreed, 12% were unsure and 12% agreed.

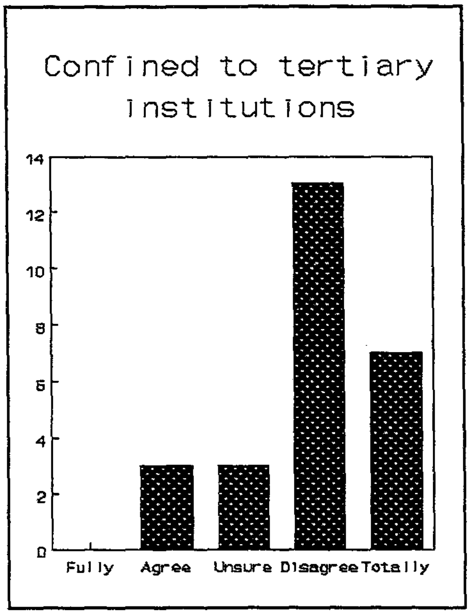

Figure 5.5: Question 9c

27% of the respondents commented further that the basic principles of computer science did not change and these should be taught at the school level. They felt that the basic principles did not require constantly changing hardware or software. An opinion was mooted that schools should possibly concentrate on theory, and therefore not require the latest hardware. One respondent felt that universities were in no better condition financially to meet the need for the latest technology. Another lecturer commented that hardware and software costs were easily afforded by schools. Three added the comment that the sooner scholars were introduced to the subject the better.

One respondent agreed that the latest technology advances should be addressed at tertiary level (leaving the basics to be taught at school). He/she mentioned all the 'useless' Apples sitting in schools. Two questioned whether costs of technology were an issue for schools.

5.4.2 Advantage for pupils in taking Computer Studies at school

Most respondents felt that Computer Studies at schools was of value: 44% very much, 32% somewhat, 16% were unsure and 8% felt that it was not.

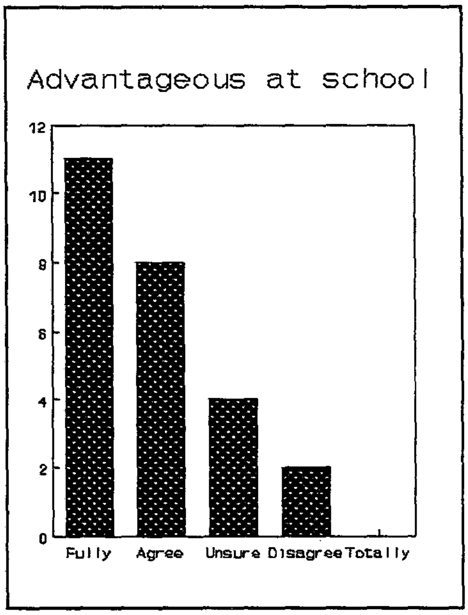

Figure 5.6: Question 91

Those who felt that it was of value gave reasons such as that it would help pupils with career decisions, and that it was good for background knowledge and 'training' programmers. It was also stated that the subject was valuable since universities could not cover all topics in this fast developing area.

Of those who were negative, there were two main reservations:

(a) the elitist nature of the course ("unfair selection") and

(b) the, at times, poor quality and lack of teachers and the difficulty of obtaining up-to-date hardware.

One opinion offered was that it was good to give the achievers challenges early, i.e. at school. This would allow them to develop even further at university, and therefore offering the subject to selected pupils should continue. Two commented that Computer Studies had value as it broadened pupils' knowledge and stimulated their interest in the subject. One respondent felt that other students (non Computer Studies students) perceived such students as having an advantage over them, that this view persisted into the second and third years of study, and that Computer Studies at school was thus advantageous for these pupils.

There was some discussion about whether Computer Studies should be offered as a sixth or a seventh subject. One respondent felt that the subject should not displace more general subjects such as History. Others felt that it should not be an extra-curricular subject, since this caused too much time pressure on the pupils.

5.4.3 Computer Literacy

However, most respondents felt that the subject, preferably in addition to Computer Literacy, should be available for <u>all</u> pupils. This came through in comments scattered throughout responses to the questionnaire.

These opinions were reiterated in response to the more precisely worded question of whether Computer Literacy should be taught to a wider spectrum of pupils rather than Computer Studies to a few. 56% agreed fully with this statement, 24% agreed, 4% (one respondent) was unsure, 12% disagreed and 4% (one respondent) disagreed totally.

Over 55% emphasized their opinion (by adding further explanatory comments) that Computer Literacy for all was preferable to Computer Studies for a few. School was seen as the correct place for this type of exposure. The opinion was also stated that two courses should co-exist i.e. Computer Studies and Computer Literacy, and that Computer Studies should still be offered to selected students as at present. Three respondents stated specifically that Computer Literacy for all was better than Computer Studies for a few. One respondent wrote that specialisation should occur later. Another said that computer education should start at primary level. It was also stated that knowledge of computers was essential intellectual equipment for all pupils, and Computer Studies gave a good background knowledge for any scholar.

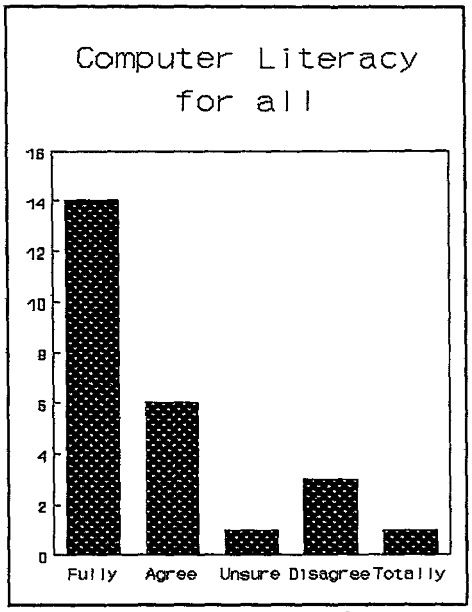

Figure 5.7: Question 9k

One respondent stated that Computer Literacy courses would provide less stimulus to encourage students to continue with Computer Science at university. Another felt that rather than provide Computer Literacy, vocational education should be provided. It is assumed that this would mean training pupils for direct entry into the job market at the operator, data-entry clerk, programmer level as discussed by Lorton and Muscat (see Chapter 2).

Another opinion voiced was that computer use should be integrated cross-curricularly.

5.4.4 Perceived effect of Computer Studies on pupils

A fear was current a few years ago (expressed in the local computer weeklies) that pupils became disenchanted with the subject at schools. These pupils were thus less likely to continue studying the topic at university: the mystique of the subject had disappeared for them. The university respondents disagreed with this statement. 20% disagreed totally, 44% disagreed, 12% were unsure, 25% agreed and 4% (one respondent) agreed totally.

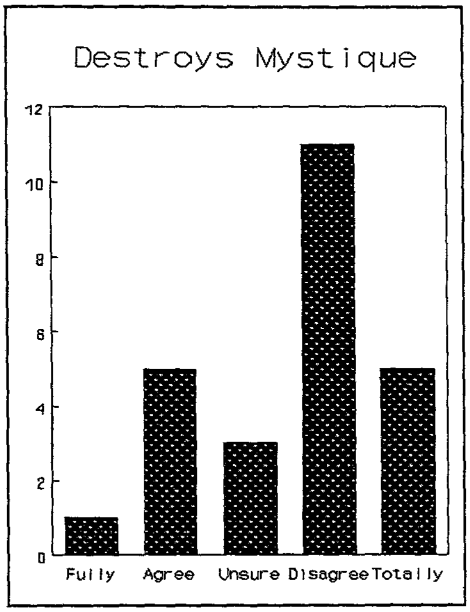

Figure 5.8: Question 9j

It was stated furthermore by three respondents that hackers or novelty seekers were not wanted at university, and that by taking the subject at school scholars could test their interest and aptitude. If the subject was properly taught it should increase interest in the topic. It was felt that bad teaching could "switch them off" (but this holds similarly for any other subject).

Of the more negative lone comments were (i) that many pupils seemed to become disenchanted by the subject at school level, (thus supporting the statement above); and (ii) that the emphasis was too much on programming at school. Another stated that Computer Studies at school encouraged hacking. Another that problem-solving was not well taught at schools and that pupils wanted to program, not analyze problems.

5.4.5 Quality of teaching at school

Respondents opined that a problem existed because the poor quality of teaching at school level made it necessary to reteach the students at university level. 15% saw this as a great problem, 44% saw it as a problem, 15% were unsure, 19% did not agree and 7% of the respondents disagreed totally.

It was stated by some that there were many under-qualified Computer Studies teachers, and that unqualified teachers should not offer the course. Competent experienced teachers were essential. One respondent stated that more and better qualified teachers were emerging, and that the problem would disappear.

However, many felt that reteaching certain areas was unavoidable regardless of the quality of teaching, because of the uneven mix of students (a small percentage only having taken Computer Studies at school). One commented that some students had bad programming habits and required reteaching, but these habits were not necessarily acquired at school.
Allied to the above was the question of the difficulty for teachers of some optional topics, and the necessary retraining of teachers. Again there was a great deal of support for this statement: 46% agreed fully, 42% agreed, and 12% disagreed.

It was stated that the advanced topics (in both compulsory and optional topics) required a great deal of knowledge from the teachers. It was felt that teaching the basic principles was enough of a challenge. One stated that teachers had no hope of covering the contents properly.

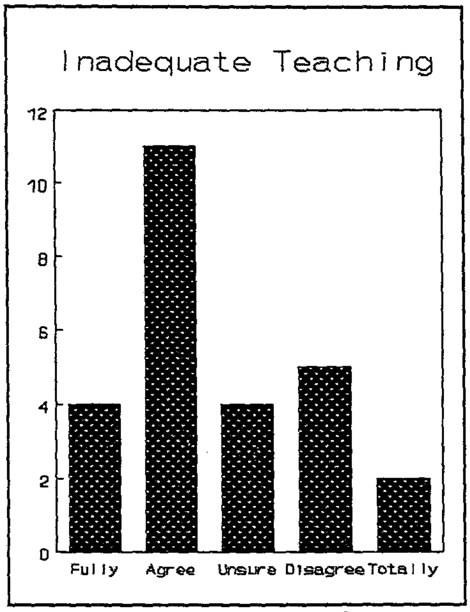

Figure 5.9: Question 9d

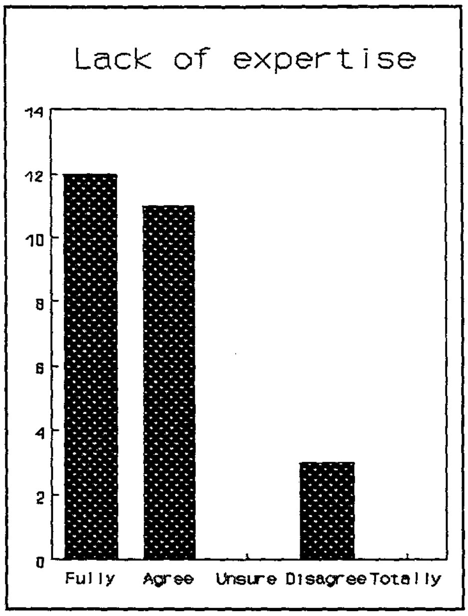

Figure 5.10:Question 9g

It was mentioned by three respondents that it was difficult to keep up with technological developments. Retraining of teachers was essential, and any 'good' teachers would be retraining themselves anyway. Two other respondents thought that any properly qualified teacher should be able to cope with this level of work.

The problem was raised that teachers with expertise could get better jobs outside the teaching profession. The solution proposed was to introduce differentiated salaries to keep such teachers.

One respondent felt that universities possibly undervalued teachers. He/she stated "Teachers are not as stupid or idle as they have been made out to be" and that schools were unnecessarily criticised: "people feel that nothing good is done at school, but this is incorrect".

5.5 Preparation for university

On the statement that the main purpose of the course was not preparation for further study at university, but as an entity in its own right, many respondents agreed: 19% agreeing fully, 46% agreeing, 23% unsure, and 12% disagreeing. Many felt that this was no problem (30% not at all, and 13% minor, whereas 52% felt it was somewhat of a problem).

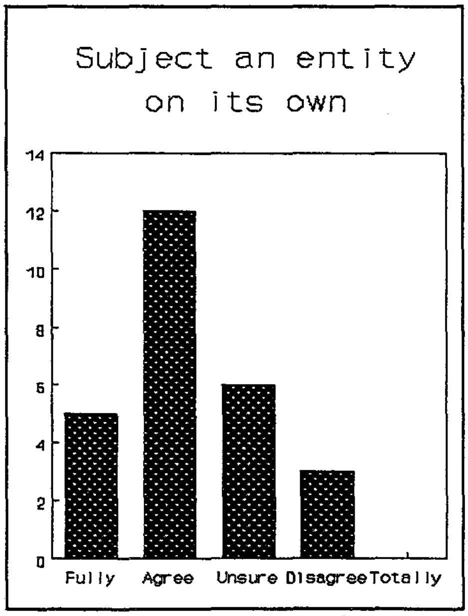

Figure 5.11:Question 9h

Some stated that the subject at school should be viewed as an entity on its own and not merely as preparation for further study at university level, and was as such no different from other subjects. Schools taught the basics. The subject at school should match the pupils' needs. It was seen to be necessary for the designers of <u>university</u> syllabuses to adapt.

Two thought that the subject did not provide the necessary background for university courses, one because the syllabus was too wide and shallow, and the other because End-user computing (EUC) was not a compulsory topic. It was also felt that the subject was too far removed from applications in business.

Two said that the subject served both the purpose of providing a general background and for preparation for university. Both purposes were thought to be necessary by some.

One respondent stated that more cooperation between school and university would solve any problem of curriculum overlap and disagreement about the contents of the Computer Studies syllabus.

5.6 Syllabus evaluation

Questioning whether the optional modules could be covered in enough depth to be of value also evoked disparate responses. 19% agreed fully with the statement that they could not, 35% agreed somewhat, 8% respondents were unsure, 27% disagreed, and 12% disagreed totally. Many felt obliged to comment on the statement.

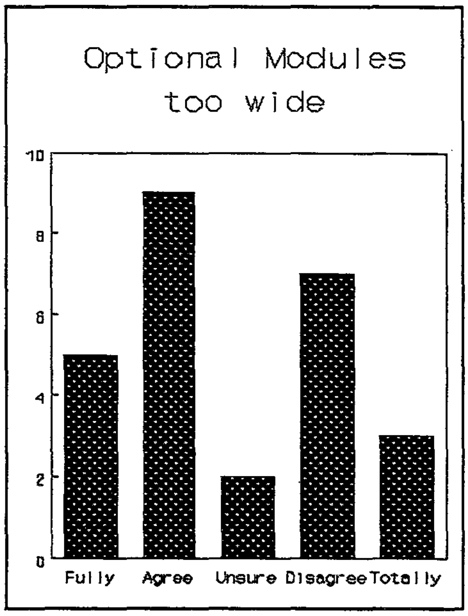

Figure 5.12:Question 9e

Of those respondents who agreed with the statement, one believed that some optional topics were more appropriate than the compulsory topics. This was echoed by another who felt that the topic on End-user computing should have been compulsory. Five others felt that the syllabus was too ambitious: it dealt unnecessarily with trendy and difficult topics (like AI): it would be better to teach basic principles. One stated that these topics could not be covered adequately. Many thought that some topics should be left to be dealt with at tertiary level, especially since there were not many teachers fully able to teach these topics. One opinion was that the advanced topics were often not well taught, and that the contents changed rapidly. Another stated that these topics were acceptable, but not enough teachers knew the field in depth. Another stated that to provide a background introduction required an excellent grasp of the subject which few teachers had an opportunity to acquire.

Of those who disagreed with the statement, three wrote that a brief review was better than none, especially if the pupils were aware of the limitations of what was taught. One stated that the compulsory topics plus choices gave a good background if correctly implemented. Another wrote that although the more advanced topics could not be explored fully, this did allow for an introduction to these topics at school level that could then be taught more easily at a later stage (the spiral approach). One lecturer asserted that the topics were suitable, if taught properly.

5.6.1 Aspects that should be changed in the school syllabus

That the syllabus emphasised programming over End-user computing was seen to be a problem by two respondents. They felt that this could have been obviated by consultation with universities. It was stated that some topics could easily be left to be taught at university level i.a. truth tables and logic circuits. It was also noted that the syllabus should concentrate more on literacy and End-user computing to complement the university courses. Another opinion voiced was that *real* Computer Studies required a logical maturity and school pupils might be too young to possess this. The opinion that there was too little emphasis at school level on general problem solving, and that problem solving was generally not well taught, was raised. Another proposed that the emphasis should be on background knowledge and not on chasing after new technology.

5.7 General comments

Some general comments were made that were not covered by the headings and statements above. One was the possibility of exemptions for Computer Studies students from some courses. Apparently this is already the case at some universities for Natal Computer Studies matriculants. It was mooted that teaching at school level should improve before this could be done generally. It was also stated that it would be a long time before Computer Studies at school would be a prerequisite for Computer Science at university.

Of interest were the comments on the trends in student numbers at university: It was found that numbers of first year Computer Science students were in general not decreasing, although no marked increase was noted overall. BIS/SEng numbers appeared to be increasing. This is not therefore as marked a trend as in the UK and other western countries (as mentioned in chapter 2).

5.8 Finally on a positive note

A respondent stated that there was no conflict between school and university - at school 'what' is taught, at universities 'why' and 'how'. Another stated that many adjustments would still be made at both levels since the subject was very new at both school and university. This should not be a cause for concern.

5.9 Limitations of the survey

The survey had a structured design but allowed for comments after each question. In retrospect some questions may have appeared ambiguous. The comment area after each question allowed for clarification in this area. Judging by the amount of comment received on many questions, it was felt that the statements and questions were accurately chosen to reflect at least some opinions and areas of concern of the target audience. The survey is thus not viewed as having major limitations. Although the majority of respondents had not seen the proposed syllabus prior to the survey, a copy was appended which was available for examination.

5.10 Conclusion

The main areas of consensus were the following: The social rationale was supported strongly in that the majority felt that Computer Literacy should be available for all, and that selection of pupils for Computer Studies was not acceptable. Computer Studies at school was seen as valuable for scholars, although changes in the approach or syllabus should possibly be made e.g. less emphasis should be placed on programming.

Lecturers experienced a great deal of difficulty in first year classes with the two streams of students. However, they felt these problems were inevitable and not necessarily a reflection of the quality of teaching at school. Reteaching of certain areas was inevitable.

Technology cost constraints were not seen as an insurmountable problem for schools that wanted to offer Computer Studies.

Some criticisms of the existing and proposed syllabuses were made. The main criticisms were that End-user Computing (the use of the computer and software packages) was underemphasised, and that the new syllabus was too wide and shallow for even the best qualified and able teachers. It was felt that there were not enough such teachers anyway. One stated that there should be more emphasis on problem solving in schools (but whether this implied that teaching problem solving was not adequate generally or whether it meant that the algorithm module should be expanded is not clear).

Many felt however, that these problems could be corrected by closer co-operation between syllabus designers and university departments.

Most endorsed the spiral approach to teaching the subject, but many felt that this was not supported by the new school syllabus.

Despite the criticisms levelled at the school subject, there were many areas of common concern voiced by both the designers of the syllabus and the university staff. The divergent and convergent issues will bè examined further in the final chapter.

Chapter 6

Conclusions and Recommendations

6.1 Synopsis of research findings.

6.1.1 Rationales.
6.1.2 The social rationale and pupil selection.
6.1.3 Costs.
6.1.4 Pupil stress.
6.1.5 Teachers and teaching at school level.
6.1.6 Teaching at university level.
6.1.7 Analysis of contents and overlap of syllabuses.
6.1.8 Comparisons with the United Kingdom.

6.2 Implications of Findings.

6.2.1 Interaction between literacy and Computer Studies.
6.2.2 Selection and drop-out rate.
6.2.3 Costs and teacher availability.
6.2.4 The content of the Computer Studies syllabus.
6.2.5 The teaching and content of the Computer Science 1 syllabuses.

6.3 Limitations of the surveys.

6.4 Conclusion.

6.1 Synopsis of research findings

The research findings relating to the main concerns voiced in the literature follow:

6.1.1 Rationales

The social, vocational and catalytic rationales were the underlying rationales which received the most support from the syllabus designers for the introduction of computer education in schools. There was a great deal of agreement amongst the syllabus designers and university computer science lecturers in support of the social rationale. The social and vocational rationales appear to be the motivation behind the "Computers in Schools and Colleges Project" (CISC Project) of the Cape Education Department:

> Education is compelled to prepare pupils for the vocational requirements resulting from these changes (caused by the advent of the PC's - writer's note). On the one hand individuals who wish to pursue a career in the computer industry should be prepared to do so. On the other hand the broader masses have to be made computer literate so that they can be prepared to use the computer as a tool in their vocation. (Cape Education Department 1991:1).

This project is part of a wider DEC computerisation project.

Both groups strongly supported wider Computer Literacy for all (thus once again supporting the social and perhaps vocational rationale). However, in the aims of the proposed Computer Studies syllabus, the vocational rationale is not reflected at all.

6.1.2 The social rationale and pupil selection

The designer group did not feel as strongly as the university educators about the selection of pupils to take Computer Studies. University representatives objected strongly to selection of pupils, seeing it as a violation of the social rationale.

6.1.3 Costs

Costs of technology were considered important, but not crippling in providing Computer Studies. Costs were perhaps more of a factor in providing Computer Literacy for all. The cost of IT was seen to be not disproportionate to the gains for the pupils. The university educators stated that it was unnecessary to "chase after new technology" (thus incurring greater costs), since the basic principles of the subject could be taught with even the so-called "useless" Apples.

6.1.4 Pupil stress

The designer group showed appreciation for the difficulties faced by pupils (and parents!) in taking the subject extra-curricularly. It was felt that it would be advantageous for the pupils if the subject could be offered as a sixth subject in schools. Competition with other subjects for a place in the curriculum was not viewed as important.

6.1.5 Teachers and teaching at school level

Of great concern to both groups was the quality, "up-to-dateness", and availability of teachers. The designers were almost unanimous that there were not enough qualified teachers, and that in-service training was an important factor. There was a qualification to this opinion: there were enough trained people, but not many wished to continue in a teaching career or wished to teach Computer studies.

The universities in general supported the statement that there was a necessity to reteach certain areas at university. Some felt that this was due to bad habits acquired elsewhere. Many stated that there was a shortage of skilled teachers. However, most of them agreed that reteaching or duplicating teaching would be necessary anyway, because of the mix of students who had and had not taken Computer Studies at schools. Some disagreed that the teaching was unsatisfactory at school level.

6.1.6 Teaching at university level

Of major concern to the universities was the problem of teaching two levels of pupils (in terms of Computer Studies knowledge) in the first year. The designer group saw this as less of a problem. This problem was ascribed to two factors:

(i) Some university lecturers felt that there were unnecessarily technical or advanced contents in the school syllabus.

(ii) Others saw the uneven mix of first year students, with the low ratio of Computer Studies students, as the main problem. This, combined with the fact that these Computer Studies students were as rule the brighter students, caused a wider disparity perhaps than in other subjects (such as Accounting, where pupils were not selected on academic merit to take the subject).

However, the majority felt that the mix of students in a class was inevitable, and that universities had to come to terms with the problem.

6.1.7 Analysis of contents and overlap of syllabuses

The aims of the proposed school syllabus as stated was

(i) to provide a computer education for pupils ending their formal computer science related education,

and

(ii) to form a base for further studies at tertiary level.

The proposed new syllabus differed from the old in two major areas:

(i) The syllabus was designed to be presented using the spiral didactic model in which topics are revisited in greater depth in succeeding years. This was an important advance in the new syllabus. The spiral approach was not as clearly defined in the existing syllabus.

(ii) Not much was removed from the old syllabus except the necessity of two programming languages. However, some areas were made optional, and expanded. A host of totally new modules were added as optional modules in the new syllabus.

The overlap in contents of the Computer Studies syllabus and university first year syllabuses exists:

(i) The main overlap firstly occurs in the use of Pascal. (Although this was no longer prescribed, it is assumed that it will be the language proposed by individual provinces).

(ii) Development of algorithms was another area of overlap.

(iii) Number representation, data types and data structures, searching and sorting techniques, computer architecture and communications; and to a lesser extent the history of the rise of computers also overlapped. So did the System Development Life Cycle, the modern topics and computer logic.

(iv) The overlap appeared to be more in the compulsory than optional modules although logic circuits and the SDLC were amongst the optional modules.

6.1.8 Comparisons with the United Kingdom

S.A. has gained from a more unitary or standardised approach to the school syllabus, which has only recently been enforced in the U.K.. Because of the syllabus standardisation, and the somewhat later development of the subject, many of the problems experienced from the lack of standardisation in IT have been avoided in S.A.. However, there has historically been a closer and more direct connection between schools and universities in the United Kingdom, which has been beneficial.
For example, the problem of overlapping course material does not appear to have arisen in the U.K. The role of the U.K. universities in setting the examinations in the past is probably the reason for this.

6.2 Implications of findings

The following issues need to be addressed:

a) The interaction between Computer Literacy and Computer Studies.

b) The selection of pupils for Computer Studies and the extra-curricular nature of the subject.

c) Availability and quality of teaching, and cost of IT.

d) The content of the Computer Studies syllabus.

e) The teaching and content of Computer Science 1 syllabuses.

6.2.1 Interaction between Computer Literacy and Computer Studies

If the social and vocational rationales are the main rationales supporting computer education in schools, then the natural progression must be to open the subject up to all pupils. This appears to be the direction already taken in Natal with the offering of Computer Literacy i.e. wordprocessing and spreadsheets at Standard 6 and 7 level. The Cape Education Department also appears to be addressing this with their Computers in Schools and Colleges (CISC) Project.

The ideal would be to offer a compulsory Computer Literacy course to all pupils in standards 6 and 7. All pupils should be computer literate to the extent that they would be able to use the common end-user packages available such as wordprocessing, spreadsheets and file handling packages. This course could be at a level slightly lower than technical college N5 courses. From an almost purely practical course, they would gain a background theoretical and working knowledge of computers which would be sufficient for use in the workplace. This would also allow for a natural progression to a more cross-curricular approach in the use of computers in schools. "Databases as well as word processors and spreadsheets are general software purpose tools that, although not specifically developed for instructional use, are likely to offer remarkable opportunities in the field of education. This last statement is generally supported by various considerations: . . . -the possibility of using these tools to conduct wide-ranging school projects based on large quantities of data." (Degl'Innocenti & Ferraris 1988:157). Anderson also emphasises the use of computers in information handling (Anderson 1991).

With this level of knowledge assured, the more specialised subject of Computer Studies can then be considered afresh.

6.2.2 Selection and drop-out rate

The selection process for Computer Studies pupils should fall away. This, combined with compulsory Computer Literacy at the lower levels, would cause the drop out rate in Computer Studies to diminish since pupils would have a better idea of what such a course contained, and would not take the course merely for the glory of being chosen. If the subject is offered as a normal choice for pupils at schools, as a sixth (or seventh subject), the problem would be further reduced.

Concerning the case of pupils wishing to become technically more proficient at programming rather than wanting to study the more theoretical Computer Studies course, a possible solution could be to offer an entirely different course perhaps extra-curricularly. Such a course could also perhaps tie in with the proposed new thrust into "Design and Technology" which individual Cape schools are proposing. This is an area which merits further research.

6.2.3 Costs and teacher availability

Costs do not appear to be viewed as a great limiting factor in teaching Computer Studies at schools, which is somewhat surprising. To offer Computer Literacy to a wider range of scholars will have cost implications which need to be examined further.

Whether there are enough teachers qualified to teach Computer Studies is a matter which again merits further research. There appear to be divergent opinions on this. One opinion is that it is an extremely demanding subject to teach. However, there exists unanimity that there are not enough qualified people <u>prepared</u> to teach Computer Studies.

With the present economic recession in place, there are many qualified skilled people out of jobs in the dataprocessing area. In addition, there are many university graduates including Computer Science majors who are not able to get jobs: most jobs require applicants to have 2 years' working experience. It is vital that attractive packages are devised immediately to channel these people into education and specifically computer education. This could be done by creating flexible means of attaining the necessary educational qualifications. If the job market improves, they may leave education, but many may remain.

6.2.4 The content of the Computer Studies syllabus

The difficulties inherent in drawing up Computer Studies and Computer Science (Business Information Systems etc.) syllabuses are similar since they are based on fast changing technology and have a large practical or training component.

There is doubt whether the new school syllabus can succeed in its stated aim of serving as a foundation course for tertiary level. It would appear from the findings that:

(i) There is a perceived element of inadequate teaching or incorrect emphasis at school level.

(ii) The problem of teaching students, bored by repetition of school work at university, is a real one.

(iii) The overlap in the school and first year university syllabuses exists.

The view held by the researcher is that the proposed syllabus has strengths: it covers the basic principles; it approaches the syllabus in a spiral manner; and it attempts to cater for inflexibility of syllabus design (because of the time-span involved) to adapt to changing technology by adding in open-ended optional modules.

However, what of the contents and emphasis of what is taught?

The introduction of Computer Literacy would firstly alleviate the call for more End User Computing (i.e. the use of packages) in the Computer Studies syllabus.

The basic principles need to be taught at both school and university levels. It is unavoidable that there is a measure of overlap at university since many students have not been exposed to Computer Studies at school. Computer architecture, algorithm design and, with programming as part of the syllabus, Pascal, all need to be taught at both levels.

However, it is felt (by the researcher) that a weakness in the new syllabus is that, in order to be flexible, it has tried to do too much. The teaching skills could well exist for the advanced topics, but the optional modules appear to have been added in a "bottom up" manner. The syllabus (again in the opinion of the researcher), has too technical a slant.

It is suggested that much can be learnt from the U.K. approach, particularly from the L.E.A.G. GCSE syllabus. This also advocates a spiral approach to the subject, but uses an application based method. The contents of the U.K. courses appear to be based on a more holistic or systems approach. They appear less fragmented but perhaps less advanced than the S.A. equivalent. It is possible that in terms of IT, the S.A. industry has been more influenced by the U.S.A in their adoption of IT. It would be instructive to compare what is being offered at 12th grade in Computer Studies at American schools. However, the standards in that vast country also vary widely. This could be a further avenue to examine.

In the U.K. syllabuses, many of the more technical features of the S.A. syllabus are ignored. All the theory is approached through applications. There is less stress on programming, and more on using the computer as a problem-solving tool. The application-based approach forms a unifying thread as well as giving students the necessary concrete-to-abstract bridge. While the basic contents are similar to that of the compulsory modules in the S.A. syllabus, the contents are less technical. The writer feels that the U.K. syllabus could have been expanded, still using the systems approach, to include more modern topics some of which are found in the S.A. syllabus.

The university lecturers were strongly in favour of the subject being offered at school level, and stated that it was to the advantage of pupils to take the course at school level. They also stated that it would provide the pupils with a better idea of what Computer Science at university would comprise and therefore allow them to make a more informed choice in when choosing to take the subject. (Computer Literacy would fulfil the same function for pupils opting for Computer Studies).

6.2.5 The teaching and content of the Computer Science 1 syllabuses

From the comments in the surveys, it is felt that the problems with teaching at university level could be improved in three ways:

Firstly, there does appear to be a great need for more communication between school and universities. Where the onus lies in initiating this contact is not clear. It is recommended that as for Accountancy "university educators develop a close relationship with high school educators so that they may better understand the high school course and contribute towards its development." (Rowlands 1988:264).

Secondly, and this may flow from the above point, the content at school level could be made somewhat less technical. The main problem appears to lie in the more advanced topics, which may be too wide and shallow to add sufficiently to student knowledge.

Thirdly, the problem of dealing with a mix of students could be eased by using the following approaches (most of these solutions are suggestions from the survey):

(i) By offering a compulsory bridging course to all non-Computer Studies students.

(ii) By giving partial credits for Computer Studies matriculants.

(iii) By providing more options in first year, as is done by English universities.

(iv) By running two courses, if numbers permit, for the different groups.

6.3 Limitations of the surveys

The above findings are related to DEC schools only, and there is the larger issue of technology in <u>all</u> schools in South Africa, which merits further investigation.

6.4 Conclusion

In the opinion of the researcher, a Computer Studies syllabus which is based on applications has even greater possibilities of flexibilty and would provide a better vehicle for moving from concrete operational thinking to formal operational thinking, relevant (as proposed by Griffin) to pupils in standards 8 to 10. It is also felt that some topics can be left happily for universities to handle without decreasing the benefits of the subject at school level to pupils who may not pursue it any further. The application approach may have further significance when considering syllabuses for DET schools, for example. Such an approach allows for flexibility, in that applications may be chosen to reflect the, at present, differing life experiences of the pupils.

However, the main area for improvement would be in better communications between schools and/or syllabus designers and university educators. This of course implies greater involvement and time commitment, but should prove fruitful to all. Better communication would foster greater understanding of the particular viewpoints, opinions, and problems of all the participants in computer education in South Africa.

REFERENCES

Anderson, J.S.A. (1991). Information Technology - A cross-curricular competence for all pupils. Computers and Education, 16(1), 23-27.

Barron, D. (1990). Thoroughly modern misses. The Times Higher Education Supplement, November, 1.

Cape Education Department. (1991). The C.I.S.C. project (computers in schools and colleges).

Cathcart, W.G. (1990). Effects of Logo instruction on cognitive style. Journal of Educational Computing Research, 6(2), 231-242.

Chen, M. (1985). A macro-focus on microcomputers. In M. Chen & W. Paisley (Eds.), Children and microcomputers (pp. 37-58). Sage Publications: Beverly Hills.

Churchill, G.A. Jnr. (1983). Marketing Research: Methodological Foundations (Third Edition). Dryden Press.

Cockcroft, W. (1982). Mathematics counts: report of the committee of inquiry into the teaching of mathematics in schools under the chairmanship of W.H. Cockcroft, (pp.116-120). London: Her Majesty's Stationery Office.

Computer Science at Warwick (1990-1991). First year student handbook. University of Warwick.

Davis, L. (1989). Can education meet IS career demands?. Datamation, March 15, 65-69.

Dean, J., Briggs, J.H. & Nichol, J. (1987). Why Prolog? In E. Scanlon & T. O'Shea (Eds.), Educational Computing (pp. 161-176). John Wiley & Sons.

Degl'Innocenti, R. & Ferraris, M. (1988). Database as a tool for promoting research activities in the classroom: An example in teaching humanities. Computers and Education, 12(1), 157-162.

Department of Education. (1985). Senior secondary course syllabus for computer studies standard grade. Provincial Administration of the Cape of Good Hope.

Department of Education. (1985). Senior secondary course syllabus for computer studies higher grade. Provincial Administration of the Cape of Good Hope.

Department of Education and Science: Welsh Office. (1985). The National Criteria: General Criteria. London: Her Majesty's Stationery Office.

Department of Education and Culture: House of Assembly. (1991). Draft core syllabus for computer studies higher grade standards 8, 9 and 10. House of Assembly.

Department of Education and Science. (1989). Information technology from 5 to 16: Curriculum matters 15. London: Her Majesty's Stationery Office.

Dodds, R. (July 1991). Lecturer, Stellenbosch University. Personal communication.

Downes, T. (1990). Tomorrow's challenge for computers in education: Putting practice into theory. Education, 110(4), 430-435.

Durndell, A., Macleod H. & Siann G. (1987). A survey of attitudes to, knowledge about and experience of computers. Computers and Education, 11(3), 167-175.

Education Authorities Directory (U.K.).

Education Broadsheet March 1990 Information Note 10 (pp. 1-2). (obtained from the British Council).

Eisner, E.W.,(1982). Cognition and the curriculum; a basis for deciding what to teach. Longman.

Evans, N. (1986). The future of the microcomputer in schools. London: Macmillan.

Fallick, P.M., & Pistorius, M.C. (1990). A review of the use of computers in education in South Africa. Part I: primary and high schools. South African Computer Journal, 2, 53-58.

Forcheri, P., Furinghetti, F. & Molfino, M.T. (1990). Integration of computer science and mathematics in upper secondary school: Reflections and realisations. Computers and Education, 14(4), 325-333.

Fritz, J.M. (1985). Rethinking computer literacy. In K. Duncan & D. Harris (Eds.), Computers in Education (pp. 705-710). North Holland.

Gay, L.R. (1987). Educational Research Competencies for Analysis and Application (Third Edition). Merrill Publishing Company.

General Certificate of Secondary Education. Computer Studies. Syllabus for the 1992 and 1993 Examinations (JMB). Northern Examining Association, 1-16.

Griffin, J. (1988). Computer studies for the general certificate of secondary education: teaching through applications. Computers and Education, 12, 443-451.

Gwyn, R.(1987). Information technology and education: the approach to policy in England, Wales and Northern Ireland. In E. Scanlon & T. O'Shea (Eds.), Educational Computing (pp. 63-72). John Wiley & Sons.

Hawkridge, D. (1987). Experience in using computers for learning. In E. Scanlon & T. O'Shea (Eds.), Educational Computing (pp. 5-33). John Wiley & Sons.

Hawkridge, D. (1990). Who needs computers in schools and why? Computers and Education, 15(1), 1-6.

Hennesey, K. (1983). In J. Maddison, Education in the microelectronics era (p.76). Open University Press: Milton Keynes.

Hennessy, A.K. & Croft, G. (1980). Carefully through a minefield. The Times Educational Supplement, (07/03/80), 33-35.

Heunis, M. (07/90). Co-ordinator of Computer Studies, Teachers Centre, East London. Personal communication.

Hodson, D. (1990). A computer-based education in New Zealand: a time of transition and uncertainty. Journal of Curriculum Studies, 22(1), 57-76.

Inquiry into R4,9m DET computer deal. (1988, June 15). Daily Dispatch.

Johnson, D.C. & Anderson, R.E. (1985). Algorithms and programming as part of computer literacy - some assessment results with implications for secondary school mathematics. In K. Duncan & D. Harris (Eds.), Computers in Education (pp. 691-698). North Holland.

Kennett, K.F. (1990). Computers in education: Rear-vision thinking or golden opportunities for the future. Education, 110(4), 403-409.

Kennewell, S., Fox, P., Mitton, C. & Selwood, I. (1989). Computer studies through applications. Oxford University Press.

Kennewell, S., Fox, P., Mitton, C. & Selwood, I. (1989). Computer studies through applications. Teacher's resource pack. Oxford University Press.

King, J. (1986). Computer Systems, (from a cutting: date and page unknown).

King's College London. Department of Computing. Undergraduate Courses for 1990-1991.

Lees, B. (1981). Computer Education in secondary schools: A framework for the future. Cambridge Journal of Education, 11(1), 36-45.

Leith, P. (1990). Computer fall-out. The Times Higher Educational Supplement, (21/12/90).

Lieberman, D. (1985). Research on children and microcomputers. In M. Chen & W. Paisley (Eds.), Children and microcomputers (pp. 59-76). Sage Publications: Beverly Hills Publisher.

London East Anglian Group. (1990). General Certificate of Secondary Education. Syllabuses: Computer Studies. LEAG for GCSE Examinations, Summer, 1-16.

Longworth, N. (1981). We're moving into the information society. What shall we teach the children? Computer Education, June, 17-19.

Longworth, N. (1982). Today's development for tomorrow's world - educating for the information society. Computer Education, June, 20-21.

Lorton, P. & Muscat, E. (1981). Preparing for the real world. Educational Computing, December, 23-24.

MacLachlan, B. D. (1991). Subject Head, Hilton College, Natal. Personal communication.

Masterton, D. (1982). What's cooking at Chelsea?. Educational Computing, July/August, 22.

McGee, P. (1986). Computer Studies at GCSE - The applications approach. Computer Education, November, 5-6.

McGee, P. (1980). The content and assessment of computer studies. Aspects of Education, 23, 69-76.

Mead, T. (1988). "Computer Science education is very much your business". Datamation, December 15, 82.

Neal, L.R. (1986). GCSE and computer studies. Computer Education, November, 2-3.

North, R. (1991). Managing the integration of information technology across the curriculum of the secondary school. Computers and Education, 16(1), 13-16.

Pea, R.D., Kurland, D.M., & Hawkins J. (1985). LOGO and the development of thinking skills. In M. Chen & W. Paisley (Eds.), Children and microcomputers (pp. 193-212). Sage Publications: Beverly Hills.

Pea, R.D. (1987). LOGO programming and problem solving. In E. Scanlon & T. O'Shea (Eds.), Educational Computing (pp.155-160). John Wiley & Sons.

Piddock, P. (1987). Computer science in schools. Educational Review, 39(2), 163-168.

Rautenbach, W.L. (1979). A report on a visit to Israel to study professional and technical education. Unpublished manuscript.

Rautenbach, W.L. (1980). A study of technical and vocational training in Taiwan. Unpublished manuscript.

Reeves, G.R. & Bussom, R.S. (1979). Information Systems Curriculum. Journal of Systems Management, March, 18-21.

Rich, B. (1986). A-level computer science. Computer Education, June, 16-19.

Riordan, D. (20/10/91). Professor in Computer Science, Technical University of Nova Scotia. Personal communication.

Roets, P. (09/09/91). Manager, ISM, East London Branch. Personal communication.

Ross, G. (14/06/90). Subject head in Computer Studies, Hillcrest High School, Natal. Personal communication.

Rowlands, J.E. (1988). Factors influencing secondary school pupils in their choice of a career. Unpublished master's thesis. Rhodes University, Grahamstown, Eastern Cape.

Ruthven, K. (1981). Does computer studies have a future?. Educational Computing, September, 5.

Schiffmann, S. (1985). Designing a computer literacy curriculum: an integrated approach. In K. Duncan & D. Harris (Eds.), Computers in Education (pp. 677-680). North Holland.

Selwood, I. (1988). The rise and fall of computer studies (A subject with a death wish). Computer Education, June, 2-3.

Skok, W. (1988). Degree Courses: From Computer Science to Business Information Technology. Computer Education, June, 27-29.

Steiner, H.G. (1991). Does today's industry need qualified computer scientists? Viewpoint I. South African Computer Journal, 4, 1-3.

Stonier, T. (1979). Recurrent education and lifelong learning. In Schuller T. & Megerry J. (Eds), Learning World Yearbook of Education (pp. 31-34). Kogan Page: London.

Terry, P. (18/06/90). Professor of Computer Science, Rhodes University, Grahamstown. Personal communication.

Tucker M.S. (1985). Computers in schools: What revolution? Journal of Communication, 35(4), 12-23.

Van Niekerk A. (20/09/91). Pupil in Standard 8 taking C.S. as a 7th subject. Personal communication.

Visser, P. (1991). Does today's industry need qualified computer scientists? Viewpoint II. South African Computer Journal, 4, 1-3.

Wellington, J.J. (1990). Computer education and the curriculum. Journal of Curriculum Studies, 22(1). 57-76.

Wheals R. (03/03/90). Tutor in Business Information Systems, Rhodes University, East London. Personal communication.

Wilkens H.J. (1990). DEC Directorate of Information. Personal communication.

Wood, D. (1988). How children think & learn. In J. Dunn (Ed.), Understanding children's worlds (pp. 148-227). Oxford: Basil Blackwell.

APPENDICES

Appendix A: U.K. Examining groups and developments

Table 1 - Examining Authorities 1985 - GCE
(Source: Department of Education and Science:Welsh Office)

1. Oxford & Cambridge Examination Board
2. University of Oxford Delegacy of Local Examinations
3. Cambridge University Local Examinations Syndicate
4. University of London
5. JMB
6. Associated Examining Board
7. Welsh Joint Education Committee

Table 2 - GCSE Groups

GCSE (comprising GCE O level, CSE & joint O/GCE level examinations)

Groups Comprising

1. London East Anglia
   The East Anglian Regional (L.E.A.G) Examining Boards
   University of London Regional Examining Board
   London Regional Examining Boards

2. Midland Examining group
   East Midland Regional Examining Board
   Oxford & Cambridge Schools Examining Board
   Southern Universities Joint Board
   The West Midlands Examination Board and Cambridge University Local Examinations Syndicate.

3. Northern Examining Assoc.
   Associated Lancashire Schools Examining Board
   JMB
   Northern Regional Examinations Board
   North West Regional Examinations Board
   Yorkshire and Humberside Regional Examinations Board.

4. Southern Examining Group
   The Associated Examining Board
   Oxford Schools Examinations Board.

5. The Welsh Joint Education Committee.
(Education Authorities Directory U.K.)

Table 3 - National Development in Britain
(Source: Broadsheet: British Council 1990)

1972:	National Development Programme for Computer Assisted Learning (aimed at tertiary level).
1978:	Advisory Unit for Computer Based Education (Hertfordshire - first Local Education Authority (LEA) advisory service for Computer education).
1980:	Start of Micro Electronic Programme (MEP) providing a service to the LEA's.
1981:	DTI 'Micros in Schools' project.
1986:	End of MEP.
1987:	Microelectronics Education Support Unit (MESU) in England Wales and Northern Ireland.
1988:	600 Advisory teachers supported by a grant to help classroom teachers in schools use IT more effectively.
1988:	MESU and Council for Educational Technology (CET) merge to form the National Council for Educational Technology (NCET).

Appendix B: Syllabuses

1. Syllabus for Computer Studies Syllabus Higher Grade 1985.
2. Draft Core Syllabus for Computer Studies Higher Grade 1991.

PROVINCIAL ADMINISTRATION OF THE CAPE OF GOOD HOPE

DEPARTMENT OF EDUCATION

SENIOR SECONDARY COURSE

SYLLABUS

FOR

COMPUTER STUDIES

HIGHER GRADE

1985

SENIOR SECONDARY COURSE : SYLLABUS FOR COMPUTER STUDIES HIGHER GRADE

The following syllabus for Computer Studies Higher Grade for the Senior Secondary Course will be introduced as from 1 January 1985.

The syllabus will be introduced in Standard 8 in 1985 and the first Senior Certificate Examination in this subject will be held in November/December 1987.

SENIOR SECONDARY COURSE : SYLLABUS FOR COMPUTER STUDIES HIGHER GRADE

1. INTRODUCTION

The syllabus is designed and is being implemented so as to lead the competent pupil into the more academic areas of computing. The emphasis will be on the more theoretical aspects of computing which will be achieved by looking at each aspect of the syllabus in greater detail than in the standard grade syllabus. The practical work will lead the pupil to a greater understanding of the theoretical component of the syllabus.

2. AIMS

2.1 General Aims

2.1.1 To study not only the computer itself, but to study theory and methods by which a variety of problems can be solved with the aid of the computer.

2.1.2 To widen the pupils' horizons and insight so as to make them conscious of the scope and limitations of the computer.

2.1.3 To held the pupils develop a frame of reference by means of which they can reliably, effectively and usefully apply this knowledge to interpret data in practice.

2.1.4 To give the pupils a basic training in the methods of data collection and processing as well as in the provision of information in a variety of fields.

2.1.5 To lead the pupil, by means of appropriate teaching, to an attitude towards problem solving which stretches beyond the scope of the computer situation.

2.1.6 To develop the pupil's critical and logical faculties of thought.

2.1.7 To develop the pupil's skill in the manipulation of symbols and logical structures.

2.2 Particular Aims

2.2.1 To increase the pupils' skill in analysing problems into fundamental steps and to increase their skill in the synthesis of solutions.

2.2.2 To develop the pupil's skill in developing an effective algorithm for the solution of a problem.

2.2.3/...

2.2.3 To verify an algorithm.

2.2.4 To write and debug a computer program for an algorithm.

2.2.5 To design the computer output so that it can be easily read and understood.

2.2.6 To study suitable and relevant problems with the aim of developing the pupil's ability to solve them.

2.2.7 To understand, in general outline, the functioning of the computer.

2.2.8 To consider the socio-economic and other implications of the use of computers and to make pupils aware of the indispensable rôle played by computers in society.

2.2.9 To foster an open mind towards possible computer applications and their feasibility.

2.2.10 To stimulate interest in further computer study.

2.2.11 To introduce pupils to software packages.

3. EXAMINING

3.1 Standards 8 and 9 will be examined internally.

3.2 The external examination in standard 10 will comprise the following:

3.2.1 Written Paper : one three hour paper will be set on the syllabus for Std. 10 and it will count 280 marks. Concepts studied in Stds 8 and 9 may also be included in this paper.

3.2.2 Practical Work : the balance of the final mark, viz. 120 marks, will be derived from practical projects. The set of practical projects submitted may be chosen from any one of the following combinations:

* 2 major assignments (60 marks each) OR
* 1 major assignment (60 marks) PLUS any combination of the following
  ** medium assignment (40 marks),
  ** minor assignment (20 marks), or
  ** micro assignments (10 marks each),

  that totals 60 marks, with a maximum of 2 micro assignments.

Each assignment must be marked by the teacher and the assignment, as well as the marks awarded, must be submitted to the examiner for moderation.

3.2.3 The major assignments should be programming projects OR any other project directly related to Computer Studies. In the latter case, the prior permission of the Department should be obtained. Application for such permission must reach the Department before 31 October of the candidate's Std. 9 year.

3.3 It will not be necessary for candidates to memorise codes or the exact syntax of each programming language studied. Where necessary, an approved set of ASCII codes and language syntax aids will be provided.

4. TEACHING SEQUENCE

Teachers may rearrange the syllabus content for any given year (standard) according to their own preferred teaching style. The order in which topics are presented must, however, make for a logical development of the subject matter.

Comments

Advice on a possible teaching sequence will be found in the Teachers' Guide.

5. HIGH LEVEL PROGRAMMING LANGUAGES

So as to create a more flexible situation the programming languages and syntax of each language will be published in The Education Gazette.

Comments

The language subset to be examined will be published in The Education Gazette. This will be both a guide to what is to be taught and what is to be examined in the Std. 10 examination.

THE SYLLABUS

	CONTENTS		COMMENTS
	STANDARD 8		
	The topics in this year of study are not intended to be introduced serially but should be integrated with each other. Whenever necessary the historical aspects of the topic under consideration should be alluded to.		Sections 2, 5 and 10 should run concurrently.
1.	A short review of the development of the computer and of data processing	1.	Must not be treated as an entity on its own but must be integrated with the relevant sections in the remainder of the syllabus.

CONTENTS		COMMENTS	
2.	The development of algorithms		
2.1	Algorithmic structures		
2.1.1	Sequential groups		
2.1.2	The binary selection construction		
2.1.3	Loop control		
	(a) Structure with a condition at the start of a loop		
	(b) Structure with a condition at the end of a loop		
2.2	Problem analysis	2.2	Great emphasis must be laid on this aspect. It is suggested that the programming language Logo be used as the vehicle for the introduction of this topic.
2.2.1	Specification		
2.2.2	Hierarchical approach (top-down approach)		
2.2.3	Stepwise refinement		
2.2.4	Modular design		
2.3	The representation by means of a specification language or flowcharts. Trace tables to locate errors in or verify the correctness of algorithms	2.3	The subset of the ANSI X3.5-1970 specifications for flowcharting symbols should be used. If a specification language (PDL) is used it should not be complicated. Pseudo-code should be used with caution.
3.	Elementary computer architecture		To be handled at the 'Black-Box' level
3.1	Simple model of a computer		
3.2	Memory, central processor, input and output units		

CONTENTS		COMMENTS	
4.	Data Representation	4.	The emphasis must not be on number conversions but rather on the explanation of why some answers resulting from calculations are incorrect and how to rectify these problems. For example: * why does the square root of 16 equal 3,99999 when determined using an iterative procedure? * calculate the factorials of large integer values. Why are some of the answers incorrect?
4.1	Decimal and binary number systems Conversion of positive integers		
4.2	Addition in binary and overflow errors		
4.3	The internal representation of data (positive integers and characters) in the computer * The bit, byte and word		
5.	Implementation of algorithms	5.	The need for modular, structured program design must be stressed.
5.1	Principles of programming * General readibility * Meaningful variable names * Indentation * Documentation		
5.2	Introduction to the first high level programming language		
5.3	Debugging * Test data * Trace tables * Structured walkthroughs/ team debugging		
6.	Introduction to elementary logical and relational operators	6.	Must be integrated with 5.2. Only the AND, OR and NOT operators should be used - do not use De Morgan's Laws

CONTENTS		COMMENTS	
7.	Introduction to operating systems and systems software * Compilers, interpreters, editors	7.	Brief explanation of the features which a pupil will require to use the system in a meaningful way in the practical work.
8.	Introduction to applications packages * Word-processing	8.	Other packages can also be used, time permitting.
9.	Social implications * Pupils must be made aware of the role of computers in society	9.	Integrate with the relevant sections in the other parts of the syllabus.
10.	Programming assignments Problems in the school situation which are of interest and lie within the capabilities of the pupils must be tackled in ascending order of complexity	10.	An opportunity exists to introduce new topics e.g. elementary statistics, elementary calculus, simulations, graphics, etc., but the emphasis must be on algorithmic development and not on teaching the underlying theory of the new topics.
	STANDARD 9		
	The topics in this year of study are not intended to be introduced serially but should be integrated with each other. Whenever necessary the historical aspects of the topic under consideration should be alluded to.		
1.	Continuation of the study of algorithmic structures		
1.1	Evaluation of structure and efficiency		
1.2	Modular design and sub-algorithms		
2.	Further aspects of computer architecture		
2.1	The control cycle	2.1	Fetch/execute cycle
2.2	Physical properties of storage units		
2.2.1.	Disk packs, diskettes, magnetic tapes and cassettes		

CONTENTS		COMMENTS	
2.2.2	Records and blocks		
3.	Data: Types, Representation and Structures	3.	Emphasis should be placed on sections 3.2 and 3.4
3.1	Fixed point numbers * Conversion of integers between radices 2, 10 and 16 * Complementation		
3.2	Floating point numbers * Floating point arithmetic and errors * Accuracy		
3.3	Alphanumeric data * Refer to BCD, EBCDIC and ASCII		
3.4	Introduction to fields, records and files		
3.5	Introduction to the logical data type * Expressions, De Morgan's Laws * Truth tables	3.5	Extensions to the Std 8 syllabus (6) to include logical expressions and truth tables. The use of De Morgan's Laws in negation should be encouraged.
4.	Continuation of programming in the first high level language * Introduction to subprograms		
5.	Continuation of problem analysis and algorithmic design	5.	Integrate with sections 4, 6 and 9
6.	Introduction to a second high level language	6.	The second programming language should be introduced at a time which suits both the teacher and the class being taught.
7.	Use of applications packages * Word-processing; spread-sheets * Introduction to data base systems		

CONTENTS		COMMENTS	
8.	Social implications Pupils must be made aware of the implications that computers will have on society at large	8.	Integrate with the relevant sections of the other parts of the syllabus.
9.	Programming assignments Problems in the school situation which are of interest and lie within the capabilities of the pupils must be tackled in ascending order of complexity	9.	Pupils must be encouraged to write programs that can be of use in the school situation. They should also be urged to do programming assignments in both of the prescribed programming languages. The pupil is not restrained in the choice of the language used for the programming assignments; it should, however, be borne in mind that the examination will be set on the prescribed programming languages and that he should get as much practice as possible in the use of these languages.
	STANDARD 10 The topics in this year of study are not intended to be introduced serially but should be integrated with each other. Whenever necessary the historical aspects of the topic under consideration should be alluded to.		
1.	Data types and data representation continued	1.	A continuation of topics introduced in Std 8 (6) and Std 9 (3)
1.1	The logical data type * Constants and variables * Operators and expressions * Truth tables		
1.2	Application in programming techniques		
1.3	Sequential, random access and indexed files		
1.4	An introduction to data bases	1.4	This is meant to be at a superficial level. This should be integrated with the section on applications packages.

CONTENTS		COMMENTS	
1.5	Introduction to data structures * Arrays, lists, stacks, queues and trees		
2.	Continuation of the first high level programming language		
2.1	Subprograms	2.1	Can quite comfortably be introduced during the Std 9 year in section 4
2.1.1	Functions		
2.1.2	Parameters; global and local variables		
2.1.3	Procedures		
2.2	Elementary file handling	2.2	Pupils must write at least one programming project using file handling techniques.
3.	Continuation of the second high level language		
4.	Systems Software		
4.1	The operating system * Functions * Job control * Multiprogramming, multiprocessing, time-sharing, real-time systems, batch processing and virtual memory * Input and output control	4.1	A case study of a readily accessible operating system (e.g. MS-DOS) should be considered. An overview is required.
4.2	Utilities / utility programs	4.2	Concepts only are required.
4.3	Assemblers, compilers and interpreters	4.3	Concepts only are required.
4.4	Data communication	4.4	Concepts only are required.
4.5	Fourth generation software	4.5	Integrate with the section on applications packages.

CONTENTS		COMMENTS	
5.	Socio-economic and other implications of computer usage Briefly discuss a few implications such as the replacement of cash, the application of computers to law enforcement, a national population register, a data bank, computer assisted learning and instruction, school administration and management information systems, word-processing, artificial intelligence, robotics, etc.		
6.	Continuation of computer architecture		
6.1	Microcomputer hardware	6.1	The 3-bus structure (control, address, data) as well as the control cycle should be discussed.
6.2	Data communications		
7.	Continuation of problem analysis and algorithmic design		
8.	Applications packages continued		
9.	Selected programming tasks Pupils are expected to produce well documented projects which include the following requirements: * Clear statement of problem * Effective and well-constructed algorithms and programs * Documented * Self-explanatory output * Errors eliminated by the processing of sufficient test data	9.	Pupils must be encouraged to write programs that can be of use in the school situation. They should also be urged to do programming assignments in both official programming languages. The pupil is not restrained in the choice of the language used for the programming assignments; it should, however, be borne in mind that the examination will be set on the official languages and that he should get as much practice as possible in the use of these languages

DEPARTMENT OF EDUCATION AND CULTURE : HOUSE OF ASSEMBLY

DRAFT CORE SYLLABUS

FOR

COMPUTER STUDIES

HIGHER GRADE

STANDARDS 8, 9 AND 10

IMPLEMENTATION DATES:

STANDARD 8 : 1993
STANDARD 9 : 1994
STANDARD 10 : 1995

Rev: 4 March 1991

(As amended at the Joint Syllabus Committee Meeting of 7 February 1991)

1. INTRODUCTION AND GENERAL AIMS

1.1 INTRODUCTION

1.1.1 The syllabus has been designed so as to introduce pupils to the fundamentals of Computer Science which will provide them with

(a) the foundations upon which to build further studies at a tertiary level, or

(b) valuable insights into the computer field should they not continue with studies in Computer Science or similar disciplines.

1.1.2 The emphasis will be on the more theoretical aspects of computing. This will be achieved by studying each module in greater detail than in the Standard Grade syllabus.

1.1.3 The practical work, which forms an integral part of the course, will lead the pupil to a greater understanding of the theoretical components of the syllabus.

1.1.4 The compulsory modules are considered to form the core upon which the optional modules can be built.

1.1.5 Care must be taken when selecting optional modules

(a) not to choose modules which will over-emphasise the theoretical aspects of the course to the detriment of practical work

(b) to choose modules which will give an overall balance between more technical and less technical modules, and

(c) to choose modules which will broaden the foundation upon which future studies can be built and consequently expose the pupil to the latest trends in computing.

1.1.6 The syllabus has been designed so as to be presented using the spiral didactic model in which topics are introduced in the Std 8 year and then revisited, in greater detail, in the Std 9 and Std 10 years.

1.1.7 To obtain maximum benefit from the course the modules should not be presented as independent modules but should rather be integrated with one another. This means that Modules 1, 2, 3 and 4, for example, could run in parallel.

2. MODULES

2.1 MODULE 1: **Development of Algorithms** - COMPULSORY -

2.1.1 Objectives

(a) To lay a solid foundation in the development of sound problem solving strategies that can be put to good use in writing well-structured computer programs

(b) To increase the pupils' skills in analysing problems into fundamental steps and in the synthesis of a solution

(c) To develop the pupils' skill in developing effective and efficient algorithms for the solution of problems

(d) To verify an algorithm for a given problem using applicable debugging techniques

2.1.2 Content

(a) Algorithmic structures such as

(i) sequential groups

(ii) n-way selection

(iii) loop control

(iv) recursion

(b) Problem analysis

(c) Physical design using techniques such as

(i) hierarchical approach (top-down design)

(ii) modular design

(d) Representation of algorithms using techniques such as program description languages (PDL), flowcharts (ANSI or CHAPIN), structured natural language (SNL) or pseudocode

(e) Debugging of solutions using appropriate techniques such as trace tables or structured walkthroughs

(f) Evaluation of the structure and efficiency of an algorithm

2.1.3 Division by Standard:

Std 8: a, b, c, d, e

Std 9: a, b, c, d, e, f

Std 10: a, b, c, d, e, f

Where a topic is covered in more than one standard the depth and content will vary from standard to standard.

2.1.4 Time required: 10% of the available time (approximately 9h) per standard. Module 1 and Module 2 together should not take more than 60% of the available time (approximately 54h) per standard.

2.2 MODULE 2: Implementation of Algorithms - Theory and Practice
- COMPULSORY -

2.2.1 Objectives

(a) To enable a pupil to write and debug a computer program for a specific algorithm

(b) To enable a pupil to write and debug a program using applicable data structures

2.2.2 Content

(a) The following basic principles of programming should be included:

(i) General readability

(ii) Meaningful variable names

(iii) Indentation

(iv) Documentation

(v) Testing

(b) Pupils must be able to write programs which meet certain criteria in a selected programming language. The programs should show that the pupil has mastered the topics mentioned in this and other modules.

(c) Debugging techniques, using for example trace tables, structured walkthroughs and debuggers.

(d) Modular programming using features such as

(i) functions

(ii) procedures

(iii) parameters; global and local variables

(iv) any other applicable subprograms

(e) Defensive programming techniques. These include input validation and other tests.

(f) Elementary file handling including use of the following filing techniques:

(i) Sequential/serial files

(ii) Direct access files

2.2.3 Division by Standard:

Std 8: a, b, c, d, e

Std 9: a, b, c, d, e, f

Std 10: a, b, c, d, e, f

Where a topic is covered in more than one standard the depth and content will vary from standard to standard.

2.2.4 Time required: 50% of the available time (approximately 45h) per standard. Module 1 and Module 2 together should not take more than 60% of the available time (approximately 54h) per standard.

2.2.5 NOTE: For the successful presentation of this module it is compulsory that pupils have access to a microcomputer for at least 2 hours per week.

2.3 MODULE 3: Computer Architecture (Hardware) - COMPULSORY -

2.3.1 Objectives

(a) To introduce pupils to the basic architecture of a computer

(b) To enable pupils to select a suitable computer configuration to meet a specific need

2.3.2 Content

(a) Simple model of the computer

(b) Memory, central processor, input and output units and their interaction

(c) The control cycle

(d) Physical properties of secondary storage units

(e) Physical properties of various input and output devices

(f) Data communications devices

2.3.3 Division by Standard:

Std 8:	a, b, d, e
Std 9:	a, b, c, d, e, f
Std 10:	a, b, c, d, e, f

Where a topic is covered in more than one standard the depth and content will vary from standard to standard.

2.3.4 Time required: 4% of the available time (approximately 3,5h) per standard

2.4 MODULE 4: Data Structures and Manipulation - COMPULSORY -

2.4.1 Objectives

(a) To enable the pupil to gain an insight into data representation (integers, reals and characters) in a computer system

(b) To enable the pupil to understand why certain errors occur in programs and how these errors can be overcome

(c) To enable the pupil to decide which data structure should be used in a given situation

(d) To introduce the pupil to the manipulation of data using searching and sorting techniques

2.4.2 Content

(a) Data representation

(i) Positional number systems such as the decimal, binary and hexadecimal systems

Conversions between various bases

(ii) Integer values

a. Addition and subtraction in binary

b. Internal representation using the implementation employed in the selected programming language

c. Errors

(iii) Floating point values

a. Errors

b. Accuracy

c. Internal representation using the implementation employed in the selected programming language

(iv) Alphanumeric data

Character codes such as ASCII and EBCDIC

(b) Data types

(i) Constants and variables

(ii) Operators and expressions

(iii) Logical data type

(iv) User-defined data types

(c) Data structures

(i) Variables, arrays

(ii) Stacks, queues and trees (concepts only)

(iii) Files, records, fields

(iv) Data bases

(d) Data manipulation - techniques such as

(i) Searching techniques

(ii) Sorting techniques

(iii) List manipulation techniques

2.4.3 Division by Standard:

Std 8: a, b, c, d

Std 9: a, b, c, d

Std 10: a, b, c, d

Where a topic is covered in more than one standard the depth and content will vary from standard to standard.

2.4.4 Time required: 7% of the available time (approximately 6h) per standard

2.5 MODULE 5: **Operating Systems and System Software** - COMPULSORY -

2.5.1 Objective

To understand, in broad outline, the functioning of an operating system as a resource manager, and other system software

2.5.2 Content

(a) Introduction to operating systems and systems software

(b) The operating system

(i) Functions

(ii) Job control

(iii) Processing techniques such as multi-tasking/multiprogramming, multiprocessing, time-sharing, real-time systems, batch processing, virtual memory, networking

(iv) input and output control

(c) System utilities

(d) Compilers and interpreters

2.5.3 Division by Standard:

Std 8: a, c, d

Std 9: b, c, d

Std 10: b, c, d

Where a topic is covered in more than one standard the depth and content will vary from standard to standard.

2.5.4 Time required: 4% of the available time (approximately 3,5h) per standard

2.6 MODULE 6: Computers - History, Implications and Applications - OPTIONAL -

2.6.1 Objectives

(a) To provide the pupil with a foundation upon which he can build further computer knowledge and also to which he can relate new developments in the computer field

(b) To consider the socio-economic and other implications of the use of the computer

(c) To make the pupil aware of the indispensible role played by the computer in modern society

(d) To foster an open mind towards possible computer applications and their feasibility

(e) To place into perspective claims made for and against the continued use of computers in society

(f) To consider the role of the computer in education

(g) To rectify misconceptions about computers

(h) To consider issues such as software piracy

2.6.2 Content

(a) Development of the computer and data-processing

- (i) Historical figures
- (ii) Different generations of computers
- (iii) Differences between mainframe, mini- and microcomputers
- (iv) Development of primary and secondary memory and the central processor
- (v) Development of calculating techniques using the abacus, logarithms, slide rules, calculating machines, pocket calculators
- (vi) Development of computers such as MARK I, ENIAC, EDVAC, APPLE series, IBM series
- (vii) Historical development of software

(b) Socio-economic and other implications of computers

(i) Social implications

(ii) The use of computers in areas such as

a. banking

b. law enforcement

c. office automation

d. management information systems

e. education

f. industry

(iii) Discussion of issues such as

a. artificial intelligence

b. security and privacy

c. computer crime

d. ethics and computing

e. computer viruses

2.6.3 Division by Standard:

Std 8: a, b

Std 9: a, b

Std 10: a, b

Where a topic is covered in more than one standard the depth and content will vary from standard to standard.

2.6.4 Time required: 5% of the available time (approximately 4h) per standard

2.7 MODULE 7: Computer Logic - OPTIONAL -

2.7.1 Objectives

(a) To introduce pupils to the basic building blocks which make up a computer's intricate logic circuitry

(b) To introduce pupils to the techniques used in simplifying logical expressions

(c) To design simple logic circuits

2.7.2 Content

(a) Two-state representation of data

(b) Elementary logic operations such as AND, OR and NOT

(c) Gate symbols such as AND, OR and NOT

(d) Truth tables and combinations of logic operations

(e) De Morgan's laws and the simplification of logical expressions

(f) Elementary logic circuits such as the half adder

2.7.3 Division by Standard:

Std 8: a, b, d

Std 9: a, b, c, d, e

Std 10: a, b, c, d, e, f

Where a topic is covered in more than one standard the depth and content will vary from standard to standard.

2.7.4 Time required: 5% of the available time (approximately 4h) per standard

2.8 MODULE 8: Data Communications and Networks - OPTIONAL -

2.8.1 Objective

To give pupils an insight into data communications and networks

2.8.2 Content

(a) Fundamentals of data communication

- (i) Types of transmission
- (ii) Speeds of transmission
- (iii) Communication protocols
- (iv) Errors in data transmission
- (v) Interface devices

(b) Networks

- (i) Local Area Networks (LAN)
- (ii) Wide Area Networks (WAN)

(c) Data communication software

- (i) Emulators
- (ii) Data transfer programs
- (iii) Bulletin boards

2.8.3 Division by Standard:

Std 8: a

Std 9: a, c(ii)

Std 10: a, b, c

Where a topic is covered in more than one standard the depth and content will vary from standard to standard.

2.8.4 Time required: 5% of the available time (approximately 4h) per standard

2.9 MODULE 9: End-User Computing - OPTIONAL -

2.9.1 Objectives

(a) To introduce pupils to the terminology used in application packages

(b) To introduce pupils to office automation techniques

(c) To enable pupils to use a word processor

(d) To enable pupils to develop solutions using a database

(e) To enable pupils to understand spreadsheet concepts

(f) To develop an understanding of a fourth generation language

(g) To enable pupils to display information using graphics packages

2.9.2 Content

(a) Application packages including the following:
* Word processor
* Data base
* Spreadsheet

(b) Utility packages

(c) Graphics packages

(d) An introduction to a fourth generation language (concepts only)

2.9.3 Division by Standard:

Std 8: a, b

Std 9: a, b, d

Std 10: a, b, c, d

Where a topic is covered in more than one standard the depth and content will vary from standard to standard.

2.9.4 Time required: 15% of the available time (approximately 13h) per standard

2.9.5 NOTE: This module must be practically orientated. The necessary software must be available.

2.10 MODULE 10: Modern Computing - OPTIONAL -

2.10.1 Objective

To introduce the pupil to recent advances in the computer field

This module must be seen as an open-ended module which will allow additional topics to be added as newer hardware and software developments appear in the marketplace.

2.10.2 Content

Topics such as

(a) Fourth Generation Languages

(b) Artificial Intelligence

(c) Robotics

(d) Expert Systems

(e) Fifth Generation Computing

(f) Computer Aided Software Engineering (CASE)

2.10.3 Division by Standard

Std 8: one topic

Std 9: a further two topics

Std 10: a further three topics

Where a topic is covered in more than one standard the depth and content will vary from standard to standard.

2.10.4 Time required: 5% of the available time (approximately 4h) per standard

2.11 MODULE 11: Software Engineering - OPTIONAL -

2.11.1 Objective

To introduce the pupil to an approach for developing large computer systems

2.11.2 Content

(a) The Systems Development life cycle

(b) Feasibility studies

(c) Logical and physical design

(d) Project management

(e) Documentation

(f) Testing and implementation of a computer system

(g) Security and privacy

(h) Maintainability of systems

(i) Portability of systems

2.13.3 Division by standard

Std 8: a, b, e

Std 9: a, b, c, d, e, f, h, i

Std 10: a, b, c, d, e, f, g, h, i

Where a topic is covered in more than one standard the depth and content will vary from standard to standard.

2.13.4 Time required: 10% of the available time (approximately 9h) per standard

3. DIDACTIC GUIDELINES

3.1 Compulsory Modules

3.1.1 M1 - Development of Algorithms
3.1.2 M2 - Implementation of Algorithms - Theory and Practice
3.1.3 M3 - Computer Architecture (Hardware)
3.1.4 M4 - Data Structures and Manipulation
3.1.5 M5 - Operating Systems and System Software

3.2 Optional Modules

3.2.1 M6 - Computers - History, Implications and Applications
3.2.2 M7 - Computer Logic
3.2.3 M8 - Data Communications and Networks
3.2.4 M9 - End-User Computing
3.2.5 M10 - Modern Computing
3.2.6 M11 - Software Engineering

3.3 Practical work

3.3.1 Practical work in all compulsory modules is understood to form part of Module 2.

3.3.2 Practical work in the optional modules forms part of each module.

3.4 Evaluation Guidelines

3.4.1 Theory work must be evaluated using tests and examinations.

3.4.2 Programming and other practical work may be evaluated by tasks such as

(a) programming projects throughout the year

(b) a practical examination

(c) written tests and examinations.

3.4.3 The mark for theoretical aspects of the syllabus must not exceed 75% of the total mark in each year. The mark for the practical/programming aspects of the syllabus must not exceed 40% of the total mark in each year. The mark for the algorithmic and programming aspects in the examination must count between 40% and 60% of the examination.

3.4.4 The overall mark allocation per module should reasonably reflect the amount of time spent on each module. The mark allocation should, therefore, not differ by more than a percentage point or two from the time allocation given in 4.1 below.

4. SUGGESTED WEIGHTING FOR EACH MODULE

4.1 Time Weighting (Approximate)

MODULE	MODULE TITLE		HG	SG
Compulsory				
M1	Development of Algorithms	{	10%	10%
M2	Implementation of Algorithms - Theory	60% {		
	and Practice	{	50%	50%
M3	Computer Architecture (Hardware)		4%	4%
M4	Data Structures and Manipulation		7%	7%
M5	Operating Systems and System Software		4%	4%
			75%	75%
Optional				
M6	Computers - History, Implications and Applications		5%	5%
M7	Computer Logic		5%	5%
M8	Data Communications and Networks		5%	5%
M9	End-User Computing		15%	15%
M10	Modern Computing		5%	5%
M11	Software Engineering		10%	10%

4.2 The examination weighting given to the modules should bear a relationship to the time weighting given above. For example the exam weighting for Module 3 - Computer Architecture can vary between 2% and 6%.

4.3 A decision will be taken at the end of each year as to what the relative weighting of each module should be for the following three years (i.e. for those pupils entering Std 8 the following year). The same will apply to the programming language which will be studied for the ensuing 3 years. This will enable the syllabus emphasis to change with the rapid changes experienced in the computer industry.

Appendix C: Analysis of Computer Science 1 Syllabuses

1. Table 1 - Condensed Summary
2. Table 2 - Details
3. Table 3 - Summarised comparisons

Table 1

Comparative University Computer Science 1 Syllabuses

Condensed Summary

Contents	Wits	TUK	RU	Dbn	PUCHO	PMB	SU	UOFS	RAU	UPE	King	Warwick
	Universities											
Algorithm concepts	y		y	y	y	y	y	y	y			y
Computation concepts			y				y		y	y		
Prog dev methodology		y	y			y	y	y		y	y	y
Language used		pasc	mod2	pasc	pasc		mod2	pasc		pasc	pasc	pasc
Data rep & no. systems	y		y	y	y	y		y		y	y	y
Data types & structures	y	y	y	y	y	y	y	y	y	y	y	y
Formal methods				y		y						y
Computer architecture	y	y	y		y	y	y	y	y	y	y	
Processing	y			y	y		y	y		y	y	y
Computer Logic	y											
Intro to software	y	y		y	y	y	y	y	y	y	y	
Modern trends	y	y					y					
Soc,Ethics & hist	y	y					y	y				
EUC				y		y		y		y	y	
SDLC				y			y	y			y	
microprogramming	y						y				y	y

Table 2

Comparative University Computer Science 1 Syllabuses

Details

Category Contents	Wits	TUK	RU	Dbn	PUCh	PMB	Stell	UOFS	RAU	UPE	King's	Warwi
Algorithms												
Algorithm concepts	y					y		y	y			y
Algorithm design			y	y	y		y	y				
Algorithm analysis & corr	y		y				y					y
Computation concepts												
Program and computatio			y						y			
Computability and compl			y				y			-		
Problem solving										y		
Program development												
Program development m		y						y			y	
Object orientated design		y	y									
Control structures										y		
Intro to programming						y						
Programming techniques						y	y			y	y	y
Debugging						y	y			y	y	
Documentation			y			y					y	
Languages used		pasc	mod2	pasc	pasc	inspe	mod2	pasc		pasc	pasc	pasc
Data representation												y
Binary numbers	y		y	y	y	y		y			y	
Octal, hex					y						y	
Boolean algebra	y			y		y						y
Radix notation	y										y	
Integers	y										y	
Floating point	y										y	
Character	y											
Strings	y				y					y		
Arithmetic											y	
Data structures												
Data structures and types	y	y	y	y	y		y	y	y	y	y	y
Records, files		y	y	y	y	y	y	y	y	y	y	y
Lists, stacks, trees and q		y	y		y		y			y	y	y
Arrays, sets		y		y	y		y	y		y	y	
Sets & Venn diagrams						y					y	
Formal methods												
Propositional calculus						y						y
Finite automata, linear gr				y		y						y
Discrete mathematics												y

Table 2

Comparative University Computer Science 1 Syllabuses

Details

Category Contents	Wits	TUK	RU	Dbn	PUChr	PMB	Stell	UOFS	RAU	UPE	King's	Warwi
Architecture												
Hardware struct and circu	y							y	y		y	
Hardware and comms		y			y		y	y		y	y	
User interface								y			y	
Architecture	y	y	y		y	y	y				y	
Processing												
Recursion	y			y			y			y	y	y
Searching, sorting	y				y		y	y		y	y	
Sequential processing					y		y					
Input, output and repetiti								y				
Logical circuit design												
Switching algebra and cir	y											
Basic electronics												
Digital system design	y											
Intro to software												
Systems Software						y						
OS	y				y	y		y		y	y	
Compilers	y				y		y					
Programming languages	y				y		y					
Database				y			y		y	y		
Software and data manag		y			y		y					
Modern trends												
Definition of AI	y											
Parallell processing		y					y					
AI		y					y					
Natural language and lea	y						y					
Expert systems							y					
Social, ethical and historical issues												
Social and ethical questio	y						y					
Responsibility and challe		y										
History	y	y						y				
Privacy, security							y					
End User Computing												
Application packages (wo				y		y		y		y	y	
System Development Lifecycle												
SoftEng & systems analy				y			y	y			y	
Software tools												
Microprogramming												
Microprogramming	y						y					
Assembly language progr											y	
Command language prog												y

Table 3
Summarised Comparisons of Syllabuses

Category Contents	S.A. Schools	S.A. Uni's Combin	U.K. King's College	LEAG Schools	S.A. Schools
Algorithms					
Algorithm concepts		4			
Algorithm design	y	4		y	y
Algorithm analysis & correctness	y	3			y
Computation concepts					
Program and computation concepts		2			
Computability and complexity		2			
Problem solving	y	2			y
Program development					
Program development methodology		2	y		
Object orientated design		2			
Control structures		2			
Intro to programming		1			
Programming techniques	y	2	y		y
Debugging	y	3	y		y
Documentation		2	y		
Languages used	logo&pasc	asc&mo	pasc		logo&pasc
Data representation					
Binary numbers	y	6	y	y	y
Octal, hex	y	1	y		y
Boolean algebra	y	3			y
Radix notation		1	y		
Integers		1	y	y	
Floating point	y	1	y		y
Character	y	1		y	y
Strings		3			
Arithmetic	y	0	y		y
Data structures					
Data structures and types		9	y		
Records, files	y	9	y	y	y
Lists, stacks, trees and queues	y	5	y		y
Arrays, sets	y	6	y		y
Sets & Venn diagrams		1	y		
Formal methods					
Propositional calculus		1			
Finite automata, linear grammars, regular expres		2			
Discrete mathematics		0			
Architecture					
Hardware struct and circuits		3	y		
Hardware and comms	y	5	y		y
User interface		1	y		
Architecture	y	5	y	y	y

Table 3

Summarised Comparisons of Syllabuses

Category Contents		S.A. Schools	S.A. Uni's Combine	U.K. King's College	LEAG Schools	S.A. Schools
Processing						
	Recursion		4	y		
	Searching, sorting	y	4	y	y	y
	Sequential processing		1			
	Input, output and repetition		1			
Logical circuit design						
	Switching algebra and circuit design		1			
	Basic electronics	y	0			y
	Digital system design		1			
Intro to software						
	Systems Software	y	1			y
	OS	y	5	y		y
	Compilers		3		y	
	Programming languages		3		y	
	Database		4			
	Software and data management		2			
Modern trends						
	Definition of AI		1			
	Parallell processing		2			
	AI	y	2			y
	Natural language and learning		2			
	Expert systems		1			
Social, ethical and historical issues						
	Social and ethical questions	y	2			y
	Responsibility and challenges		1			
	History	y	3			
	Privacy, security	y	1			y
End User Computing						
	Application packages (wordpro,ss, db)	y	4	y		y
System Development Lifecycle						
	SoftEng & systems analysis	y	3	y	y	y
Microprogramming						
	Microprogramming		2			
	Assembly language programming		0	y		

Appendix D: Designers' Questionnaire

1. Designers' questionnaire (in English).
2. Designers' questionnaire (in Afrikaans).
3. Summarised Response to Questionnaire 1 Section 1.
4. Summarised Response to Questionnaire 1 Section 2.
5. Summarised Response to Questionnaire 1 Section 3.

1. Computer Studies in schools in South Africa is by now a well-established subject in Department of Education and Culture schools.

The following reasons for offering the subject at school level are often advanced. Could you rate your opinion of these by ticking one choice for each reason put forward, where 1 signifies Full agreement and 5 signifies Total disagreement?

a) Computers are pervading industrial societies and are likely to be important in all countries. Since schools prepare children for life, they should prepare them to deal with computers.

1	2	3	4	5

b) Parents and the community expect us to provide the pupils with Computer Studies.

1	2	3	4	5

c) Children should learn to operate computers so as to be able to enter the work place with basic usable skills and knowledge in the computer field.

1	2	3	4	5

d) Computer Studies can have a beneficial effect on mathematical skills.

1	2	3	4	5

e) The subject will provide the gifted pupil with a further challenge.

1	2	3	4	5

f) Computers allow for exploratory learning.

1	2	3	4	5

g) Computers encourage pupils to learn by working together.

1	2	3	4	5

h) Learning with computers requires less memorising of facts and more information-handling and problem-solving.

1	2	3	4	5

i) Problem solving skills achieved in this subject will be transferable to other areas.

1	2	3	4	5

j) There is a dearth of skilled manpower in South Africa and computer skills are greatly in demand. Schools therefore help in fulfilling this need.

1	2	3	4	5

k) It is a good basis for continuing with Computer Science at university.

1	2	3	4	5

Any other reasons for offering Computer Studies as a 6th or 7th subject?

l) ....
....

m) ....
....

Could you rank these reasons in order of importance by filling in a number from 1 to 11 (or 13 if you added any other reasons under l. and m.) corresponding to each of the reasons given above? The most relevant reason should be ranked 1 down to the least relevant which should be ranked 11 (or 13).

a)	b)	c)	d)	e)	f)	g)	h)	i)	j)	k)	l)	m)

2. There are some differences between the proposed and existing syllabuses. What problems were associated with the old syllabus, and how will the new syllabus attempt to rectify these?

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

Ctd /3.

3. Problems have been associated with the teaching of Computer Studies.
Several reasons for these have been advanced.

Could you indicate your opinion of these by ticking one choice for each problem or reason put forward, where 1 signifies Full agreement and 5 signifies Total disagreement?

a) There are not enough qualified teachers.

1	2	3	4	5

b) Teachers need constant retraining.

1	2	3	4	5

c) Since pupils are selected on merit to take Computer Studies, the subject is seen as elitist.

1	2	3	4	5

d) It would be better to introduce computer literacy for all pupils than Computer Studies for a few.

1	2	3	4	5

e) If the subject is introduced as a sixth subject, it will displace a subject that may have given the pupil a broader education.

1	2	3	4	5

f) The cost of purchasing computers to enable schools to offer the subject as a sixth subject is enormous, and could be better spent on other areas.

1	2	3	4	5

g) The hardware is constantly becoming outdated and this leads to an ongoing financial drain.

1	2	3	4	5

h) As a seventh subject offered after hours, pupils find it difficult to persevere in the subject, and there is a considerable drop-out rate. This may result in less pupils being interested in studying Computer Science at university at a later stage.

1	2	3	4	5

Ctd /i)

i) There is an overlap with universities' first year syllabi.

1	2	3	4	5

j) Although doing Computer Studies at school gives these pupils a head start (as does Advanced Maths and Accounting), these are precisely the pupils who do not need this advantage.

1	2	3	4	5

k) Students find it demotivating to repeat a lot of the work in the first year at University.

1	2	3	4	5

l) Any other?

....

....

m)

....

....

Could you rank these problems in order of importance by filling in a number from 1 to 11 (or 13 if you added any other reasons under l. and m.) corresponding to each of the reasons given above? The most important problem should be ranked 1, down to the least relevant one which should be ranked 11 (or 13).

a)	b)	c)	d)	e)	f)	g)	h)	i)	j)	k)	l)	m)

Thank you for your time.

Rinette Roets.

1. Rekenaarstudie word in die meeste skole van die Departement Onderwys en Kultuur aangebied. Die volgende redes word soms geopper vir die aanbieding van die vak op skoolvlak. Kan u u mening omtrent die belangrikheid van die redes aandui deur een van die opsies te merk, waar 1 'Baie belangrik' is en 5 'Totaal onbelangrik' is.

a) Rekenaars kom orals in geïndustrialiseerde lande voor en sal in alle lande belangrik wees. Omdat skole kinders vir die lewe voorberei, moet hulle dus ook kinders voorberei om met rekenaars te kan werk.

1	2	3	4	5

b) Ouers en die gemeenskap verwag dat skole Rekenaarstudies aanbied.

1	2	3	4	5

c) Kinders moet leer om met rekenaars te werk om sodoende die arbeidsmark in die rekenaarwêreld to kan betree.

1	2	3	4	5

d) Rekenaarstudie het 'n positiewe invloed op wiskundige vaardighede.

1	2	3	4	5

e) Die vak sal die begaafde kind 'n ander uitdaging bied.

1	2	3	4	5

f) Rekenaars laat kinders toe om deur ontdekking te leer.

1	2	3	4	5

g) Rekenaars moedig kinders aan om te leer deur saam te werk.

1	2	3	4	5

h) Rekenaars lê minder klem op memorisering en meer op probleemoplossing en die hantering van inligting.

1	2	3	4	5

i) Probleemoplossingsvaardighede wat in hierdie vak aangeleer is, kan na ander gebiede oorgedra word.

1	2	3	4	5

j) Daár is 'n gebrek aan geskoolde arbeid in Suid-Afrika en rekenaarvaardighede is in aanvraag. Skole help om hierdie leemte te vul.

1	2	3	4	5

k) Die vak bied 'n goeie basis vir verdere Rekenaarwetenskap studies op universiteit.

1	2	3	4	5

Enige ander redes vir die aanbieding van Rekenaarstudies as 'n 6de of 7de vak?

l) ....

....

m) ....

....

Kan u 'n rangorde van tussen 1 en 11 (of 13 as u ander redes onder l. and m. aangevul het) toeken aan die bostaande redes? The belangrikste of mees beduidende rede moet die rangorde 1 kry en die mins belangrik 'n rangorde van 11 (of 13).

a)	b)	c)	d)	e)	f)	g)	h)	i)	j)	k)	l)	m)

2. Daar is verskeie verskille tussen die ou en die nuwe sillabus. Watter probleme het die ou sillabus gehad, en hoe word dit deur die nuwe sillabus opgelos?

3. Verskeie probleme word ondervind met die aanbieding van Rekenaarstudie op skool. Hoe belangrik is elk van die probleme? Kan u u mening omtrent die belangrikheid van die voorgestelde redes aandui deur een van die opsies te merk, waar 1 'Baie belangrik' is en 5 'Totaal onbelangrik' is.

a) Daar is 'n tekort aan gekwalifiseerde onderwysers.

1	2	3	4	5

b) Onderwysers moet voortdurend heropgelei word.

1	2	3	4	5

c) Rekenaarstudie word beskou as slegs vir die elite omdat skoliere op grond van prestasie kwalifiseer om die vak te neem.

1	2	3	4	5

d) Dit sou beter wees om rekenaarsgeletterdheid aan alle skoliere in die skool aan te bied eerder as om Rekenaarstudie aan net 'n paar te gee.

1	2	3	4	5

e) As die vak as 'n sesde vak aangebied word, sal dit moontlik 'n ander vak wat die kind 'n breër opvoeding gee, vervang.

1	2	3	4	5

f) Die onkoste verbonde aan die aankoop van rekenaars is groot, en skole of die Onderwysdepartement kan die geld beter op ander gebiede bestee.

1	2	3	4	5

g) Rekenaarapparatuur verouder vinning, en om byderwets te bly verg aanhoudende onkoste.

1	2	3	4	5

h) Skoliere vind dit moelik om te volhard met die vak as dit as 'n 7de vak naskools aangebied word. Die uitvalsyfer is redelik hoog. Moontlik veroorsaak dit dat hierdie skoliere minder belangstel in verdere Rekenaarstudie op universiteit.

1	2	3	4	5

i) Die skool- en universiteitskursusse oorvleuel mekaar.

1	2	3	4	5

j) Die skoliere wat Rekenaarstudie op skool geneem het, het 'n voorsprong op universiteit (net soos met Gevorderde Wiskunde en Rekeningkunde), maar dit is juis nie die studente wat so 'n voorsprong nodig het nie.

1	2	3	4	5

k) Studente vind dit demotiverend om baie van die werk op universiteit te herhaal.

1	2	3	4	5

l) Enige ander probleme?

....

....

m)

....

....

Kan u 'n rangorde van tussen 1 en 11 (of 13 as u ander redes onder l. and m. aangevul het) toeken aan die bostaande redes? The belangrikste of mees beduidende rede moet die rangorde 1 kry en die mins belangrik 'n rangorde van 11 (of 13).

a)	b)	c)	d)	e)	f)	g)	h)	i)	j)	k)	l)	m)

Baie dankie vir u tyd.

Rinette Roets.

3. Summarised Response to Questionnaire 1 Section 1

Syllabus Designers' Responses

a) Computers are pervading industrial societies and are likely to be important in all countries. Since schools prepare children for life, they should prepare them to deal with computers.

1	2	3	4	5
7	1	2		2

b) Parents and the community expect us to provide the pupils with Computer Studies.

1	2	3	4	5
2	3	3	3	1

c) Children should learn to operate computers so as to be able to enter the work place with basic usable skills and knowledge in the computer field.

1	2	3	4	5
5	1	3		3

d) Computer Studies can have a beneficial effect on mathematical skills.

1	2	3	4	5
1	4	2	2	3

e) The subject will provide the gifted pupil with a further challenge.

1	2	3	4	5
4	4	4		

f) Computers allow for exploratory learning.

1	2	3	4	5
3	6	1		2

g) Computers encourage pupils to learn by working together.

1	2	3	4	5
3	3	3	2	1

h) Learning with computers requires less memorising of facts and more information-handling and problem-solving.

1	2	3	4	5
5	6			1

i) Problem solving skills achieved in this subject will be transferable to other areas.

1	2	3	4	5
4	4	2	1	1

j) There is a dearth of skilled manpower in South Africa and computer skills are greatly in demand. Schools therefore help in fulfilling this need.

1	2	3	4	5
1	2	3	3	4

k) It is a good basis for continuing with Computer Science at university.

1	2	3	4	5
4	4	2		2

Any other reasons for offering Computer Studies as a 6th or 7th subject?

1) Problem solving and data processing using computers affects many other subjects. This makes an understanding of computers essential.

   Computer Studies allows the development of moral values such as truthfulness and modesty.

   Computers are (in themselves) an interesting and stimulating subject of study.

   Pupils who do not intend to follow a computing career deliberately choose the school subject to help them in their future (non-computing) careers.

   (6x) More opportunity for average students should be created

Response Questionnaire 1

Section 1

Qnc	1	2	3	4	5	Agreement rating
1a	7	1	2		2	11
1b	2	3	3	3	1	2
1c	5	1	3		3	5
1d	1	4	2	2	3	-2
1e	4	4	4			12
1f	3	6	1	2		10
1g	3	3	3	2	1	5
1h	5	6			1	14
1i	4	4	2	1	1	9
1j	1	2	3	3	4	-7
1k	4	4	2		2	8

Sorted in order of agreement

Qno	Agreement Rating
1h	14
1e	12
1a	11
1f	10
1i	9
1k	8
1g	5
1c	5
1b	2
1d	-2
1j	-7

Ranking of Questionnaire 1 Section 1

Response no

Qnc	1	2	3	4	5	6	7	8	9	10	11
1a	2	1	13	1	1	6	3	1	1	7	1
1b	5	12	13	11	5	11	7	2	5	9	3
1c	1	11	13	3	4	1	8	3	3	10	4
1d	11	8	13	9	6	7			8	11	7
1e	6	7	13	5	9	3			6	5	5
1f	7	5	13	6	8	4	2		11	6	8
1g	3	6	13	10	11	5	1		10	8	10
1h	8	2	13	2	2	9	4		9	4	11
1i	9	3	13	7	7	2			4	1	6
1j	4	10	13	8	10	10	9		2	2	2
1k	12	9	13	4	3	8		4	4	3	9

Sorted Weigted Ranking

Response no

Qnc	1	2	3	4	5	6	7	8	9	10	11	Rank Avg	Rank Positic	Agreement Rating
1a	2	1	13	1	1	6	3	1	1	7	1	3.4	1	11
1c	1	11	13	3	4	1	8	3	3	10	4	5.5	2	5
1h	8	2	13	2	2	9	4	13	9	4	11	7.0	3	14
1i	9	3	13	7	7	2	13	13	4	1	6	7.1	4	9
1f	7	5	13	6	8	4	2	13	11	6	8	7.5	5	10
1k	12	9	13	4	3	8	13	4	4	3	9	7.5	6	8
1b	5	12	13	11	5	11	7	2	5	9	3	7.5	7	2
1j	4	10	13	8	10	10	9	13	2	2	2	7.5	8	-7
1e	6	7	13	5	9	3	13	13	6	5	5	7.7	9	12
1g	3	6	13	10	11	5	1	13	10	8	10	8.2	10	5
1d	11	8	13	9	6	7	13	13	8	11	7	9.6	11	-2

Response Questionnaire 1
Section 1

Sorted Weigted Ranking

Qno		Rank Avg	Rank Posit	Agreement Rating
1a		3.4	1	11
1c		5.5	2	5
1h		7.0	3	14
1i		7.1	4	9
1f		7.5	5	10
1k		7.5	6	8
1b		7.5	7	2
1j		7.5	8	-7
1e		7.7	9	12
1g		8.2	10	5
1d		9.6	11	-2

Stripped Sorted Weighted Ranking

Qno		Rank Avg	Rank Positic	Agreement Rating
1a		3.4	1	11
1c		5.5	2	5
1h		7.0	3	14
1i		7.1	4	9
1f		7.5	5	10
1k		7.5	6	8
1j		7.5	7	-7
1e		7.7	8	12

Sorted Weigted Ranking by category

Categor		Qno	avg	rank	rating		
cat		1f	7.5	6	10		
cat		1i	7.1	4	9		
cat		1h	7.0	3	14		
cat		1g	8.2	10	5	catsum	7.4
it		1j	7.5	8	-7	itsum	7.5
odd		1k	7.5	5	8	oddsum	7.5
ped		1e	7.7	9	12		
ped		1d	9.6	11	-2	pedsum	8.6
soc		1b	7.5	7	2		
soc		1a	3.4	1	11	socsum	5.4
voc		1c	5.5	2	5	vocsum	5.5

Stripped Sorted Ranking

Cat	Qno	avg			rank	ratin
cat	1f	7.5			5	10
cat	1i	7.1			4	9
cat	1h	7.0	catsun	7.2	3	14
it	1j	7.5	itsum	7.5	7	-7
odd	1k	7.5	oddsu	7.5	6	8
ped	1e	7.7	pedsu	7.7	8	12
soc	1a	3.4	socsu	3.4	1	11
voc	1c	5.5	vocsu	5.5	2	5

Sorted average category rankings

socsum	5.4	1
vocsum	5.5	2
catsum	7.4	3
oddsum	7.5	4
itsum	7.5	5
pedsum	8.6	6

Stripped sorted category rankings

socsum	3.4	1
vocsum	5.5	2
catsum	7.2	3
oddsum	7.5	4
itsum	7.5	5
pedsum	7.7	6

4. Summarised Response to Questionnaire 1 Section 2.

There are some differences between the proposed and existing syllabuses. What problems were associated with the old syllabus, and how will the new syllabus attempt to rectify these?

Response 1.
Problems with old:

In some aspects it was out of date; not well structured; the balance between programming and theory was not well defined; it was possible to fulfil the requirements of the syllabus without doing programming; the syllabus was set very widely, allowing for too much own interpretation.

The new:

Modules are better structured; the necessity for a second programming language has been removed; there is greater emphasis on applications software. The compulsory topics give a good overview, while the optional modules are supplementary (aanvullend). Problems will be experienced in choosing the optional modules. Overall there is a better balance. The contents are defined better, but allows for enough adaptability in the fast changing IT world.

Response 2.
The syllabus in the past has been very rigid, and difficult to adapt to the changing nature of IT. The new syllabus has left topics open-ended. The move is towards the use of application software and therefore a de-emphasis on programming skills. It is the responsibility of education to expose all pupils to the types of application in which computers are used i.e. computer literacy for all, introduced cross-curricularly. Computer Studies could then be offered to the interested group on an informal after hours basis.

Response 3.
The new syllabus is more flexible to cater for changes in technology.

Response 4.
The old syllabus was dated. The optional modules will allow more up-to-date subject matter to be added.

Response 5.
The new syllabus is more flexible to adapt to new developments. To add to flexibility, the programming language is not specified. (for example). Greater emphasis has been put on communication and other new developments.

Response 6&7.
Nil

Response 8.
The existing syllabus allowed enough scope for adjustment. The proposed syllabus also allows for adjustment. The proposed module on educational software is not ideal. (Writer's note -

this has now been removed). Provided there are teachers capable to teach them, the other modules are good.

Response 9.

The new syllabus allows for a dynamic syllabus (leerplan) to adapt to changing technology.

Response 10

The old syllabus was somewhat too rigid. The new is designed to be more flexible. The new syllabus de-emphasizes programming. 4GL's will be allowed under the new syllabus.

Response 11

Some of the theory modules that could not be demonstrated in an application have been switched to optional modules; dated sections have been updated; new developments have been added in; data communications and end user computing reflect the changing face of subject; the need to introduce a second programming language has been dropped; more emphasis has been laid on the development of algorithms as distinct from programming - this should be carried through.

Response 12

The new syllabus is less academic - catering for wider group of pupils; it is more market related, teaching skills to allow pupils to enter the jobmarket sooner; it caters for differentiated preferences.

5. Summarised Response to Questionnaire 1 Section 3.

Problems have been associated with the teaching of Computer Studies. Several reasons for these have been advanced.

a) There are not enough qualified teachers.

1	2	3	4	5
10		2		

b) Teachers need constant retraining.

1	2	3	4	5
3	4	2	3	

c) Since pupils are selected on merit to take Computer Studies, the subject is seen as elitist.

1	2	3	4	5
3	1	3	3	2

d) It would be better to introduce computer literacy for all pupils than Computer Studies for a few.

1	2	3	4	5
5	2	3	2	

e) If the subject is introduced as a sixth subject, it will displace a subject that may have given the pupil a broader education.

1	2	3	4	5
1	2	3	2	4

f) The cost of purchasing computers to enable schools to offer the subject as a sixth subject is enormous, and could be better spent on other areas.

1	2	3	4	5
0	2	1	5	4

g) The hardware is constantly becoming outdated and this leads to an ongoing financial drain.

1	2	3	4	5
1	2	4	3	2

h) As a seventh subject offered after hours, pupils find it difficult to persevere in the subject, and there is a considerable drop-out rate. This may result in less pupils being interested in studying Computer Science at university at a later stage.

1	2	3	4	5
4	3	2	2	1

i) There is an overlap with universities' first year syllabi.

1	2	3	4	5
2	5	2	2	1

j) Although doing Computer Studies at school gives these pupils a head start (as does Advanced Maths and Accounting), these are precisely the pupils who do not need this advantage.

1	2	3	4	5
2	2	2	2	3

k) Students find it demotivating to repeat a lot of the work in the first year at University.

1	2	3	4	5
1	7	1		2

l) Any other?

Universities underestimate the level at schools and therefore some areas are repeated unnecessarily at schools. Parents think Computer Studies is essential in order to be able to use a computer.
The emphasis at school is on programming which is not essential in later (business) life.
Computer Studies is offered only as a seventh subject.
Pupils experience problems associated with travel to get to centres.
Taking the subject extra-curricularly puts strain on the pupils (who mostly already have a full day's activities).
The costs of hardware and software are especially inhibiting if the subject is to be made available to a wider selection of pupils.

Other comments - summarised
(3x) - Programming is over-emphasized.
(2x) - Extra burden on pupils. (1x) - Travel problems.
(2x) - Costs.

Response Questionnaire 1
Section 3

	Qno	(2 1	1 2	0 3	-1 4	-2) 5	Agreement Rating
*	3a	10		2			20
*	3b	3	4	2	3		7
	3c	3	1	3	3	2	0
	3d	5	2	3	2		10
	3e	1	2	3	2	4	-6
	3f		2	1	5	4	-11
	3g	1	2	4	3	2	-3
*	3h	4	3	2	2	1	7
	3i	2	5	2	2	1	5
	3j	2	2	2	2	3	-2
	3k	1	7	1		2	5

Sorted in order of agreement

	Qno	Agreement Rating
*	3a	20
	3d	10
*	3b	7
*	3h	7
	3k	5
	3i	5
	3c	0
	3j	-2
	3g	-3
	3e	-6
	3f	-11

Ranking of Questionnaire 1 Section 3

Response no

Qno	1	2	3	4	5	6	7	8	9	10	11	12
3a	4	1		3	1	1	1	1	1	1	2	4
3b	2	2		4	6	3	2		2		6	3
3c	1		11	8	4	11		2	10		10	1
3d	5	3		9	9	6		3	3		9	5
3e	8	7		10	12	7			7		7	7
3f	6	9		6	10	8		4	11	2	11	8
3g	7	8		7	11	2		6	4		8	10
3h	9	4		5	3	9	3	5	8		1	9
3i	10	10		11	7	4			5		3	6
3j	11	5		12	8	10			9		4	2
3k	12	6	12	13	5	5			6		5	11

Sorted Weighted Ranking of Questionnaire 1 Section 3

Response no

Qno	1	2	3	4	5	6	7	8	9	10	11	12	Avg Rank	Rank Position	Agreement Rating
3a	4	1	13	3	1	1	1	1	1	1	2	4	2.8	1	20
3b	2	2	13	4	6	3	2	13	2	13	6	3	5.8	2	7
3h	9	4	13	5	3	9	3	5	8	13	1	9	6.8	3	7
3d	5	3	13	9	9	6	13	3	3	13	9	5	7.6	1	10
3c	1	13	11	8	4	11	13	2	10	13	10	1	8.1	2	0
3f	6	9	13	6	10	8	13	4	11	2	11	8	8.4	3	-11
3g	7	8	13	7	11	2	13	6	4	13	8	10	8.5	4	-3
3i	10	10	13	11	7	4	13	13	5	13	3	6	9.0	5	5
3j	11	5	13	12	8	10	13	13	9	13	4	2	9.4	6	-2
3k	12	6	12	13	5	5	13	13	6	13	5	11	9.5	7	5
3e	8	7	13	10	12	7	13	13	7	13	7	7	9.8	8	-6

Response Questionnaire 1
Section 3

Sorted Weighted Ranking of Questionnaire 1 Section 3

Qno	Avg Rank	Rank Position	Agreement Rating
3a	2.8	1	20
3b	5.8	2	7
3h	6.8	3	7

3d	7.6	1	10
3c	8.1	2	0
3f	8.4	3	-11
3g	8.5	4	-3
3i	9.0	5	5
3j	9.4	6	-2
3k	9.5	7	5
3e	9.8	8	-6

Appendix E: University Questionnaire

1. University Computer Science department Questionnaire (in English).

2. University Computer Science department Questionnaire (in Afrikaans).

3. Summarised response to University Questionnaire.

1. Universiteit:

....

2. Pos (d.w.s. Dosent, Departementshoof ens.):

....

3. Fakulteit, Departement en Vak (byv., Handel/ Rekeningkunde/ Informatika - praktiese programmering):

....

4. Het u al vantevore die nuwe skoolleerplan vir Rekenaarstudie gesien?

Ja ☐ Nee ☐

5. Indien 'Ja' in 4, was u betrokke by die samestelling van die bestaande of beoogde sillabus vir Rekenaarstudie op skool?

Ja ☐ Nee ☐

6. Indien 'Ja' in 5, wat was die aard van u betrokkenheid?

Kommentaar	Advies	Konsultasie	Deelname
☐	☐	☐	☐

7. Sommige universiteite het deelgeneem aan die samestelling van die nuwe sillabus. Voel u dat meer universiteite geraadpleeg moes gewees het?

Baie belangrik	Ja	Onseker	Nee	Glad nie
☐	☐	☐	☐	☐

Kommentaar:...

....

8. Elders (d.w.s. in Europa en die VSA) is die aantal eerstejaarstudente in Rekenaarwetenskap aan die verminder. Is daar 'n dergelyke trant by u universiteit in RK of RI?

RK			RI		
Minder Studente	Onveranderd	Meer Studente	Minder Studente	Onveranderd	Meer Studente
☐	☐	☐	☐	☐	☐

Kommentaar:...

9. Hierna volg aanmerkings op die huidige en nuwe sillabusse ('n kopie van die nuwe is ingesluit). Probleme wat ondervind word met die dosering van RK of RI as gevolg van die aanbieding van die skoolvak Rekenaarstudie word ook genoem. Wat is u mening oor die volgende stellings?

a) Daar is 'n oorvleueling van die matrieksillabus en dit wat op universiteit in RK en RI gegee is. Dit is 'n vermorsing van onderrighulpbronne. Stem u saam?

Baie beslis ☐ Tot 'n mate ☐ Onseker ☐ Nee ☐ Glad nie ☐

Kommentaar:
....

....

b) Deur Rekenaarstudie op skool aan te bied, word die gaping tussen beter en swakker studente vergroot. Skep dit vir u probleme?

Baie beslis ☐ Tot 'n mate ☐ Onseker ☐ Nee ☐ Glad nie ☐

Kommentaar:
....

....

c) Tegnologie is aanhoudend aan die verander. Nuwe programmatuur verg gedurig nuwe rekenaarapparatuur. Die onkoste verbonde hieraan is groot. Om meer rasionalisasie te weeg te bring behoort hierdie vak op Universiteits- of Technikonvlak aangebied te word eerder as op skool.

Baie beslis ☐ Tot 'n mate ☐ Onseker ☐ Nee ☐ Glad nie ☐

Kommentaar:
....

....

1. University:

....

2. Present position (i.e., Lecturer, Head of Department etc.):

....

3. Faculty, Department and Subject (e.g., Commerce/ Accounts/ BIS - programming):

....

4. Have you seen the proposed new school syllabus for Computer Studies prior to today? (please tick the block showing your response)

Yes ☐ No ☐

5. If 'Yes' in 4 above, were you involved in drawing up the existing or new syllabus for Computer Studies at school?

Yes ☐ No ☐

6. If 'Yes' in 5 above, what was the nature of that involvement?

Comment ☐ Advice ☐ Consultation ☐ Participation. ☐

7. Some universities have been involved in drawing up the school syllabuses. Do you feel more should be consulted?

Vital ☐ Advantageous ☐ Unsure ☐ No ☐ Not at all ☐

Comments:...

....

8. Elsewhere (i.e. Europe and the States) the number of students choosing Computer Science is dropping. Are you experiencing a similar trend in CS, SEng or BIS?

CS			BIS/SEng		
Decrease	Same	Increase	Decrease	Same	Increase
☐	☐	☐	☐	☐	☐

Comments:...

9. Some comments on the present and proposed syllabus (a copy of which is included) follow. Also mentioned are some problems associated with the teaching of Computer Studies at schools as related to university teaching. Could you rate your opinion of these comments by ticking the relevant box for each point?

a) There is some overlap in the matriculation syllabus and the first year CS, SEng or BIS course. This is wasteful of teaching resources.

Agree fully	Agree	Unsure	Disagree	Disagree totally
☐	☐	☐	☐	☐

Comments: ....

....

b) Teaching two different levels of students (i.e., those who have and those who have not taken Computer Studies at school) causes difficulties by widening the spectrum between the brightest and least bright students. Have you found this to be a problem?

Very much	Somewhat	Unsure	No	Not at all
☐	☐	☐	☐	☐

Comments: ....

....

c) Technology is changing constantly. More, and more powerful, software is becoming available that requires constant updating of hardware. The costs are exorbitant and in the interest of rationalisation in South Africa, this subject would be better addressed at University (or Technikon) level than at schools.

Agree fully	Agree	Unsure	Disagree	Disagree totally
☐	☐	☐	☐	☐

Comments: ....

....

d) A problem mentioned in the U.K. is that some teaching at school level has not been adequate (or has had the wrong emphasis). This leads to the need to reteach certain areas at university. Have you found this a problem?

Very much	Somewhat	Unsure	No	Not at all
☐	☐	☐	☐	☐

Comments: ....

....

e) Some optional topics in the new syllabus can be covered only cursorily in the time allocated and thus is not of much use to the pupils.

Agree fully	Agree	Unsure	Disagree	Disagree totally
☐	☐	☐	☐	☐

Comments: ....

....

f) Although the more advanced topics cannot be explored fully, this does allow for an introduction to these topics at school level that can then be taught more easily at a later stage (the spiral approach).

Agree fully	Agree	Unsure	Disagree	Disagree totally
☐	☐	☐	☐	☐

Comments: ....

....

g) Some optional topics require expert knowledge that is probably outside the capability of most Computer Studies teachers without constant retraining.

Agree fully	Agree	Unsure	Disagree	Disagree totally
☐	☐	☐	☐	☐

Comments: ....

....

h) The purpose of the course at school does not appear to be primarily a stepping stone for further study at University: it is an entity on its own.

Agree fully	Agree	Unsure	Disagree	Disagree totally
☐	☐	☐	☐	☐

Comments: ....

....

i) If you agree with the above statement, would you say that this is necessarily a cause for concern for the designers of the school syllabuses?

No problem	Minor	Unsure	Somewhat	Major problem
☐	☐	☐	☐	☐

Comments: ....

....

j) Scholars who have taken Computer Studies at school are unlikely to take Computer Science at university since some of the 'mystique' of the subject has disappeared.

Agree fully	Agree	Unsure	Disagree	Disagree totally
☐	☐	☐	☐	☐

Comments: ....

....

k) The students who are selected for Computer Studies are almost invariably those who will attend University and can thus take the subject at that level. It is better to expose a wider range of scholars to Computer Literacy.

Agree fully	Agree	Unsure	Disagree	Disagree totally
☐	☐	☐	☐	☐

Comments: ....

....

1) Do you feel that it is advantageous to offer Computer Studies at school level?

Very much so	Somewhat	Unsure	No	Not at all
☐	☐	☐	☐	☐

Could you amplify on your opinion of 1) above?

....

....

....

....

....

10. Any other comments on the syllabuses at school and university or the interaction between school and university?

....

....

....

....

....

....

Thank you very much for your co-operation. If you would like to receive the results of this survey please tick the box below and enter your return address.

Yes ☐ No ☐

Address:

....

....

....

....

R A Roets

d) 'n Probleem wat ondervind word in die Verenigde Koningryk is dat in sekere gevalle die onderwys op skoolvlak van 'n swak gehalte is (of dat dit die verkeerde klem het). Dit lei daartoe dat sekere dele van die vak herhaal moet word op universiteit. Het u al so 'n probleem ondervind?

Baie beslis	Tot 'n mate	Onseker	Nee	Glad nie
☐	☐	☐	☐	☐

Kommentaar:
....

....

e) Daar is nie genoeg tyd om van die opsionele dele van die nuwe sillabus deeglik te behandel nie. Die dele is dus van minder waarde vir die leerling. Stem u saam?

Baie beslis	Tot 'n mate	Onseker	Nee	Glad nie
☐	☐	☐	☐	☐

Kommentaar:
....

....

f) Alhoewel die gevorderde dele van die nuwe sillabus nie deeglik behandel kan word nie, is dit tog nuttig as 'n inleiding tot die universiteitsvak. Stem u saam?

Baie beslis	Tot 'n mate	Onseker	Nee	Glad nie
☐	☐	☐	☐	☐

Kommentaar:
....

....

g) Van die opsionele dele van die nuwe sillabus verg baie gespesialiseerde kennis van die onderwysers. Dit beteken dat die meeste onderwysers dus aanhoudend heropleiding moet ondergaan. Stem u saam?

Baie beslis	Tot 'n mate	Onseker	Nee	Glad nie
☐	☐	☐	☐	☐

Kommentaar:
....

....

h) Die doel van die skoolsillabus is blykbaar nie om die leerling voor te berei vir verdere rekenaarstudie op universiteitsvlak nie. Inteendeel, dit is 'n kursus op sy eie. Stem u saam?

Baie beslis	Tot 'n mate	Onseker	Nee	Glad nie
☐	☐	☐	☐	☐

Kommentaar: ....

....

i) As u saamstem met die bogenoemde stelling, sou u sê dat dit noodwendig 'n probleem skep vir die persone wat die skoolsillabus moet opstel?

Baie beslis	Tot 'n mate	Onseker	Nee	Glad nie
☐	☐	☐	☐	☐

Kommentaar: ....

....

j) Studente wat rekenaarstudie op skool neem is minder geneig om met die vak voort te gaan op universiteit aangesien die aantrekkingskrag (mystique) van die vak verdwyn het. Stem u saam?

Baie beslis	Tot 'n mate	Onseker	Nee	Glad nie
☐	☐	☐	☐	☐

Kommentaar: ....

....

k) Die student wat gekies word om rekenaarstudie te neem op skoolvlak is oor die algemeen juis die studente wat verder op universiteit sal studeer en dus die vak op daardie vlak kan studeer. Dit sou beter wees om meer skoliere 'n basiese rekenaargeletterdheid te gee. Stem u saam?

Baie beslis	Tot 'n mate	Onseker	Nee	Glad nie
☐	☐	☐	☐	☐

Kommentaar: ....

....

1) Voel u dat dit voordelig is om rekenaarstudie op skoolvlak aan te bied?

Baie beslis ☐ Tot 'n mate ☐ Onseker ☐ Nee ☐ Glad nie ☐

Kan u u opinie hieroor verder verduidelik?

....

....

....

....

....

10. Het u enige ander kommentaar oor die sillabus op skool en universiteitsvlak en die verhouding tussen die twee?

....

....

....

....

....

....

Dankie vir u samewerking. As u belangstel in die uitslag van die navorsing, kan u asseblief dit hieronder aandui en 'n adres verskaf?

Ja ☐ Nee ☐

Adres:

....

....

....

....

R A Roets

Universities Questionnaire

Quest No 4, 5

y/n	Seen (4)	Involved (5)
y	8	4
n	18	4
blank	0	8
total	26	16

Quest No 6

c/a/o/p	Nature
c	3
a	0
o	0
p	1
blank	0
total	4

Quest No 7

v/a/u/n/x	Uniinv
v	8
a	10
u	4
n	3
x	0
blank	1
total	26

Quest No 8

d/s/i	Decrease
d	7
s	7
i	10
blank	2
total	26

Quest No 9a

f/a/u/d/t	wasteful
f	1
a	12
u	1
d	11
t	1
blank	0
total	26

Quest No 9b

v/s/u/n/a	problem
v	8
s	12
u	0
n	4
a	1
blank	1
total	26

Quest No 9c

f/a/u/d/t	only ter
f	0
a	3
u	3
d	13
t	7
blank	0
total	26

Quest No 9d

v/s/u/n/x	badteach
v	4
s	11
u	4
n	5
x	2
blank	0
total	26

Quest No 9e, 9f, 9g, 9h

f/a/u/d/t	optio (9e)	spiral (9f)	xpert (9g)	owment (9h)
f	5	7	12	5
a	9	11	11	12
u	2	3	0	6
d	7	5	3	3
t	3	0	0	0
blank	0	0	0	0
total	26	26	26	26

Quest No 9i

n/m/u/s/p	probdes
n	7
m	3
u	1
s	12
p	0
blank	3
total	26

Quest No 9j, 9k

f/a/u/d/t	myst (9j)	complit (9k)
f	1	14
a	5	6
u	3	1
d	11	3
t	5	1
blank	1	1
total	26	26

Quest No 9l

v/s/u/n/a	schobl
v	11
s	8
u	4
n	2
a	0
blank	1
total	26

3. <u>Summarised response to University Questionnaire.</u>

<u>Questionnaire Summary: University influence in SA and relationship between school and university subject.</u>

4. Have you seen the proposed new school syllabus for Computer Studies prior to today? (please tick the block showing your response)

Yes	8	No	18

5. If 'Yes' in 4 above, were you involved in drawing up the existing or new syllabus for Computer Studies at school?

Yes	4	No	

6. If 'Yes' in 5 above, what was the nature of that involvement?

Comment	Advice	Consultation	Participation.
3			1

7. Some universities have been involved in drawing up the school syllabuses. Do you feel more should be consulted?

Vital	Advantageous	Unsure	No	Not at all
8	10	4	3	0

<u>Comments:</u>
Not all should be consulted.
There should be close co-operation between universities and education departments.
Generally a good idea, but universities could be out of touch with requirements of school pupils.
This is the only way universities can draw up realistic curricula.
Universities should take note of the new syllabus.
The new syllabus will force major changes at universities and increase gap between those with CS and those without.
It is important for universities to be aware of the developments at school and vice versa.
Information exchange is sometimes valuable.
The syllabus emphasises programming over EUC: consultation would have sorted that out.
Not all universities should be consulted, but enough to prevent bias.
If more were consulted it could end up with a big argument.
Large committee syndrome would make the cohesion worse.
It would help in removing duplication.
Yes, since decisions affect CS/Bis departments.

8. Elsewhere (i.e. Europe and the States) the number of students choosing Computer Science is dropping. Are you experiencing a similar trend in CS, SEng or BIS?

<u>CS</u>			<u>BIS/SEng</u>		
Decrease	Same	Increase	Decrease	Same	Increase
7	6	7	0	2	9

9. Some comments on the present and proposed syllabus (a copy of which is included) follow. Also mentioned are some problems associated with the teaching of Computer Studies at schools as related to university teaching. Could you rate your opinion of these comments by ticking the relevant box for each point?

a) There is some overlap in the matriculation syllabus and the first year CS, Seng or BIS course. This is wasteful of teaching resources.

Agree fully	Agree	Unsure	Disagree	totally
1	12	1	11	1

Comments:

Theoretically yes, but some teaching is poor so material must be retaught anyway.
University syllabuses must adapt.
Unavoidable.
Unavoidable.
Inevitable.
More communication would improve matters.
Bridging course might be required to solve this.
Unavoidable.
Only wasteful if it affects the majority - which it does not.
University courses will change to accommodate school syllabus.
Yes, and gets worse every year.
Some topics could easily be left to university level e.g. truth tables, logic circuits, i.a..
True in the first semester, less so in the second.
Some people feel that nothing good is done at school but this is incorrect.

b) Teaching two different levels of students (i.e., those who have and those who have not taken Computer Studies at school) causes difficulties by widening the spectrum between the brightest and least bright students. Have you found this to be a problem?

Very much	Somewhat	Unsure	No	Not at all
8	12	0	4	1

Comments:

This is particularly the case with educationally disadvantaged students in class.
Universities must accommodate both groups.
Perhaps a computer literacy course should be offered for lower level students.
An introductory non-credit course is available.
Perhaps a bridging course could be provided.
There is little emphasis at school level of general problem solving.
Not a major problem.

Often those who have not taken C.S. are the best students at Computer Science.
Bridging courses could be arranged.
School studies should eventually become a prerequisite for the university course.
Students keep themselves busy with advanced self-initiated work in first semester.
Good to give the achievers challenges early i.e. at school, to allow them to develop even further at university.

c) Technology is changing constantly. More, and more powerful, software is becoming available that requires constant updating of hardware. The costs are exorbitant and in the interest of rationalisation in South Africa, this subject would be better addressed at University (or Technikon) level than at schools.

Agree fully	Agree	Unsure	Disagree	Disagree totally
0	3	3	13	7

Comments:

No- computing knowledge should be made available as widely as possible.
Basic concepts should be taught at schools.
Basic principles should be taught at school.
Underlying principles remain constant.
The subject should be introduced at primary level.
Perhaps schools should concentrate on theory and thus be less pressurised to keep up with hardware developments.
Basic hardware and software can be acquired - it is not necessary to acquire every new invention.
Agree, but C.S. should still be offered to selected students as at present.
Agree - all the useless Apples sitting in schools.
No - universities cannot meet these costs either.
Possibly, although concepts do not require latest technology; perhaps community centres should be set up and made available to schools and the community.
The latest technology is not required to teach basic concepts.
All scholars need a knowledge of computers.
Pupils must be exposed to computers at school level.
Pupils must develop their talents early.
If costs limited education, there would be no education; the costs of both hardware and software are minimal.

d) A problem mentioned in the U.K. is that some teaching at school level has not been adequate (or has had the wrong emphasis). This leads to the need to reteach certain areas at university. Have you found this a problem?

Very much	Somewhat	Unsure	No	Not at all
4	11	4	5	2

Comments:

Some students have bad programming habits (wherever acquired).

Unqualified teachers is a problem - they should not offer CS.
Competent experienced teachers are essential.
As better qualified teachers emerge, this problem will disappear.
Some areas are not well taught e.g. problem solving: students want to program not analyze.
Most teachers are underqualified.
Schools and universities should co-ordinate their activities.
Reteaching is unavoidable anyway.
Teaching of C.S. is generally adequate to good: reteaching is unavoidable because of disparities.
As above.

e) Some optional topics in the new syllabus can be covered only cursorily in the time allocated and thus is not of much use to the pupils.

Agree fully	Agree	Unsure	Disagree	Disagree totally
5	9	2	7	3

Comments:

Some optional topics are more appropriate than the compulsory topics.
Better than nothing at all.
Bothering too much with trendy (and difficult) topics like AI.
The compulsory topics plus choices give a good background if correctly implemented.
Far too ambitious.
Brief review acceptable provided the student knows the limitations.
Cursory exposure is better than none.
Exposure will enhance general computer literacy.
Fundamentals should be taught in preference.
Some coverage better than none.
Cannot be covered well.
The optional topics are dealt with thoroughly at university level, and there are not many teachers 100% able to teach these topics.
Some topics can be left out to be dealt with at university level.
Overview knowledge may be of some benefit.
7th subject causes time pressure for pupils.
No: EUC (or End User Computing) should be made compulsory for example.

f) Although the more advanced topics cannot be explored fully, this does allow for an introduction to these topics at school level that can then be taught more easily at a later stage (the spiral approach).

Agree fully	Agree	Unsure	Disagree	Disagree totally
7	11	3	5	0

Comments:

Leave out the more advanced topics. They are usually badly taught and change fast.

To provide a background introduction requires an excellent grasp of the subject that few teachers have had an opportunity to achieve.
This may be so as long as no difficult-to-break bad habits are taught.
Only if teachers know the field in depth, which is generally not the case at present.
As long as it does not put the pupils off the topic as a whole.
Teachers are not as stupid or idle as they have been made out to be.
Teachers should be retraining themselves.

g) Some optional topics require expert knowledge that is probably outside the capability of most Computer Studies teachers without constant retraining.

Agree fully	Agree	Unsure	Disagree	Disagree totally
12	11	0	3	0

Comments:

True of advanced, not just optional, topics.
The basics provide enough challenge.
Any C.S. graduate teacher should be able to cope.
It is hard to keep up with developments.
The emphasis should be on background: not chasing after new technology.
Hopefully most teachers can cope with 1st year level work.
Those who gain enough C.S. expertise change jobs.
These topics must be introduced.
Teachers haven't a hope of covering the contents properly.
Introduce differentiated pay scales to attract competent people.
There must be constant (self)retraining anyway.

h) The purpose of the course at school does not appear to be primarily a stepping stone for further study at University: it is an entity on its own.

Agree fully	Agree	Unsure	Disagree	Disagree totally
5	12	6	3	0

Comments:

The course must have a degree of completeness.
Co-operation between school and university should solve this problem.
Schools give basic programming.
It should prepare scholars for the Information Age as well as University.
The syllabus seems to serve both purposes.
The syllabus is too big an entity and too shallow.
This holds for most school subjects.
First year students do not have enough knowledge of EUC.

i) If you agree with the above statement, would you say that this is necessarily a cause for concern for the designers of the school syllabuses?

No problem	Minor	Unsure	Somewhat	Major problem
7	3	1	12	0

Comments:

It is important to match the pupils' needs.
It is more a problem for designers of university courses.
It is only a problem if credits are expected at university.
Universities and the Information Age should be taken into account in the scholars' interest.
Entrance requirements for C.S. at school should be lower to allow those genuinely interested access to course.
The designers of the C.S. school syllabus are perhaps too far removed from the business application of computing.

j) Scholars who have taken Computer Studies at school are unlikely to take Computer Science at university since some of the 'mystique' of the subject has disappeared.

Agree fully	Agree	Unsure	Disagree	Disagree totally
1	5	3	11	5

Comments:

Scholars have the advantage of testing his interest and aptitude.
Novelty seekers are not wanted at university: if the subject is properly taught it should increase their interest.
Mystique is not enough to get through a C.S. course.
This is probably a good thing: universities do not want hackers.
Scholars think C.S.is just programming and they do not see the Science part of it.
Many are disenchanted with C.S..
Bad teaching can switch them off.
Universities do not need mystique seekers.
Not at all: it enables those students to decide earlier that they have a talent for the subject.

k) The students who are selected for Computer Studies are almost invariably those who will attend University and can thus take the subject at that level. It is better to expose a wider range of scholars to Computer Literacy.

Agree fully	Agree	Unsure	Disagree	Disagree totally
14	6	1	3	1

Comments:

Ideally, both levels should be available.
If pupils' interest in C.S. could be awakened at school, more would study the subject at university: Computer Literacy would not achieve this.
Both can be accommodated.
Computer Literacy for all more important than C.S. for a few.
Providing Computer Literacy as a different course would answer this problem.
Most people should have a level of computer literacy.
Both should be taught at schools.
It would be better to provide vocational education.
Computer Literacy is important for all.
Nonsense.
It is good to have a wider range of scholars exposed to computing at school level.
The selection system is bad in that it gives the wrong idea that C.S. is for the mathematically elite.
Computer Literacy is useful.
Specialisation should occur much later.
Both should be available.
Computer Literacy (or EUC) is essential for all.

1) Do you feel that it is advantageous to offer Computer Studies at school level?

Very much so	Somewhat	Unsure	No	Not at all
11	8	4	2	0

Could you amplify on your opinion of 1) above?

Schools can equip people with the vital skills of computing which is required by all (not just those who specialise in the subject later).
If there is more than one subject (i.e. C.S. and Computer Literacy) and the approach is different at schools (no further amplification of this given).
This knowledge is useful for all scholars in later life.
Only if staff, time (not extra-curricularly) and equipment are available, otherwise Computer Literacy might be better.
It is a good background for all students.
Yes, but Computer Literacy is better.
Universities cannot address all topics in this fast developing area; Computer Literacy is necessary intellectual equipment for all.
C.S. assists students in career decisions.
Good for background knowledge and 'training' programmers.
Computer use should be integrated cross-curricularly.
It would be better to offer a general literacy course.
It would be better to expose all students to the capabilities of the computer.
No: there are badly trained teachers and outdated equipment, and hacking is seen as C.S..
Two types of courses are required to cater for future computer users.
Computer Literacy definitely, C.S. perhaps: it should not replace subjects of a more general nature such as History or Geography.

As a 7th optional subject that is C.S. is acceptable or advantageous for pupils.
Computing is essential at school level.
It is imperative to know something about computers.
Only advantageous if good teachers and equipment are available.
Not while unfair selection is part of the system.
Yes: any Computer Literacy is essential.
The subject broadens students and gets them interested.
Yes, stimulates interest.
Yes, preferably as a sixth subject and if the selection is less elitist.
Yes, good background knowledge and skills are picked up and these students are perceived by other students, rightly or wrongly, to have a great advantage.

10. Any other comments on the syllabuses at school and university or the interaction between school and university?

Teaching must be improved so that exemption can be exercised at university level.
The syllabus should concentrate more on literacy and end user computing to complement the university courses.
Real C.S. requires a logical maturity and school pupils may be too young for this level.
More interaction will prevent overlapping/disjoint curricula

Designers and universities must co-ordinate in respect of modern trends.
It will be a long time before C.S. at school will be a prerequisite for Computer Science at university. The subject is still very new in both school and university.
There is no conflict between school and university: at school 'what' is taught, at universities 'why' and 'how'.
Universities may have to offer two first year courses if C.S. is properly done at schools.