SCIENCE AND ENGINEERING RESEARCH COUNCIL
RUTHERFORD APPLETON LABORATORY

INFORMATICS DIVISION

SOFTWARE ENGINEERING GROUP NOTE 125

issued by
R W Witty
29 May 1986

RECOMMENDATIONS TO SCIT CS DEPT
China Trip Report, May 1986

DISTRIBUTION: Prof Chen Xingye, SCIT
Mr Chen Dongfan, SCIT
H Beemer, World Bank
Prof O Zienkiewicz, Swansea
J Burren, RAL
EL/China file

(see next page)

1. INTRODUCTION

In accordance with the guidelines issued by the Chinese Universities Development Project Office, the following recommendations are made as a result of my visit to the South China Institute of Technology Computer Science Department to discuss their teaching and research in the field of Software Engineering.

These recommendations have been discussed with SCIT and a copy of them left with SCIT before my departure.

2. CURRICULUM DEVELOPMENT

1. The Computer Science BSc degree course has its origins in computer hardware and has only recently begun to encompass the software aspects of computer science. The course is now looking to have a 'hardware' and a 'software' stream. I would urge SCIT to try to avoid too great a polarisation into two 'factions'. Graduates need to be 'systems engineers' with a good grounding in both hardware and software principles. The idea of a common course in years 1 and 2, with options in the 3rd and 4th years is good. To avoid the hardware v software polarisation students should be able to choose both hardware and software options and not be too rigidly divided into one faction or the other.

2. The current operating system course uses the source code of Unix version 6 and Chinese language text books on V6 and V7. SCIT has no ability to actually run the Unix operating system. I suggest that, as Unix V6 has been obsolete for 10 years, and was considerably poorer quality than V7, its use should be replaced by a study of the current Berkeley system Unix, BSD4.3. Many good, up to date text books, in English are now available for Unix. (SCIT need to be able to run Unix too, see below).

3. Students should be encouraged to read, study and modify existing (large) source codes, not just write new small programs from scratch. This will give them a better feel for the scale of software engineering and a more realistic view of software engineering as the development and maintenance of existing code rather than always being new implementation work.

   The obvious large source code to use is Unix. This is already done at SCIT with V6 as a paper exercise but needs to be done with BSD4.3 on a real machine.

   It would also be extremely useful if other large source codes could be made available to students such as large application programs like stock control or engineering design calculations packages. These are not readily available from Chinese sources as Chinese industry is not so 'computerised' as the west. The World Bank could help by providing funds to supply such source codes. This would be a relatively small amount of money compared to buying new equipment but would be of great value.

4. The amount of computer science theory taught needs to be increased. Very little is done now. A fourth year option on 'theoretical aspects' should be introduced. This should encompass topics such as grammar theory, computability, recursive function theory, predicate logic, semantics, specification, proof of correctness. As this option gets established it should be used to feed back into the earlier years so that more computer science principles are introduced earlier, preparing the way for the 4th year course to extend its scope as the sophistication of both staff and students increases, in computer science terms, over the next 10 years.

5. SCIT should consider the advantages of teaching Pascal in the first year rather than BASIC (which could be done later in the second year if still considered necessary). The first programming language someone learns affects their development as a programmer more than any subsequent language. Pascal teaches a much better discipline of programming than BASIC and so should be taught first.

6. During the first two years four programming languages are taught (BASIC, Pascal, Fortran, C) but little tuition is given to the common principles of program design, layout, commenting, construction etc. This needs to be built up so that students gain an ability which is independent of programming language. I studied some student programs and they were poorly commented (perhaps the lack of Chinese language features is a hindrance) and poorly laid out. Variable names were badly chosen (being difficult to disambiguate). Again the Chinese/English problem is likely to be the root. The programs I saw were written 'in English' which was in itself a quite impressive accomplishment. Concepts such as 'defensive' programming need to be taught.

7. The idea of students doing a small number of 'experiments', ie practical course work, in teams should be taken up. Real programming and software engineering is always a team effort and the advantages, disadvantages and human/managerial issues of team work need to be experienced by the students. Perhaps one team project per year, of 4 weeks duration would a good initial target. Team size 3-6 members. Assessment by 1) final software product (team), 2) written report by each member on his/her contribution and 3) short interview with staff assessor to verify individual's contribution.

8. Students should be given a short (2 week) course on operating a large computer such as the Honeywell in the Computer Centre. This course should be run by the Computer Centre and should include tuition in basic things such as loading, care and maintenance of magnetic tapes and disks, operation of input/output peripherals and supervision of the whole system via system console operation. Real experience of running a machine, scheduling work, offering a service etc should be communicated to the students. Year 3 would be a suitable point in the course.

9. The CS Dept is quite rightly going to improve its teaching of software engineering by introducing a new course, from September 1987, in Software Engineering and my lectures at SCIT were designed to help formulate this new course. It is very difficult to teach a subject whose problems are all about the scale and complexity of 'industrial' scale software production without the personal experience of these problems and a local industrial infrastructure which exhibits these problems. For industrial experience see recommendation (8.2).

I suggest that at least some of the staff involved with teaching the new software engineering course should be funded by the World Bank to spend a year at a Western University helping an experienced Western lecturer or professor give his software engineering course. This will improve both the experience of the SCIT staff and provide more material for the SCIT course.

3. NEW TECHNICAL EQUIPMENT

1. The present quantity and type of hardware and software tools available to SCIT for teaching and research into programming and software engineering is considerably less than would be considered a viable minimum in the West. With over 350 undergraduates and over 50 staff and postgraduates, a 4-user Honeywell minicomputer plus some IBM PCs and a share of the Institute's Apple II machines is just not enough to enable students majoring in Computer Science to get adequate experimental experience and do enough practical course work. Research in such circumstances has little chance of getting off the ground.

   I suggest that the priority need is for a modern VAX minicomputer, big enough to handle 32 terminals, running the Unix BSD4.3 system. A Unix source code licence is also necessary. Amongst the peripherals should be a small 100 pixels/inch laser printer to give decent quality Chinese character output and graphics capability.

2. The VAX should be equipped with a modem and telephone line for remote diagnostic engineering purposes and to enable the VAX to be connected to the international Unix computer network, called Usenet. This will connect SCIT to the international computing 'grapevine' - a development which would bring many advantages to speed modernisation.

3. Unix should also be purchased to run on several of the larger IBM PCs. This will give a common system amongst disparate machines and reduce the training and documentation overheads. The PCs could then be connected via the UUCP protocols to the VAX enabling access to the larger peripherals, backing store and laser printer for Chinese output. Unix contains TCP/IP which runs over Ethernet and should be of interest to the Local Area Network group.

4. The Computer Centre should investigate the feasibility of running Unix on their Gould Minicomputer. The Computer Centre should gain experience of Unix which is becoming increasingly widespread in Western Universities and computer manufacturers.

5. The CS Dept should obtain the Kerninghan and Plauger Software Tools set in Pascal. These should be mounted on the Honeywell, PCs and Apples to give a common capability across the Department's resources. An excellent text book is available to accompany this software.

4. RESEARCH GOALS

1. SCIT's entry into the Software Engineering field is so recent that little research activity has built up yet. The Department needs to be realistic about what it can achieve in practical terms given its present limited resources.

2. A short term goal should be to undertake a survey of the current state of the art in China relative to the state of the art in the West. The compilation of such a report is now feasible with the flow of SCIT staff and students to the West increasing as are connections with Western engineers thanks to the World Bank scheme.

3. The Department should then use this survey to produce some outline strategy for the next 5 years (analogous to the Chinese national planning system) so that progress against objectives can be assessed.

4. The Department needs to increase its theoretical underpinnings in software engineering so it might be useful to encourage some postgraduate research projects towards theoretical work. This is a) needed and b) not capital intensive and so is feasible immediately. There is considerable interest in concurrency and the resulting theoretical implications and demands in the West. Given the activities of the communications group and the concurrent nature of hardware, theoretical work on concurrency would contribute to the whole department's work.

5. Practical software engineering, as represented by tools and environments, will be expensive to build up to match current Western capabilities. Such thing cannot be bought 'off the shelf'. So SCIT should begin with Unix and the Kerninghan and Plauger Software Tools and from there begin to develop their own capability. To be relevant, such work should be directly related to current industrial practice. So contracts with Chinese industry, to enable tool development to be practically orientated to local needs is probably a better course for the next 5 years rather than simply trying to replicate Western technology.

6. A better graphics software capability needs to be developed to serve as a basis for Chinese output, software tools and CAD. The new ISO standard graphics system, GKS, should be obtained for the Computer Centre's CAD system and for the Departmental Unix systems.

7. Graphics is also a very suitable subject for MSc research projects.

5. FACULTY DEVELOPMENT

1. The CS Department should try to recruit reaching staff from outside SCIT who have strong software backgrounds and qualifications. These could be recent MSc students, like Mr Chen from Wuhan, whose expertise in theorem proving and functional languages is a great asset to the department, or recruits could be drawn from industry to bring real industry and commercial experience into the department.

2. Staff at all levels, and postgraduate students, should be encouraged to travel within China to conferences and meetings so as to stimulate academic interchange of ideas and experience. So far as I can tell, this aspect of Chinese computer science is reasonably well developed under the auspices of the Chinese Computer Society. Internal meetings, conferences and exchanges should continue to be encouraged at all levels.

3. The Chinese Ministry of Education, rather than SCIT, might like to consider an option which exists at many UK universities - that of allowing academic staff to undertake PhD work in their research and spare time. UK members of staff have a special set of regulations covering their work towards a PhD. Normally a UK member of staff, studying only part time, will expect to complete a PhD in 5-6 years as opposed to the 3-4 years for a UK full time student. Given that a Chinese MSc takes 3 years then perhaps a Chinese academic staff already holding an MSc degree could be expected to complete a part time PhD in 3-5 years. Such a 'staff regulations' PhD would help to a) stimulate teaching staff to do more research, b) help to overcome the current difficulty for Chinese academics obtaining PhDs through full time study and c) give an opportunity to older members of staff who due to circumstances beyond their control were unable to study for MSc or PhD degrees when younger.

4. To enable departmental facilities to be used more extensively (ie for more hours/day) and to give greater tuition to the undergraduates, SCIT should consider paying postgraduates an extra amount of money to undertake supervision of undergraduate 'experimental' periods (ie student use of computing facilities and tutorials to help write and debug programs).

6. FURTHER SPECIALISTS TO BE INVITED

1. The new software engineering course at SCIT is scheduled to run from September 1987-August 1988. I would suggest that a university teacher of a similar course at a Western university be invited to SCIT in November 1988 for 4 weeks to help SCIT review the success of the first year of teaching the new course and help develop the course for the subsequent year(s). SCIT staff, from September 1987, should retain all the students' course work results (experiments) and examination questions papers, and results so that the Western teacher can get a good view not just of the course syllabus but of the students performance.

7. LIBRARY FACILITIES

1. The SCIT main library and departmental library have a much smaller stock of computer science and software engineering text books than would be considered adequate in a Western university. Perhaps World Bank funds could help here as text books are relatively expensive to purchase from the West and no doubt the Chinese will not wish to ignore the copyright question in order to save foreign exchange.

2. A reasonable selection of technical journals is available at SCIT although they seem to reach SCIT about 6-12 months later than they are available in the West. Perhaps Departments could purchase them directly from the West instead of having to wait for them to be copied and distributed in China?

3. The library has no access to computer based abstracts services or database search services. These are extremely useful to Western researchers and China is beginning to produce its own versions (I read in China Daily) so perhaps the SCIT library could have the use of a terminal, modem and phone line (that is all that is needed) to firstly gain experience by using the available Western services, so as to be ready to use the emerging Chinese facilities.

4. No facilities exist to handle microfiche. Many technical reports are available in microfiche in the West. It is cheap, convenient, saves storage space and inexpensive. The library should investigate this possibility.

8. SCIT STUDENTS AND STAFF FOREIGN TRAVEL

1. Chinese students are well aware of the advantages of studying for a PhD in the West at the present time. The ones I met did not have enough experience to know that a PhD is not the only sort of training they need (or the West can offer). In an engineering subject like software an academic qualification such as a PhD is only one step towards becoming a software engineer. Without adequate and appropriate practical, industrial experience they will not become good engineers.

2. Both SCIT staff and postgraduates should consider spending time in the West as staff members of Western industrial companies as well as being students. The quality of Chinese students and academics, and the shortage of programming skills in the West, means that jobs of 1-3 years, paying good salaries, are readily available to English speaking graduate programmers.

   MSc, PhD and academic staff should all consider the benefits of working in Western industry. Because industrial companies will pay salaries (greatly in excess of Chinese academic scales!) such an idea will not cost China foreign exchange but will bring valuable training and experience back to China.

3. Efforts by the World Bank and Western Governments to help train Chinese students in the West should grow.

9. CONCLUSION

I was continually impressed by the ability of the staff and students at SCIT to speak and read/write English. This is an extremely encouraging symptom as English is the key technical language for computer science. I was also impressed by the enthusiasm to modernise and progress. The present rate of new building in SCIT and Guangzhou indicates that this enthusiasm is making very concrete progress. So I am optimistic that SCIT will continue to make rapid progress in the teaching and research field of software engineering.

seg 125/seg3