A Digital Ecosystem for ICT Educators, ICT Industry and ICT Students

Kum Leng Chin, Elizabeth Chang, Doug Atkinson
School of IS and DEBI Institute
Curtin University of Technology
kl.chin {e.chang, d.atkinson}@curtin.edu.au

Abstract - Worldwide ICT education (Information Communication Technology) is facing a major challenge of declining student enrolments; battling to keep its curriculum relevant and up-to-date while trying to meet the high demand of ICT skilled workers in domain, such as resources, health, government and commerce. This paper, documenting research in progress, discusses these issues and challenges in ICT education and proposes a solution in the form of a digital ecosystem in ICT education involving three main stakeholders: academics, students and the IT industry, and how they could come together to tackle the problems faced.

Index Terms - ICT education, curriculum development, ontologies, digital ecosystems

I. INTRODUCTION

Recently ICT-related courses in many universities across the globe are experiencing low student enrolments, with some at crisis level resulting in a reduction of staff numbers or even closure of departments [1]. Some are experiencing five years in a row of steady decreasing by approx 15-20% in undergraduate intake. The situation is no different in Australia where universities have suffered a significant declined in student enrolment in ICT-related courses over the past five years [2].

Has the idea of a career in Information Communication Technology (ICT) lost its attraction for young people? A recent study found that most teenagers perceive computing as boring, antisocial and irrelevant to their lives [3]. Another study stated that 93% of the 126 respondents of university freshmen who had not chosen a major area of study declared that they will choose an area other than computing [4].

Meantime, not only the ICT industry, but also the resources sector, such as oil and gas, the health sector, government and businesses are facing the increasing challenge of a skills shortage. Not just because of the declining number of ICT graduates but also the skill base of the graduates. Numerous studies conclude that MIS graduates (Management of Information System, a new discipline that arose in the last decade) are generally lacking the quality in skills that are of most interests to employers [5]. As the start of the twenty-first century sees the overall economies viability of regions, states and countries link directly to the viability of technology-enabled, knowledge-intensive sectors; it is vital to ensure the availability of an IT talent pool be maintained [6].

Obviously, the skills shortage problem in the ICT sector is a complex one and may require multiple strategies to tackle the problem. However, one must not deny that the curriculum content plays an important role in attracting students to study a particular course. The design of any ICT-related curriculum should take into consideration the needs and interests of young people (students) in order to attract them to enrol and study in a course. It is also vital that any ICT-related curriculum stay relevant to industry practise to ensure graduates equipped with the right knowledge and skills are set to fill job positions.

To achieve the above, ICT education curriculum design and development plays a key role in the next wave of ICT education. ICT education should never been treated as an academic exercise but rather garner feedback from industry and graduates regarding gaps in the units, courses and degrees provided by the academics. Academics know that the ICT curriculum should be carried out with the involvement of students and industry representatives; however, this is easier to say than do. Many have tried to incorporate such input but little impact has been achieved towards the shift to students’ and industry’s demand. Despite that, many suggestions and approaches have been proposed such as how the contribution from external parties needs to be ongoing to be of value and how the support of appropriate technology, infrastructure and organisation is required to make it possible.

This paper presents Part I of our detailed analysis of the ICT courses, targeted on computer science, and proposes a digital ecosystem for ICT education where multiple stakeholders are involved to attain certain goals. It is hoped that the outcome of such a system will serve all stakeholders better and resolve some of the problems faced by university ICT-related courses and the IT industry. This paper will start out by examining some of the issues surrounding ICT education with a detailed explanation of the Australian Computer Science Curriculum patterns, follow by explaining why a digital ecosystem for curriculum development may be considered.

II. ICT AND ICT EDUCATION

There are different definitions of ICT that are being used. In the last decade, ICT was derived from the two fields of studies, namely Information Technologies (IT) and Communication Technologies (CT).

ICT is defined as an 'umbrella' [7] that covers many aspects of computing, information management and communication networks, ranging from developing and installing applications to engineering or designing complex computer systems and networks including information databases. The duties of ICT professionals are typically expected to be expert in data management, network management, computer hardware and software, application and system management.
Thus, ICT branches out into several disciplines and fields of study including those such as Computing, Computer Science (CS), Computer Engineering (CE), Computer System Engineering (CSE), Software Engineering (SE), Information Technology (IT), Information Systems (IS), Management of Information Systems (MIS), Business Information Technology (BIT), and the like. New branches are continually added such as E-Commerce (EC), Information Science, Bio-Informatics, Geo-Information Systems (GIS), Digital Engineering and BioTech and so forth. We also note that many major disciplines have been slowly shrinking to subject areas rather than disciplines including Artificial Intelligence (AI) and Computer Networks (CN).

ICT education is defined as the courses that are in one or more of the above areas of study at a tertiary level such as those offered in universities. Students who are enrolled in these courses are taught to become ICT professionals upon graduation.

III. BROADER ISSUES IN ICT EDUCATION

There are currently a number of broader issues surrounding ICT education in Australia.

Firstly, all ICT related disciplines are suffering an identity crisis. Being a relatively new discipline that has emerged over the last thirty years for CS, CE, SE, IT, IS, MIS or BIT and so forth, as well as the boom of new ICT disciplines, such as Information Science, GIS, XX-Informatics (such as Bio, Industrial etc.), many academics, students and industry employers were unable to distinguish the difference between the fields and believe that they are the same thing. Many people will just simply understand that ICT is related to working with computers. Such a simplified understanding coupled with many assumptions such as ‘computer nerd’ or ‘anti-social behaviours’, or ‘lack of business acumen’ and the like, did not served to enhance the reputation that these new disciplines afforded comprehensive knowledge and skill training.

Secondly, confusion exists in the subject area within each discipline. When coming to select a degree, many of the courses offered within the ICT related disciplines do not offer a student a clear indication of what a degree in CS, EC, IT, IS or MIS, BIT will lead to in terms of their targeted profession or employment area; where as the clarity of one choosing to study a law degree is vastly different. If one studies a medical degree, it is assumed that one can later work as a doctor. But where does an ICT degree lead? Some courses from one university’s Computer Science degree or IS degree are completely different from another university’s that uses the same named degrees. Same courses offering courses of different names, such as an II or IS courses, look just like a CS degree or SE degree. This adds to the confusion of many prospective students when not even the ‘expert’ can tell you the difference between them [4]. Further confusion arises when a quick look at these courses shows that some are run by Science Faculties/Schools, some by the Engineering Faculty or School, some are within the Faculty of Business and a few are independent of all faculties.

Thirdly, classification of the ICT discipline as a science, engineering or business only is flawed. It is noted that many universities classify the ICT degree as either science, engineering or business whereas the ICT discipline is a multi-disciplinary field and cannot stand-alone. No university clearly classifies ICT as a multi-disciplinary field nor teaches it in such a way. As a result, prospective students do not know what the course of study involves – does it require mathematics? Would it be alright for me to do a science degree without a mathematics background and to still become a Computer Scientist or Computer Engineer? Or will it be possible to be a CEO of a dot.com without business studies? In addition, the burst of dot.com worldwide at the beginning of the millennium and more recently the widespread report of ICT being outsourced to many developing countries did not help to paint a bright future the pure science, engineering or business based ICT degrees. All in all, it is no wonder that student enrolment numbers in many universities across Australia have dropped significantly in the past few years.

Forthly, more and more ICT graduates are not well equipped with the skills and knowledge needed by today’s industry. Over the last couple of years, we see an increasing demand of IT skilled workers in all domains of industry and government organisations. Some fear that the situation may worsen to the point of affecting productivity of the sector and consequently impacting the country’s economy [4]. Some belief that ICT functions have been outsourced to countries with cheaper labour or regional areas, therefore, the major industries in major cities will not need ICT workers. In fact, this is not true and some industries will never be able to do this. Recently, the Australian government is allowing a higher number of immigrants with ICT qualifications to live in Australia. However, this may not solve the skills shortage problem in the long run. Ultimately, the solution lies in being able to attract more young people to take up a career in ICT areas. In 2007, the Australia Research Council funded more than twice as much to the ICT education with the model proposed by the ACM/IEEE model curriculum, see Table 1.

IV. THE COMPARATIVE STUDIES BETWEEN ACM/IEEE COMPUTER CURRICULUM AND AUSTRALIAN ICT EDUCATION

In this section, we compare the current Australian ICT education with the model proposed by the Computing Curriculum (mainly for Computer Science) developed by the Joint Curriculum Task Force of the ACM and the IEEE Computer Society [8].

Choice of data and approach:

(1) We only address five ICT discipline areas for this comparative study namely: Computer Engineering (CE), Computer Science (CS), Information Systems (IS), Information Technologies (IT), and Software Engineering (SE) as these five disciplines were chosen in the ACM/IEEE model curriculum, see Table 1.

(2) We have chosen twenty-five Australian universities for the comparative study as they have ACS
The listed Comparison Table 1 and 2 are Part 1 of
some universities have a degree named Bachelor of
Computing Model Curriculum. On the right, five major
degrees in 5 columns, namely CE, CS, IS, IT and SE are
shown. The intersection of each row and column of the
table lists the weight of each core knowledge area that
should be addressed in each discipline area. The weight is
between 1 and 5. A weighting of five means it holds the
‘Highest emphasis’ and 0 means ‘Least emphasis’. For
each knowledge topic (row), in each discipline (column),
there is a minimum level of emphasis and a maximum
level of emphasis, to indicate the importance of the topic
covered in that discipline.

Here, we use the following notations to help our
discussion:

- A weight assigned either 5-5 or 4-5 represents a 5* importance
- The weight assigned 3-5 represents a 4* importance
- The weight ranges of 2-5, 3-4 and 2-4 represent a 3* importance
- The weight below those levels represent a 2* and
below 2* which indicates that the topic is not too important

Table 2(a) shows the coverage of the ACM/IEEE Computing Model Curriculum (with the weight indicating the importance to CS, Computer Science programmes) by the CS programmes of Australian universities. A ‘tick’ shows that the topic is covered. If there is no ‘tick’, it implies a topic is not being covered by the CS programme of that university.

At the bottom of the rows, we have added three popular topics that are covered in many Australia computer science degrees but not in the ACM/IEEE Computing Model Curriculum.

A number of observations can be made from this study when mapping the ACM/IEEE Computing Model Curriculum:

- 2 out of 40 topics are the most important topics. Here, we classify it as 5* because they carry a weight of between 4 and 5 for CS as ranked by ACM/IEEE namely: Programming fundamentals and Algorithms are covered by all 25 universities in Australia.
- 3 out of 40 topics are of the second greatest importance. Here we define them as 4* because they carry a weight of between 3 and 5. Only 60% of Australian universities cover them, namely: Operating System, Theory of Programming language, and Software Design.
- 10 out of 40 topics are ranked lower. Here, we classify them as 3* as they are weighted between 2 and 5 and are covered by an average of 60% Australian universities (see table 2(a)).
- 25 other topics are ranked below 3* because they carry a weight between 0 and 4 and are covered by
an average of less than 25% of the Australian universities. However, two out of these lower ranked
25 topics namely: Information Management/DB practice and Foundation of Software Engineering are
highly covered by Australian universities at a rate above 80%.
- Three topics are not in the ACM/IEEE computing model curriculum but are frequently taught in CS degrees including Project Management, Internet/Web Technologies, and Computer Gaming, see the bottom of the Table 2(a).
- Overall, the comparison demonstrates that over 70% are in agreement with the top five topics as ranked by ACM/IEEE and about 50% or less show a
mismatch of what should be included in CS at Australia universities according to the ACM/IEEE defined computing model curriculum.

• Other disciplines such as Information Systems (IS), Software Engineering and the like have demonstrated a similar trend. That is, there is a general agreement of what should be taught as core topics (usually 4-5 of the core) and over a 50% mismatch seen in the other units or electives.

Table 3(a) shows the coverage of the 40 ACM/IEEE Computing Model Curriculum (with the weight indicating the importance to SE, Software Engineering programme) by the SE programmes of Australian Universities. A ‘tick’ shows that the topic is covered. If there is no ‘tick’, it implies a topic is not being covered by the SE programme of that University.

On the bottom rows, we add one of most popular topics that are covered in many Australia computer science degrees but not in the ACM/IEEE Computing Model Curriculum.

A. Mapping the ACM/IEEE Computing Model Curriculum

We make the following observations:

• 4 out of 40 topics are the most important. Here, we define them as 5* because they have a weighting of between 4 and 5 for SE as ranked by ACM/IEEE namely:
  - Programming Fundamentals
  - Software Modelling and Analysis
  - Software Design
  - Software Verification and Validation
90% are covered by all 22 universities in Australia. These 22 Universities have ACS accredited SE programmes.

• 2 out of 40 topics are of second importance carrying a 4* rating due to a weighting of between 3 and 5, namely: Human-Computer Interaction and Analysis of Technical Requirements
These are covered by about 35% of Australian universities.

• 13 out of 40 topics are ranked as 3* topics for SE, with a weighting range between 2-5, 3-4 and 2-4. These are covered by an average of 60% of Australian universities (see table 3(a)).

• 21 other topics ranked as 2* or below 2* (because they have weight of emphases between 0-4) are covered an average of less than 25% by Australian Universities. However, 1 f these lower rank 21 topics namely: Information Management/DB practice is covered highly by Australian Universities, at 70%.

• 1 topic that is not in the ACM/IEEE computing model curriculum but is frequently taught in SE degrees - Project Management as seen at the bottom of Table 3(a). Note that Internet/Web Technology is also a popular topic that is widely taught in Australian universities.

• Overall, the comparison demonstrates that there is 90% agreement about the core topics (top 4 topics ranked by ACM/IEEE) and about 50% or less in the mismatch of what should be included in SE programmes at Australia universities compared to what ACM/IEEE defined computing model curriculum.

• Our other discipline studies such as Information Systems (IS), have demonstrated the similar trend, that is, there is a general agreement about what should be taught as core topics (usually 4-5 of the core) and over 50% mismatch in the other units or electives occurs.

B. Australian University Course Comparison

In our study we found that:

• Many units offered as ‘introductory’ which cover a bit of everything related to computing, seem a little too basic and could be learnt from the internet or self taught within two weeks.

• Some CS or SE degrees offer more than one teaching units/subjects/topics in the same knowledge area, hence, redundancy or overlapping between the units exist. This also causes less coverage of other areas.

• Some CS or SE degrees offer several units that cover multiple knowledge area but not in much depth. For example, an Information Management unit could cover data warehousing, data mining and knowledge discovery.

• A few courses cover cutting edge technologies such as XML, web services, ontologies and the semantic web, computer gaming and the like.

• Project Management is a very popular subject in all Australian universities and in all related ICT fields of study. However, it is not part of the ACM/IEEE computing model curriculum.

Our study showed that worldwide educators are only in agreement with less than 15% of all units that uniquely identify each of the ICT disciplines. It is not surprising that ICT educations are facing issues and confusion.

Michael Brodie [9] indicated that there were 161 exabyte of information created/replicated in 2006 alone, and that it was greater than the last 5000 years of data all together and it is estimated there will be six times growth by 2010 and after, double every two years. With rapid growth of web based interaction, the Tsunami of data exabyte of information created/replicated in 2006 alone, and that it was greater than the last 5000 years of data all together and it is estimated there will be six times growth by 2010 and after, double every two years. With rapid growth of web based interaction, the Tsunami of data per year. How can we create a dynamic ICT curriculum that addresses the need for our future graduates to survive in the digital world? All academics understand that a curriculum development involves mainly three stakeholders: academics, students and the IT industry. However, very few groups can demonstrate the success of the involvement of the three parties resulting in cutting edge ICT programmes.

VI. A DIGITAL ECOSYSTEM APPROACH TO ICT CURRICULUM DEVELOPMENT

Boley and Chang [11] stated that ‘an ecosystem is a loosely coupled, domain clustered environment inhabited by species, each proactive and responsive regarding its own benefit while conserving the environment’.
To consider the IT education curriculum development ecosystem, we first need to identify its inhabited 'species' – academics, students and the IT industry in this case. There may be other species within the system but for now we will just consider these three main ones. Each of these species can contribute and want to benefit from the system in its own way.

Academics design and teach courses as well as carry out research in their chosen field. However, without students, academics will not be needed. Academics can contribute to the design and development as knowledge experts in the field but also find their experiences in pedagogical issues. Traditionally, academics are entrusted with the task of curriculum design and development; however, being in a field as dynamic as IT, input from other perspectives will certainly help keep the balance needed for sustainability.

Students study and learn from the course. Students will engage better in learning if they believe the course can help them step into their desired career. Students will also learn better if the course is interesting and enjoyable. Students can help to improve the outcome of curriculum by voicing their concerns, learning needs and expectations. However, students need to make a commitment to the course by first enrolling and subsequently staying in the course until graduation.

Industry employs graduates and also occasionally offers scholarships or sponsorships for high performing students. Industry also offers funding for research projects from time to time. The industry expects employees to equip themselves with certain skills and knowledge so as to fill job positions and contribute to the productivity of the IT industry. Apart from knowledge and skill sets, industry is also concerned with competency as different job roles in the same area require different levels of competency in the same knowledge and skill domain. However, industry needs to communicate their needs to academics and students. They also need to be supportive of academic activities and be involved in the educational process.

Ideally, within an IT education curriculum development system these stakeholders should work together, in continuing dialogue to ensure IT education curricula stay up-to-date, attractive and relevant. However, in reality, due to resource constraints, many curricula development were carried out mainly by academics with the occasional input from students, possibly in the form of a survey; and equally limited contribution from the IT industry, possibly in the form of an advisory committee that meet only once or twice a year.

Species within an ecosystem interact and engage amongst themselves for survival and also for the betterment of their environment. The proposed ICT education curriculum development digital ecosystem will use an ontology for ‘ICT knowledge organisation’ as the backbone with its evolutionary nature to enable continuing contribution from multiple stakeholders via the web. Applications via ontological views allow access to the ontology for various purposes such as:

- Enquiry tools for the IT industry to determine which course is producing graduates with what knowledge and skill sets they desire.

![Layered Ontological representation of a Pedagogical System](image)

In general, a course may be developed through top-down or a bottom-up approach. (Note that in Australia, a 'course' is equivalent to a 'degree' in the USA). Normally, when a course is created, subjects within a course must be defined followed by detailed topics within each subject. However, there are several problems that have to be considered including:

- One subject may have different names across different universities or the same name may be used for subjects having totally different content;
- A subject may not cover all the topics that it should or be equivalent to subjects covered at other universities; or
- A subject may not align with what industry requires.

Therefore, we propose a layered ontology view of a Pedagogical System that can be applied to any IT-related curriculum, starting at the concept level, then organised into subject level, and finally into a course level, so that we produce a layered ontology as depicted in Figure 1.

Figure 1 shows that a curriculum is developed by carefully defining ‘Concepts’ and grouping them into ‘Topics’. The grouping of ‘Topics’ forms ‘Subject’ areas, and grouping and sequencing of ‘Subjects’ leads to ‘Courses’. Detailed design of the set of properties of each element within the ontology will take into consideration the three main stakeholders - academics, students and industry - in order to ensure that the ontology can satisfy queries from all three perspectives.

Another concept that will be defined in the ontology is ‘Competency’. For example, what expectations does industry have for a software engineering professional, a computer science professional, or an information systems' professional? This study will adopt, as a foundation, the Skills Framework for the Information Age (SFIA) established by the British Computer Society (BCS) [12]. Competency related to the IT field will be tentatively modelled into:

1) Category and Subcategory - in which skills are grouped into categories and subcategories describing broad areas of work;
2) Level - which represents the degree of responsibility that an IT practitioner exercises; and
3) Skill - showing a recognisable area of IT competence within the workplace.
Figure 2 demonstrates a conceptual view of the mapping between the competencies and ontological hierarchy of the IT-related pedagogical system. We map between competencies at the concept level, topic level, subject level, or course level. This involves:

a) grouping the knowledge at concept, topic, subject, and course level, taking note that the required knowledge may be satisfied by a combination of concepts or topics or subjects or even courses;

b) sequencing the concept, topic, and subject deliveries where each of these could be covered in various depth such as introductory, intermediate or advanced level; and

c) defining prerequisites between courses, subjects, topics, and concepts to ensure the integrity and quality of any curriculum based on the ontology.

Competency involves broader issues, including technical issues, oral and physical presentation issues, feasibility studies and the like. For example, an IT professional must be capable of presenting a case explaining the need for and intricacies of the development of a particular system, and must also be capable of carrying out feasibility analysis and cost benefit analysis. This requires knowledge of several topics. One should be able to group such topics to identify a subject.

For some subjects the title is irrelevant; certain common topics always seem to be present. Therefore, the mapping of competencies and topic areas for these subjects is commonly accepted.

A number of applications can be developed accessing the ontology to serve different purposes:

- An application to help prospective students to choose the right course for their desire career choice. The application should enable individual student to design their study plan matching their study objectives that reflect the skills and knowledge needed for specific career path.

- An application for IT industry to match the quality of graduates from each study programme with the desirable quality they are seeking

- An application for IT industry to define new job roles or revising existing ones by specifying the knowledge, skills and level of competency needed.

An Ontology View mechanism [13] will be included to provide three levels of views of varying granularity: a conceptual level view that allows users to view at a high level abstraction by specifying a constraint of their view; a logical view that provides a view of a subset of the ontology (sub-ontology) from the IT Pedagogical Knowledge Framework; and a physical level view that allows educators to customise the sub-ontology to their specific educational needs.

It is envisaged that the three main stakeholders: academic, student and the IT industry will benefit from the proposed IT Education Curriculum Development Digital Ecosystem.

VII. CONCLUSION AND FUTURE WORK

This paper outlines ICT education issues and uses CS as a case study to demonstrate the issues that are crucial to addressing today’s fast growing digital world, digital economic and digital ecosystems. A proposed ontology based digital ecosystem for ICT curriculum development to promote knowledge sharing and collaboration among the main stakeholder; namely academics, students and the IT industry to help keep pace with rapid web development. With the help of advanced web technologies, particularly the semantic web [14], and the capability of ontology, practical and resource constraints can be overcome to ensure the sustainability of the system. As indicated earlier, this study is part of a PhD research work. We understand the data collection and some conclusions may be too narrow or limited. We are continuing work on it. It is envisaged that we have many conceptual and technical challenges ahead of us. However, we hope that the outcome of this study may provide some relief and motivation for moving ICT education to a new level.
VIII. REFERENCES


12. BCS: http://www.bcs.org/


### Table 1: ACM / IEEE Computing Curricula

**Computing Curricula 2005**

<table>
<thead>
<tr>
<th>Knowledge Area</th>
<th>CE</th>
<th>CS</th>
<th>IS</th>
<th>IT</th>
<th>SE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>min</td>
<td>max</td>
<td>min</td>
<td>max</td>
<td>min</td>
</tr>
<tr>
<td>Programming Fundamentals</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Integrative Programming</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Algorithms and Complexity</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Computer Architecture and Organisation</td>
<td>5</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>Operating Systems Principles and Design</td>
<td>2</td>
<td>5</td>
<td>3</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Operating Systems Configuration and Use</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Net Centric Principles and Design</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>Net Centric Use and Configuration</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Platform Technologies</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Theory of Programming Languages</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Human-Computer Interaction</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Graphics and Visualisation</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Intelligent Systems (AI)</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Information Management (DB) Theory</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Information Management (DB) Practice</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Scientific computing (Numerical methods)</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Legal / Professional / Ethics / Society</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Information Systems Development</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>Analysis of Business Requirements</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>E-business</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>Analysis of Technical Requirements</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Engineering Foundations for SW</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Engineering Economics for SW</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Software Modelling and Analysis</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Software Design</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Software Verification and Validation</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Software Evolution (Maintenance)</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Software Process</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Software Quality</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Comp Systems Engineering</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Digital Logic</td>
<td>5</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Embedded Systems</td>
<td>2</td>
<td>5</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Distributed Systems</td>
<td>3</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Security: Issues and Principles</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Security: Implementation and Mgt</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Systems Administration</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Management of Info Systems Org.</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>Systems Integration</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Digital Media Development</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Technical Support</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

0 – Least emphasis  
5 – Highest emphasis

Table 1: ACM/IEEE Computing Curricula for CE, CS, IS, IT and SE
<table>
<thead>
<tr>
<th>Knowledge Area</th>
<th>Min</th>
<th>max</th>
<th>Cardiff University</th>
<th>Deakin University</th>
<th>Edith Cowan</th>
<th>Flinders University</th>
<th>James Cook University</th>
<th>La Trobe University</th>
<th>Macquarie University</th>
<th>Monash University</th>
<th>University of Adelaide</th>
<th>University of Ballarat</th>
<th>University of New England</th>
<th>University of Newcastle</th>
<th>University of Queensland</th>
<th>University of South Australia</th>
<th>University of Sydney</th>
<th>University of Western Australia</th>
<th>University of Wollongong</th>
<th>Victoria University</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming Fundamentals</td>
<td>4</td>
<td>5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Integrative Programming</td>
<td>1</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Algorithms and Complexity</td>
<td>4</td>
<td>5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Computer Architecture and Organisation</td>
<td>2</td>
<td>4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Operating Systems Principles &amp; Design</td>
<td>3</td>
<td>5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Operating Systems Configuration &amp; Use</td>
<td>2</td>
<td>4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Net Centric Principles and Design</td>
<td>2</td>
<td>4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Net Centric Use and Configuration</td>
<td>2</td>
<td>3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Platform Technologies</td>
<td>0</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Theory of Programming Languages</td>
<td>3</td>
<td>5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Human-Computer Interaction</td>
<td>2</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Graphics and Visualisation</td>
<td>1</td>
<td>5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Intelligent Systems (AI)</td>
<td>2</td>
<td>5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Information Management (DB) Theory</td>
<td>2</td>
<td>5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Information Management (DB) Practice</td>
<td>1</td>
<td>4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Scientific Computing (Numerical Methods)</td>
<td>0</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Legal / Professional / Ethics / Society</td>
<td>2</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Information Systems Development</td>
<td>0</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
### Table 2(a) Coverage of knowledge areas by Australian CS programmes

Knowledge areas that are highlighted in yellow are those with higher emphasis according to the model curricula. The last three which are highlighted in blue were added by the authors but are not in the ACM/IEEE CS model curriculum but they are a popular subject/ unit in Australian CS programmes.
Table 2(b) – Australian Universities with ACS Accredited Computer Science Courses

1. Curtin University of Technology
   Department of Computing
   Bachelor of Science (Computer Science)

2. Deakin University
   School of Information Technology
   Bachelor of Information Technology (Computer Science and Software Development)
   *** Comp Science single major was indicated in ACS website but not found in the university website. The double major computer science and software development does not constitute the expected comp sci content, therefore, will not be included in this analysis.

3. Edith Cowan University
   School of Computer and Information Science
   Bachelor of Science (Computer Science)
   *** Communicating in IT environment: information sources and info retrieval

4. Flinders University
   School of Informatics and Engineering
   Bachelor of Science (Computer Science)
   *** Information Communication Technology: to provide students with a consolidated knowledge of the basic fundamentals, key concepts and terminology of computer science and information technology
   *** Data Mining and Knowledge Discovery

5. Griffith University
   School of Information & Communication Technology
   Bachelor of Information Technology (Computer Science)
   *** Computer Science major is no longer being offered as from 2008

6. James Cook University
   School of Information Technology
   Bachelor of Science (Computer Science)

7. La Trobe University
   School of Engineering and Mathematical Sciences
   Bachelor of Computer Science

8. Macquarie University
   Department of Computing
   Bachelor of Computer Science

9. Monash University
   Faculty of Information Technology
   Bachelor of Computer Science 2010
   Bachelor of Computing (Computer Science) 2006
   Bachelor of Science (Computer Science) 2006

10. Murdoch University
    School of Information & Technology
    Bachelor of Science (Computer Science)
    Bachelor of Science (Computer Science)

11. Queensland University of Technology
    Faculty of Information Technology
    Bachelor of Information Technology (Software Engineering)
    *** Moved to Software Engineering list

12. RMIT University
    School of Computer Science and Information Technology
    Bachelor of Applied Science (Computer Science)
    Bachelor of Computer Science [specialisation]

13. Swinburne University of Technology
    Faculty of Information & Communication Technologies
    Bachelor of Science (Computer Science & Software Engineering)
    *** Engineering based

14. University of Adelaide

15. University of Ballarat
    School of Information Technology and Mathematical Sciences
    Bachelor of Applied Computing

16. University of Melbourne
    Department of Computer Science and Software Engineering
    Bachelor of Computer Science
    *** Some of the knowledge areas are covered by ‘CS electives’

17. University of New England
    School of Mathematics Statistics & Computer Science
    Bachelor of Computer Science
    *** Only the core topics were analysed, major units were not included as there are 7 different majors to choose from

18. University of New South Wales
    School of Computer Science & Engineering
    Bachelor of Science (Computer Science)

19. University of Newcastle
    School of Electrical Engineering & Computer Science
    Bachelor of Computer Science

20. University of Queensland
    School of Information Technology and Electronic Engineering
    Bachelor of Science (Computer Science)

21. University of South Australia
    Department of Mathematics and Computing
    Bachelor of Information Technology (Applied Computer Science)

22. University of Southern Queensland
    School of Information Technologies
    Bachelor of Computer Science and Technology
    *** Website listed all IT/CS/SE units together and allow students to pick their choices

23. University of Sydney
    School of Information Technologies
    Bachelor of Computer Science and Technology
    *** Website listed all IT/CS/SE units together and allow students to pick their choices

24. University of Tasmania
    School of Computing
    Bachelor of Computing
    Bachelor of Science (Computing)

25. University of Western Australia
    School of Computer Science and Software Engineering
    Bachelor of Computer and Mathematical Sciences (Computer Science)
    Bachelor of Computer Science
    Bachelor of Science (Computer Science)

26. University of Western Sydney
    School of Information Technology
    Bachelor of Computer Science

27. University of Wollongong
    School of Information Technology and Computer Science
    Bachelor of Computer Science

28. Victoria University
    School of Computer Science and Mathematics
    Bachelor of Science in Computer Science
Table 3(a) Comparison between ACM/IEEE's Software Engineering (SE) curriculum and Aust SE curriculum

| Knowledge Area                              | min | max | Curtin University | Edith Cowan University | Flinders University | Griffith University | La Trobe University | Monash University | Queensland University of Technology | RMIT University | Swinburne University | University of Adelaide | University of Melbourne | University of Newcastle | University of Queensland | University of South Australia | University of Sydney | University of Technology, Sydney | University of Western Australia | University of Wollongong |
|---------------------------------------------|-----|-----|-------------------|------------------------|-------------------|--------------------|--------------------|-------------------|-------------------------------|------------------|-----------------------|------------------------|------------------------|--------------------------|-----------------------|------------------------|-------------------------------|------------------|-----------------------|------------------------|------------------------|
| Programming Fundamentals                   | 5   | 5   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Integrative Programming                    | 1   | 3   |                   |                        |                   |                    |                    |                   |                               |                  |                        |                        |                        |                          |                        |                        |                               |                  |                        |                        |                        |
| Algorithms and Complexity                  | 3   | 4   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Computer Architecture and Organisation    | 2   | 4   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Operating Systems Principles and Design    | 3   | 4   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Operating Systems Configuration and Use    | 2   | 4   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Net Centric Principles and Design          | 2   | 4   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Net Centric Use and Configuration          | 2   | 3   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Platform Technologies                      | 0   | 3   |                   |                        |                   |                    |                    |                   |                               |                  |                        |                        |                        |                          |                        |                        |                               |                  |                        |                        |                        |
| Theory of Programming Languages            | 2   | 4   |                   |                        |                   |                    |                    |                   |                               |                  |                        |                        |                        |                          |                        |                        |                               |                  |                        |                        |                        |
| Human-Computer Interaction                 | 3   | 5   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Graphics and Visualisation                 | 1   | 3   |                   |                        |                   |                    |                    |                   |                               |                  |                        |                        |                        |                          |                        |                        |                               |                  |                        |                        |                        |
| Intelligent Systems (AI)                   | 0   | 0   |                   |                        |                   |                    |                    |                   |                               |                  |                        |                        |                        |                          |                        |                        |                               |                  |                        |                        |                        |
| Information Management (DB) Theory         | 2   | 5   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Information Management (DB) Practice       | 1   | 4   |                   |                        |                   |                    |                    |                   |                               |                  |                        |                        |                        |                          |                        |                        |                               |                  |                        |                        |                        |
| Scientific Computing (Numerical Methods)   | 0   | 0   |                   |                        |                   |                    |                    |                   |                               |                  |                        |                        |                        |                          |                        |                        |                               |                  |                        |                        |                        |
| Legal / Professional / Ethics / Society    | 2   | 5   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Information Systems Development            | 2   | 4   |                   |                        |                   |                    |                    |                   |                               |                  |                        |                        |                        |                          |                        |                        |                               |                  |                        |                        |                        |
| Analysis of Business Requirements          | 1   | 3   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| E-business                                 | 0   | 3   |                   |                        |                   |                    |                    |                   |                               |                  |                        |                        |                        |                          |                        |                        |                               |                  |                        |                        |                        |
| Analysis of Technical Requirements         | 3   | 5   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Engineering Foundations for SW             | 2   | 5   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Engineering Economics for SW               | 2   | 3   |                   |                        |                   |                    |                    |                   |                               |                  |                        |                        |                        |                          |                        |                        |                               |                  |                        |                        |                        |
| Software Modelling and Analysis            | 4   | 5   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
| Software Design                            | 5   | 5   | √                 | √                      | √                 | √                  | √                  | √                 | √                             | √                | √                      | √                      | √                      | √                        | √                      | √                      | √                             | √                | √                      | √                      | √                      |
Table 3(a): Coverage of knowledge areas by Australian university SE programmes. Knowledge areas that are highlighted in yellow are those with higher emphasis according the model curricula. The last, project management which is highlighted in blue, was added by the authors but not in the ACM/IEEE SE model curriculum but is popular in Australian SE programmes.
Table 3(b) – Australia Universities with ACS Accredited Software Engineering Courses

1. Curtin University of Technology  
   Bachelor of Science (Software Engineering)  
   Bachelor of Engineering (Software Engineering)
2. Edith Cowan University  
   School of Computer and Information Science  
   Bachelor of Science (Software Engineering)
3. Flinders University  
   School of Informatics and Engineering  
   Bachelor of Engineering (Software Engineering)
4. Griffith University  
   School of Information & Communication Technology  
   Bachelor of Engineering (Software Engineering)  
   Bachelor of Information Technology (Software Engineering)
5. La Trobe University  
   School of Engineering and Mathematical Sciences  
   Bachelor of Software Engineering
6. Monash University  
   Faculty of Information Technology  
   Bachelor of Software Engineering
7. Murdoch University  
   School of Engineering Science  
   Bachelor of Engineering (Software Engineering)
8. Queensland University of Technology  
   Faculty of Information Technology  
   Bachelor of Information Technology (Software Engineering)
9. RMIT University  
   School of Computer Science and Information Technology  
   Bachelor of Applied Science (Software Engineering)
10. Swinburne University of Technology  
    Faculty of Information & Communication Technologies  
    Bachelor of Software Engineering
11. University of Adelaide  
    School of Computer Science  
    Bachelor of Computer Science (Software Engineering)
12. University of Canberra  
    School of Information Sciences and Engineering  
    Bachelor of Engineering with a specialisation in Software Engineering
13. University of Melbourne  
    Department of Computer Science and Software Engineering  
    Bachelor of Engineering (Software Engineering)
14. University of New South Wales  
    School of Computer Science & Engineering  
    Bachelor of Engineering (Software Engineering)  
    *** Workshop based approach to teaching
15. University of Newcastle  
    School of Electrical Engineering & Computer Science  
    Bachelor of Engineering (Software Engineering)
16. University of Queensland  
    School of Information Technology and Electronic Engineering  
    Bachelor of Engineering (Software Engineering)
17. University of South Australia  
    School of Computer and Information Science  
    Bachelor of Information Technology (Software Engineering)  
    Bachelor of Software Engineering
18. University of Southern Queensland  
    Department of Mathematics and Computing  
    Bachelor of Information Technology (Software Engineering)
19. University of Sydney  
    School of Electrical and Information Engineering  
    Bachelor of Engineering (Software Engineering)
20. University of Technology Sydney  
    Faculty of Engineering  
    Bachelor of Engineering (Software Engineering)  
    Bachelor of Engineering Science (Software Engineering)  
    Faculty of IT, Department of Software Engineering (the department does not offer direct degree in SE, but in IT)
21. University of Western Australia  
    School of Computer Science and Software Engineering  
    Bachelor of Engineering (Software)
22. University of Wollongong  
    School of Information Technology and Computer Science  
    Bachelor of Information and Communication Technology (Software Engineering)