

School of Computer Science

Undergraduate Information for McGill Students 2009 - 2010

Computer Science covers the theory and practice behind the design and implementation of computer and information systems. Fundamental to computer science are questions how to describe, process, manage, and analyze information and computation. A fundamental building block is the study of **algorithms**. An algorithm presents a detailed sequence of actions solving a particular task. A **computer program** is the implementation of an algorithm in a specific programming language so that a computer can execute the algorithm. **Software** generally refers to a computer program or a set of related computer programs.

Based on the building blocks of algorithms and programs, computer science is split into many different areas such as the study of **algorithms and data structures**, **programming languages** and methodology, **theory of computation**, **software engineering** (the design of large software systems), **computer architecture** (the structure of the hardware), **communication** between computers, **operating systems** (the software that shields users from the underlying hardware), **database systems** (software that handles large amounts of data efficiently), **artificial intelligence** (algorithms that imitate human information processing), **computer vision** (algorithms that let computers see and recognize their environment), **computer graphics**, **robotics** (algorithms that control robots), and **computational biology** (algorithms and *methods that address problems inspired by biology*). **Computer science also plays an important role in many other fields**, including Biology, Physics, Engineering, Business, Music, and Neuroscience, where it is necessary to process and reason about large amounts of data.

Computer Science is strongly related to mathematics, linguistics and engineering. In contrast to traditional fields of science that aim in understanding the natural world, computer science advances our understanding of information and computation, and applies science to create algorithms and software that can be executed on computers.

A degree in computer science offers excellent job prospects. As the use of computers and specialized software has played an important role in any area of business, science and our personal life, computer graduates are in high demand. Computer scientists find *jobs in software development in many areas of computer science, in consulting, and in project management*. As computer scientists often develop the software for a specific application domain (e.g., business, engineering, medicine), computer scientists must be prepared and willing to get to know their application area.

The School of Computer Science offers a wide range of programs. Most programs start with the same set of basic courses allowing students to decide on their exact program once they got a basic understanding of the discipline. There are a major, a honours, a liberal and a minor program in Computer Science, a major and a liberal program in Software Engineering, a major in Computer Science: Computer Games Option, a minor in Computational Biology, a joint major and joint honours in Mathematics and Computer Science, a joint major and honours in Statistics and Computer Science, a joint major in Physics and Computer Science, and a joint major in Computer Science and Biology. There are also computer science programs through the Faculty of Arts (major and minor concentrations in Computer Science). The School's courses are also available as electives to Engineering students.

Most course instructors are faculty members of the School that do research in the areas they teach. Students will learn in a small classroom environment, get to know their professors and have opportunity to do cutting-edge research. Some graduate courses in Computer Science are available to suitable qualified senior undergraduates. The School offers their students large computing labs in the Lorne Trottier Building that is dedicated to undergraduate students.

Content

- 1) The Freshman Program and Computer Science
- 2) Choosing the right program in Computer Science or Software Engineering
 - 2.1 Deciding between a BSc, BA or BA & Sc degree
 - 2.2 The differences between Computer Science, Computer Engineering and Software Engineering
 - 2.3 Games Programming
 - 2.4 Combining Computer Science with another discipline
- 3) Internships
- 4) Computer Science and Graduate Studies
- 5) Possible Streams for Major Program in Computer Science
- 6) Examples of Program curriculum:
 - Major in Computer Science
 - Major in Computer Science: Computer Games Option
 - Major in Software Engineering
 - Honours in Computer Science
 - Liberal program: Core Science Component in Computer Science
 - Liberal program: Core Science Component in Software Engineering
 - Major Concentration in Computer Science
 - Major Concentration in Software Engineering

The information captured in this catalogue is also available at: www.cs.mcgill.ca/undergraduate

1) The Freshman Program and Computer Science

Academic programs at McGill are normally four-year, 120-credit programs. In the B.Sc., B.A. and B.A. & Sc. degrees, this is split into a 30-credit freshman program and a three-year, 90-credit departmental program.

Students entering directly from high schools (outside of Quebec) usually start with the 30-credit freshman program, whereas students coming from Quebec CEGEP programs are usually granted 30 credits at the time of admission and they start directly into the 90-credit departmental program. *If you are a CEGEP student, you do not need to read any more about the freshman program, since you will not need to take it.*

If you are a high school student, then you will need to consider which courses to take in your freshman year to best prepare you for a Computer Science or Software Engineering Major. There is a separate freshman program for each degree, as specified at:

- B.Sc. Freshman Program: <http://www.mcgill.ca/artscisao/bsc/freshman/>
- B.A. Freshman Program: <http://www.mcgill.ca/artscisao/ba/freshman/>
- B.A. & Sc. Freshman Program: <http://www.mcgill.ca/artscisao/basc/freshman/>

In all cases, if you intend to study Computer Science or Software Engineering, then you should complete two calculus courses in your freshman year (MATH 139 or MATH 140 or MATH 141) and (MATH 141 or MATH 151).

If you can find room in your freshman year, you should also try to take MATH 133 (133 Vectors, Matrices and Geometry) and COMP 202. If you are unable to do so, you can still follow a Computer Science or Software Engineering major, but you will not be able to proceed to the higher-level courses as quickly.

Science Freshman Program: Computer Science / Software Engineering

[COMP 202](#) Intro to Computing 1
[MATH 133](#) Vectors, Matrices and Geometry
two calculus courses*
plus at least 3 of BIOL/CHEM/PHYS**

Choosing your calculus Courses::

- Students with no previous knowledge of Calculus should take [MATH 139](#) Calculus, followed by [MATH 141](#) Calculus 2.

Students with high school calculus take either the sequence [MATH 140](#) Calculus 1 / [MATH 141](#) Calculus 2 or the sequence [MATH 150](#) Calculus A / [MATH 151](#) Calculus B . The MATH 150/MATH 151 sequence covers the material of (and gives equivalence for) MATH 140, MATH 141 and [MATH 222](#) Calculus 3 and is more challenging than the standard MATH 140/MATH 141 sequence.

- Students who do not take COMP 202 in the freshman year may still follow these majors, but might have to start with COMP 202 in the U1 year.
- Students interested in the Cognitive Science minor may also want to take [PSYC 100](#) Introduction to Psychology.
- Students interested in Joint Physics and Computer Science programs must take [PHYS 131](#) Mechanics and Waves and [PHYS 142](#) Electromagnetism & Optics.
- Students with High School calculus and who are interested in the Joint Honours Mathematics and Computer Science program are strongly advised to consider taking the more challenging calculus sequence [MATH 150](#) Calculus A and [MATH 151](#) Calculus B .
- Students who have not taken all of Biology, Chemistry and Physics at the grade 12 level should include any missing subjects in their freshman program.

2) Choosing the right major in Computer Science or Software Engineering

There are many possible options for studying Computer Science or Software Engineering at McGill. The School of Computer Science offers many programs. The next pages want to give a rough overview of the existing options. You can find similar information at <http://www.cs.mcgill.ca/prospective-students/undergraduate>.

2.1) Deciding between a BSc, BA or BA & Sc degree

Bachelor of Science - B.Sc. (Faculty of Science):

Most of our students choose a B.Sc. degree program. The B.Sc. degree is suitable for you if: you are coming from a high school program outside of Quebec and you have enjoyed both Mathematics and Science; or you are coming from any Science CEGEP program from within Quebec. *If you decide on a B.Sc. degree, then you should choose "Faculty of Science" on your application form.*

The following programs are available leading to a B.Sc. degree:

- **Major Programs**
 - Major in Computer Science
 - Major in Computer Science: Computer Games Option
 - Major in Software Engineering
- **Honours Programs**
 - Honours in Computer Science
- **Liberal Programs**
 - Core Science Component in Computer Science
 - Core Science Component in Software Engineering
- **Joint Programs**
 - Major in Mathematics and Computer Science
 - Major in Statistics and Computer Science
 - Honours in Mathematics and Computer Science
 - Honours in Statistics and Computer Science
 - Major in Physics and Computer Science
 - Major in Computer Science and Biology

The **Major in Computer Science** is the standard major program offered by the School of Computer Science. It provides a broad introduction to the principles of computer science and offers ample opportunity to acquire in-depth knowledge in one or more sub-areas of interest to the student. Given its credit requirements, this program can be easily combined with minors or minor concentrations of other disciplines.

The **Major in Computer Science: Computer Games Option** is a specialization within Computer Science. It fulfills all the basic requirements of the Major in Computer Science. Complementary courses focus on topics that are important to understanding the technology behind computer games and to gaining experience in software development and design needed for computer game development.

The **Major in Software Engineering** provides a broad introduction to the principles of computer science and covers in depth the design and development of software systems. A more detailed explanation of the difference between the Computer Science and Software Engineering majors is given below.

The **Honours in Computer Science** provides a broad introduction to the principles of computer science and offers ample opportunity to acquire in-depth knowledge of several sub-disciplines. Compared to the major in Computer Science, it has a stronger focus on the foundations of algorithms, contains a research component and requires students to take advanced courses that cover sub-areas in more detail.

The **Core Science Components in Computer Science or Software Engineering of the B.Sc. Liberal program** are similar in spirit to the corresponding major programs, but with less credit requirements. Thus, they do not provide quite as much depth than the major programs but allow students plenty of room for a breadth options which could be a minor

or major concentration from a wide variety of areas. s just one example, if you are interested in studying Computer Science or Software Engineering and also preparing for medical school or life science interdisciplinary studies, you could combine your Computer Science Core Science Component with the new Interdisciplinary Minor in Life Sciences.

The **Joint Programs** are discussed in detail below in **Section 2.4**

Bachelor of Arts - B.A. (Faculty of Arts):

The following programs are available leading to a B.A. degree:

- Major Concentration in Computer Science
Supplementary Minor Concentration in Computer Science
- Minor Concentration in Computer Science
- Major Concentration in Software Engineering

Some students choose to complete a B.A. degree. In fact, it is possible to get quite a strong training in Computer Science within the B.A. degree program by taking both the **Computer Science Major Concentration** along with the **Supplementary Minor in Computer Science**. In terms of Computer Science content, this is almost the same at the B.Sc. major in Computer Science. You will be taking the same Computer Science classes and you will work in the same state-of-the-art computer labs as students in the B.Sc. program. In fact, in the classes and labs there are no distinctions between students in the B.A. program and students in the B.Sc. program - they are equal.

The B.A. degree is suitable for you if your background does not include very much Science, but you do have a good background in Mathematics and you have an interest and aptitude for logical reasoning. *If you decide on a B.A. degree, then you should choose "Faculty of Arts" on your application form.*

Bachelor of Arts and Science - B.A. & B.Sc. (Faculties of Arts and Science):

There is one final possibility, and that is the combined B.A. & Sc. degree, which is jointly offered by the Faculty of Science and the Faculty of Arts. In the B.A. & Sc. degree you can combine a 36-credit Major Concentration in Computer Science or Software Engineering with a Major Concentration or two Minor Concentrations from Arts. For example, you could combine your Computer Science studies with areas such Economics, Anthropology, Philosophy, Linguistics, Geography and many other possibilities.

Another option is the Interfaculty or Honours program in Cognitive Science. This is currently the only major and honours programs in Cognitive Science available at McGill (there is Cognitive Science minor in the B.Sc. program). *If you decide on a B.A. & Sc. degree, you should choose "Bachelor of Arts & Science" on your application form.*

2.2) The differences between Computer Science, Computer Engineering and Software Engineering

Computer Science and Computer Engineering

The School of Computer Science (in the Faculty of Science) offers degrees in Computer Science, whereas the Faculty of Engineering offers a degree in Computer Engineering. It is important for you to understand the difference, so that you can apply to the correct Faculty.

A **Computer Science degree** is more suitable for students more focused on the foundations of computer science and the development of software, or students who would like to combine their studies with mathematics or physics or with a wide variety of minors in Science, Management or Arts. Within a Computer Science degree you have a lot of flexibility to learn about many areas of foundational and applied topics as it relates mostly to software.

A **Computer Engineering degree** is suitable for students who want core engineering courses and prefer to specialize more in computer hardware than in computer software. The Engineering degree programs have significantly less flexibility for combined studies and many fewer courses relating to software.

Students interested in a Computer Science degree should apply to either the Faculty of Science or Faculty of Arts, as outlined above.

Computer Science and Software Engineering

The School of Computer Science, in the Faculty of Science, offers two majors - a Major in Computer Science and a Major in Software Engineering. We have professors who are specialists in both areas, with lots of exciting courses to choose from. So, which major is better for you? In fact, both programs share some common core courses which provide the foundations of computer science.

The main difference is that the Software Engineering program contains a larger set of required courses concerning the principled design and development of software. See our Software Engineering page for more information.

If you are entering McGill from a high school, you may not have to decide which you prefer right away. You can start with the introductory courses that are common to both programs and then decide which you prefer.

If you are entering McGill from CEGEP, then you will be asked to specify a major when you apply. However, there is still quite a bit of flexibility and you will have some time to decide which you prefer. If you are accepted to McGill in the Faculty of Science you can change your major from Computer Science to Software Engineering (or vice versa) when you actually register at McGill, and you can even change it after you have taken some introductory courses.

Remember, to leave your options open to choose between a Computer Science major or a Software Engineering major, make sure you apply to the Faculty of Science. Only the Faculty of Science offers both of these programs.

The B.Sc. degree *and* B.S.E. degree in Software Engineering

If you have decided that you would like to pursue a degree in Software Engineering, you still have to decide if you would prefer the B.Sc. in Software Engineering, offered through the School of Computer Science in the Faculty of Science, or the B.S.E. in Software Engineering offered through the Faculty of Engineering. What is the difference?

This is actually quite an important decision since the programs are offered in two different Faculties and it is much trickier to switch between Faculties than to switch between majors within one Faculty.

Both the B.Sc. and B.S.E programs in Software Engineering share a common core of software engineering courses, and you will be with the same students in many of these core courses. The big difference is in what courses you must take outside of this common core.

The **B.Sc. degree** is structured so that you have quite a bit of flexibility to take more courses about software or general Computer Science. You also have the option to follow 7 or 8 elective courses which gives you the opportunity to take further higher-level software courses or to pursue a minor program in another field. With the B.Sc. degree you will learn all about Software Engineering and whatever else interests you. You will have the opportunity to take a wide range of classes with both foundational and practical applications. In addition, you can choose your complementary and elective courses to learn more about computer applications or to broaden your studies in Arts, Sciences or Management.

For students coming from CEGEP, the B.Sc. degree fits into 90 credits and can be completed in three years.

The **B.S.E degree** is structured so that in addition to courses specific to Software Engineering, you must also take many engineering-specific courses that are common to the Engineering degrees, plus more courses related to hardware. For example, students in the B.S.E. program must take required courses such as Fundamentals of Electrical Engineering, Circuit Analysis, Fundamentals of Signals and Systems, and so on. Since the Engineering courses require quite a few credits, there is less flexibility for studying more software courses and many fewer electives. The inclusion of all of these extra engineering courses also means that students must take more total credits. For students coming from CEGEP, the B.S.E. degree is 112-114 credits, meaning that it takes 3 1/2 years to complete. Students graduating from this program are

eligible for registration as professional engineers, since the program has been accredited by the Canadian Engineering Accreditation Board.

Students in both programs have a wide variety of job opportunities and access to internship opportunities. Students in the B.Sc. program have more flexibility to choose their programs and can, for example, select courses that also prepare them for a career in games programming or to study a minor in Management or Marketing.

Students who wish to take the B.Sc. degree in Software Engineering should indicate the "Faculty of Science" on their application form.

2.3) Can I learn about games programming at McGill?

Yes! Many students become interested in computer science through computer games. How do they work? How can you understand more about them or even make your own?

There are several ways to get involved in games programming. You can either enrol in the new Major in Computer Science - Computer Games Option, a special version of the Computer Science Major with an emphasis on Games, or you can include relevant courses in any of the Computer Science and Software Engineering major programs.

There are a variety of **game-related courses** offered at the School of Computer Science at McGill and many directions you can specialize in and pursue if you want to concentrate your studies on computer games, whether you want to do game research, find a job in the game industry, or just want to understand them better.

Several professors use computer games in their research, or do **research on computer games**. You can find out more about current projects at the GR@M website, covering various aspects of Games Research at McGill and the Mammoth project web-page. Mammoth is a collective framework for multiplayer game research used by several faculty and students, and built by both graduate and undergraduate students.

Montreal has several local game companies and branches of large game companies. Graduates have found work in EA, Ubisoft, among many others.

The School of Computer Science also holds a yearly Summer Camp for high school students. This includes game programming as a major topic.

2.4) Combining Computer Science with another discipline



For many students, combining computer science with another discipline is a great way to explore a wider range of options.

Joint Major (or Honours) in Mathematics/Statistics and CS

Computer science and mathematics are a natural fit. In fact, many of the early pioneers of computer science, such as **Alan Turing** and **John von Neumann**, were mathematicians themselves! By combining the two fields, students develop the rigorous conceptual thinking and problem-solving abilities that characterise training in mathematics. At the same time, they are continually challenged by their computer science courses to put those ideas into practice. Advanced computer science subjects ranging from machine learning to cryptography to internet searching are often highly mathematical, so the solid training our joint programs provide is an excellent preparation for graduate school or careers in industry. The joint honours programs are particularly challenging and, therefore, particularly rewarding. Many of our best students follow this option and go on to complete PhD s in their choice of computer science or mathematics at the best universities in the world. They invariably say that their training in the joint honours program was decisive in their success.

Joint Major in Physics and Computer Science

Computer science and physics are no strangers either. From the invention of randomized algorithms to study nuclear explosions during the top-secret Manhattan Project, to the invention of the World Wide Web at the CERN particle physics laboratory, physicists have been enthusiastic users and innovators when it comes to computers. As time goes on, large-scale computer simulations play an ever-increasing role in physics research, resulting in an ever-increasing need for physicists with real training in computer science. On another front, the world internet speed record is currently held by scientists working to develop a worldwide distributed computing system for analyzing the avalanche of data expected to be generated by the next generation of experiments at that same CERN that invented the Web. A joint major in computer science and physics positions students to work as professional physicists, as software engineers, or somewhere in between, pushing the frontiers of scientific technology.

Joint Major in Computer Science and Biology

In recent years, computer technology has become an integral part of research and development in the life-sciences such as medicine, biology, or biochemistry. This joint major will train students in the fundamentals of biology – with a focus on molecular biology - and will give them computational and mathematical skills needed to manage, analyze, and model large biological datasets. Additionally, the program contains several integrative features such as a joint independent study course and computer science courses especially designed for the life-sciences.

Liberal Program with a Core Science Component in Computer Science or Software Engineering

The Liberal programs allow students to combine studying the core of one science discipline with a breadth component. This breadth component may contain a minor program, and Arts Minor or Major Concentration, a further core science component. Computer science and physics are no strangers either. Thus, the liberal program is ideal for students who want to cover more than one discipline in sufficient depth.

Adding a minor program

Computer Science students are often interested in many other areas. If you choose a Computer Science Major program you use 60 credits to fulfill the requirements for the Major, but you still have 30 credits (10 courses) that can be taken as freely chosen electives. Some students like to use their elective credits to complete a Minor Program. Minor programs are usually between 18 and 24 credits and there are a wide variety of minors that Computer Science students can take. These include some popular choices like Cognitive Science, Management, Mathematics, Music Technology and Technical Entrepreneurship. However, minors are not limited to technology-related subjects and a wide range of other Science minors are available including Biology, Biotechnology, Earth and Planetary Sciences and Environment. You can even broaden your studies outside of Science and select a minor from Arts. You can choose from a huge list of minor and major concentrations in Arts including Economics, English Literature, Linguistics, Philosophy and many others. By combining a Computer Science or Software Engineering major with a minor you can really build the combined program of your choice!

Minor in Computer Science or in Computational Molecular Biology

Perhaps your main interests lie outside of Computer Science, but you still have an interest in a smaller program in Computer Science. The Computer Science minor is a great way to combine some Computer Science studies with your main interests. If you end up really liking your Computer Science classes you can consider switching into the Computer Science Major program.

There is also a Computational Molecular Biology Minor. If your main major is in the biological sciences, this minor may interest you. It provides you with the skills from computer science to solve computational problems arising in molecular biology and genomics and also provides you with the necessary skills to build software tools from these algorithms.

3) Internship during your undergraduate studies?

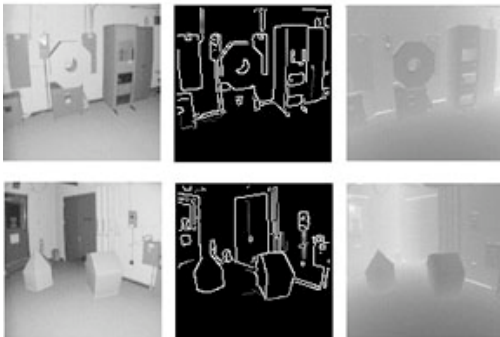


- Some students would like to gain valuable paid practical experiences during their degree studies. In Computer Science at McGill there are two good ways to do this.
- The IYS (*Internship Year in Science*) is a paid 8, 12 or 16 month internship available to B.Sc. students. It is a career-related, professionally supervised, paid work term which allows you to experience firsthand the current practices and technologies in business and industry.
- The IYS program offers you the chance to test your career goals, gain confidence and experience and to return to your studies more focused, with a sense of where your academic careers might lead.

Due to the longer work term, employers tend to assign IYS interns to significant projects which can be seen through from start to finish.

- If you complete an IYS internship then this will be recognized by adding the clause "Internship Program" to your degree. So, a degree "Major in Computer Science" would become "Major in Computer Science - Internship Program".
- Some students find an internship rather long, and would prefer a shorter work experience that might fit into the summer term. The **Industrial Practicum** works well for this situation. It is a 4-month work term. If you complete an Industrial Practicum you will have this recognized by having a 0-credit course called "Industrial Practicum" added to your transcript.
- For more information about IP and IYS: <http://www.mcgill.ca/science/internships-field/internships/>.

4) A Computer Science Degree and Graduate Studies



Many of our students go on to graduate studies at either the M.Sc. or Ph.D. level at top universities in Canada, the US, and throughout the world. Sometimes our McGill Computer Science undergraduates stay at McGill to do their M.Sc. studies and sometimes they go directly to M.Sc. or Ph.D. studies at another university. Although we hate to see our great students leave us, it is usually advisable for Ph.D. students to get some graduate studies experiences at another university.

Which program best prepares me for graduate studies?

An honours program provides the most rigorous education and provides an ideal preparation for graduate school. However, many of our majors students also go on to be excellent graduate students. Students interested in interdisciplinary graduate research may want to consider a joint major in Math or Physics or combining a major in Computer Science with a minor in another discipline, for example Cognitive Science.

Whichever program you choose, you should pick upper-level courses that interest you and where you can get to know some professors in a smaller class setting. When you apply to graduate school it will be helpful if several professors have gotten to know you well and can write good recommendation letters for you. We also encourage you to get involved in some research experiences during your undergraduate studies. This is the best way to discover the excitement of doing research and to get some ideas for potential M.Sc. or Ph.D. research topics. Our professors really enjoy working with our undergraduate students and we have a lot of interesting projects.

What about financial support and fellowships for graduate studies?

Many undergraduate students are not aware that most graduate work in Computer Science comes with some sort of financial support, whether that is through teaching assistantships, research assistantships, fellowships or a combination of these.

Students who are Canadian citizens or permanent residents should be aware of the **NSERC** postgraduate fellowships. Remember to apply for these at the beginning of your last year of undergraduate studies. Don't be shy to ask for help from our CS professors in preparing your applications and to provide reference letters. We like to see our students be successful in these competitions.

Students who are residents of Quebec may, in addition to NSERC, also apply for an **FQRNT fellowship**. This is also done at the beginning of your last year of undergraduate studies.

Even if you do not qualify for one of these fellowships, computer science departments will usually find other means to provide financial support for qualified graduate students.

5) Possible Streams for Major program in Computer Science

Because of the prerequisite structure, it can be difficult to fit all the core courses into a three-year time sequence. To help students do this, the following tables suggest sequences of core courses. Students usually take 5 courses (15 credits) per term. Options and electives can be added to make up 15 credits each term.

- 1) Possible Core Course Schedule for three-year B.Sc. Major Program in Computer Science for students with sufficient programming background.

| Year | COMP | MATH |
|------|--|--|
| U1 | 250, 251, 206, 302 | 222, 240 223 |
| U2 | 273 310 303 or 304, At least two of 330, 350, 360 | At least one of 318, 323, 324 and 340 <i>(minimum requirement must include at least one of MATH 323 & 340).</i> |
| U3 | Remaining credits from 300-level or above CS courses. | At least one of 318, 323, 324 and 340 <i>(minimum requirement must include at least one of MATH 323 & 340).</i> |

- 2) Possible Streams for three-year B.Sc. Major Program in Computer Science for students who start with COMP 202.

| Year | COMP | MATH |
|------|--|--|
| U1 | 202 250, 206 | 222, 240 223 |
| U2 | 251, 273, 302 Possibly 310 303 or 304 At least 1 of 330, 350, 360 | At least two of 318, 323, 324 and 340 <i>(minimum requirement must include at least one of MATH 323 & 340).</i> |
| U3 | Possibly 310 At least 1 of 330, 350, 360 Remaining credits from 300-level or above CS courses. | |

We recommend that MATH 240 be taken simultaneously with COMP 202 or COMP 250. These are only suggestions and any changes, which fit the pre-requisite structure, are acceptable.

6.1) MAJOR IN COMPUTER SCIENCE (60-63 credits)

Students should talk to an academic adviser before choosing their complementary courses.

Required Courses

(27-30 credits)

| | | |
|-----------|-----|-------------------------------------|
| COMP 202* | (3) | Introduction to Computing 1 |
| COMP 206 | (3) | Introduction to Software Systems |
| COMP 250 | (3) | Introduction to Computer Science |
| COMP 251 | (3) | Data Structures and Algorithms |
| COMP 273 | (3) | Introduction to Computer Systems |
| COMP 302 | (3) | Programming Languages and Paradigms |
| COMP 310 | (3) | Operating Systems |
| MATH 222 | (3) | Calculus 3 |
| MATH 223 | (3) | Linear Algebra |
| MATH 240 | (3) | Discrete Structures 1 |

* Students who have sufficient knowledge in a programming language do not need to take COMP 202.

Complementary Courses

(33 credits)

At least 6 credits selected from:

| | | |
|----------|-----|---------------------------------------|
| COMP 330 | (3) | Theoretical Aspects: Computer Science |
| COMP 350 | (3) | Numerical Computing |
| COMP 360 | (3) | Algorithm Design Techniques |

At least 3 credits selected from:

| | | |
|----------|-----|------------------------|
| COMP 303 | (3) | Software Development |
| COMP 304 | (3) | Object-oriented Design |

3-9 credits selected from:

(Must include at least one of MATH 323 and MATH 340)

| | | |
|----------|-----|-----------------------|
| MATH 318 | (3) | Mathematical Logic |
| MATH 323 | (3) | Probability |
| MATH 324 | (3) | Statistics |
| MATH 340 | (3) | Discrete Structures 2 |

The remaining credits selected from computer science courses at the 300-level or above (except COMP 364, COMP 396, COMP 400, COMP 431) and ECSE 508.

Note: Students have to make sure that they have the appropriate pre-requisites when choosing upper level courses.

6.2) MAJOR IN COMPUTER SCIENCE: COMPUTER GAMES OPTION (62-69 CREDITS)

Required Courses (41-44 credits)

| | | |
|-----------|-----|--------------------------------------|
| COMP 202* | (3) | Introduction to Computing 1 |
| COMP 250 | (3) | Introduction to Computer Science |
| COMP 251 | (3) | Data Structures & Algorithms |
| COMP 206 | (3) | Introduction to Software Systems |
| COMP 273 | (3) | Introduction to Computer Systems |
| COMP 302 | (3) | Programming Languages & Paradigms |
| COMP 308 | (1) | Computer Systems Lab |
| COMP 310 | (3) | Operating Systems |
| COMP 330 | (3) | Theoretical Aspects: Compute Science |
| COMP 322 | (1) | C++ |
| COMP 361 | (3) | System Development Project |
| COMP 557 | (3) | Fundamentals of Computer Graphics |
| MATH 222 | (3) | Calculus 3 |
| MATH 223 | (3) | Linear Algebra |
| MATH 240 | (3) | Discrete Structures |
| MATH 323 | (3) | Probability |

* Students who have sufficient knowledge in a programming language are not required to take Comp 202.

Complementary Courses (21-25 credits)

3 credits selected from:

| | | |
|----------|-----|-----------------------------|
| COMP 350 | (3) | Numerical Computing |
| COMP 360 | (3) | Algorithm Design Techniques |

6-8 credits selected from:

| | | |
|----------|-----|--------------------------------------|
| COMP 303 | (3) | Software Development |
| COMP 304 | (4) | Object-oriented Design |
| COMP 335 | (3) | Software Engineering Methods |
| COMP 529 | (3) | Software Architecture |
| COMP 533 | (3) | Object-oriented Software Development |

6 credits selected from:

| | | |
|-------------|-----|-----------------------------|
| COMP 421 | (3) | Database Systems |
| COMP 535 | (3) | Computer Networks 1 |
| or COMP 435 | (3) | Basics of Computer Networks |
| COMP 409 | (3) | Concurrent Programming |

6-8 credits selected from:

| | | |
|----------|-----|-----------------------------------|
| COMP 424 | (3) | Topics: Artificial Intelligence 1 |
| COMP 507 | (3) | Computational Geometry |
| COMP 521 | (4) | Modern Computer Games |
| COMP 522 | (4) | Modelling and Simulation |

6.3) MAJOR IN SOFTWARE ENGINEERING (69 credits)

Required Courses (60 credits)

| | | |
|-------------|-----|---------------------------------------|
| COMP 202* | (3) | Introduction to Computing 1 |
| COMP 206 | (3) | Introduction to Software Systems |
| COMP 250 | (3) | Introduction to Computer Science |
| COMP 251 | (3) | Data Structures and Algorithms |
| COMP 273 | (3) | Introduction to Computer Systems |
| COMP 302 | (3) | Programming Languages and Paradigms |
| COMP 304 | (3) | Object-oriented Design |
| COMP 310 | (3) | Operating Systems |
| or ECSE 427 | (3) | Operating Systems |
| COMP 330 | (3) | Theoretical Aspects: Computer Science |
| COMP 360 | (3) | Algorithm Design Techniques |
| COMP 361 | (3) | Systems Development Project |
| ECSE 321 | (3) | Introduction to Software Engineering |
| ECSE 428 | (3) | Software Engineering Practice |
| ECSE 429 | (3) | Software Validation |
| ECSE 495 | (3) | Software Engineering Design Project |
| MATH 222 | (3) | Calculus 3 |
| MATH 223 | (3) | Linear Algebra |
| MATH 240 | (3) | Discrete Structures 1 |
| MATH 323 | (3) | Probability |
| MATH 324 | (3) | Statistics |

* Students who have sufficient knowledge in a programming language do not need to take COMP 202.

Complementary Courses (9 credits)

Selected from COMP, ECSE or MATH courses listed in the undergraduate calendar.

6.4) HONOURS IN COMPUTER SCIENCE (72-75 credits)

Honours students must maintain a CGPA of at least 3.00 during their studies and at graduation.

Required Courses (42-45 credits)

| | | |
|-------------|-----|---------------------------------------|
| COMP 202* | (3) | Introduction to Computing 1 |
| COMP 206 | (3) | Introduction to Software Systems |
| COMP 250 | (3) | Introduction to Computer Science |
| COMP 252 | (3) | Algorithms and Data Structures |
| COMP 273 | (3) | Introduction to Computer Systems |
| COMP 302 | (3) | Programming Languages and Paradigms |
| COMP 310 | (3) | Operating Systems |
| COMP 330 | (3) | Theoretical Aspects: Computer Science |
| COMP 350 | (3) | Numerical Computing |
| COMP 362 | (3) | Honours Algorithm Design |
| COMP 400 | (3) | Technical Project and Report |
| MATH 222 | (3) | Calculus 3 |
| MATH 223 | (3) | Linear Algebra |
| MATH 240 | (3) | Discrete Structures 1 |
| MATH 340 | (3) | Discrete Structures 2 |
| or MATH 350 | (3) | Graph Theory and Combinatorics |

* Students who have sufficient knowledge in a programming language do not need to take COMP 202.

Complementary Courses (30 credits)

At least 3 credits selected from:

| | | |
|----------|-----|------------------------|
| COMP 303 | (3) | Software Development |
| COMP 304 | (3) | Object-oriented Design |

6 credits selected from:

| | | |
|----------|-----|--------------------|
| MATH 318 | (3) | Mathematical Logic |
| MATH 323 | (3) | Probability |
| MATH 324 | (3) | Statistics |

The remaining credits selected from computer science courses at the 300-level or above (except COMP 364, COMP 396, COMP 400, COMP 431) and ECSE 508. At least 12 credits must be at the 500 level.

6.5) LIBERAL PROGRAM:

Core Science Component in *COMPUTER SCIENCE* (45 credits)

Required Courses (21 credits)

| | | |
|-----------|-----|----------------------------------|
| COMP 202* | (3) | Introduction to Computing 1 |
| COMP 206 | (3) | Introduction to Software Systems |
| COMP 250 | (3) | Introduction to Computer Science |
| COMP 251 | (3) | Data Structures and Algorithms |
| COMP 273 | (3) | Introduction to Computer Systems |
| MATH 222 | (3) | Calculus 3 |
| MATH 240 | (3) | Discrete Structures 1 |

* Students who have sufficient knowledge in a programming language do not need to take COMP 202, but it must be replaced with an additional computer science complementary course.

Complementary Courses (24 credits)

3 - 6 credits from:

| | | |
|----------|-----|--------------------|
| MATH 223 | (3) | Linear Algebra |
| MATH 318 | (3) | Mathematical Logic |
| MATH 323 | (3) | Probability |
| MATH 324 | (3) | Statistics. |

MATH 340 (3) Discrete Structures 2

At least 3 credits from:

COMP 330 (3) Theoretical Aspects: Computer Science

COMP 350 (3) Numerical Computing

COMP 360 (3) Algorithm Design Techniques

At least 3 credits from:

COMP 302 (3) Programming Languages and Paradigms

COMP 303 (3) Software Development

The remaining complementary courses should be selected from any COMP courses at the 300-level or above except COMP 364, COMP 396, COMP 400 and COMP 431.

Note: Advanced COMP courses have more pre-requisites than the required courses for this program. Students have to make sure that they have the appropriate pre-requisites when choosing upper level courses.

6.6) LIBERAL PROGRAM:

Core Science Component in SOFTWARE ENGINEERING (48 - 49 credits)

Required Courses (36 credits)

COMP 202* (3) Introduction to Computing 1
COMP 206 (3) Introduction to Software Systems
COMP 250 (3) Introduction to Computer Science
COMP 251 (3) Data Structures and Algorithms
COMP 273 (3) Introduction to Computer Systems
COMP 302 (3) Programming Languages and Paradigms
COMP 303 (3) Software Development
COMP 304 (3) Object-oriented Design
COMP 310 (3) Operating Systems
COMP 361 (3) Systems Development Project
MATH 223 (3) Linear Algebra
MATH 240 (3) Discrete Structures 1

* Students who have sufficient knowledge in a programming language do not need to take COMP 202 and can replace it with additional computer science complementary course credits.

Complementary Courses (12 - 13 credits)

3 credits selected from:

COMP 330 (3) Theoretical Aspects: Computer Science

COMP 360 (3) Algorithm Design Techniques

3 credits selected from:

COMP 335 (3) Software Engineering Methods

ECSE 321 (3) Introduction to Software Engineering

6 - 7 credits selected from:

COMP 322 (1) Introduction to C++

COMP 409 (3) Concurrent Programming

COMP 421 (3) Database Systems

COMP 435 (3) Basics of Computer Networks

Or COMP 535 (3) Computer Networks 1

COMP 520 (4) Compiler Design

COMP 525 (3) Formal Verification

COMP 529 (4) Software Architecture

COMP 533 (3) Object-Oriented Software Development

Or any computer science course at the 300-level or above, **excluding** COMP 364, COMP 396, and COMP 431.

6.7) Major Concentration in Computer Science (36 credits)

This major concentration represents an in-depth introduction to computer science and its sub-areas. For students interested in further study in Computer Science can combine the Major Concentration in Computer Science with the Supplementary Minor Concentration in Computer Science to constitute a program very close to the Major in Computer Science offered by the Faculty of Science. For further information, please consult the program adviser.

NOTE: Students with two programs in the same department must have a third in a different discipline to be eligible to graduate. Please refer to the Faculty of Arts Degree Requirements, departmental programs.

Required Courses (21 credits)

| | | |
|-------------|-----|----------------------------------|
| COMP 202* | (3) | Introduction to Computing 1 |
| COMP 203 | (3) | Introduction to Computing 2 |
| or COMP 250 | (3) | Introduction to Computer Science |
| COMP 206 | (3) | Introduction to Software Systems |
| COMP 251 | (3) | Data Structures & Algorithms. |
| COMP 273 | (3) | Introduction to Computer Systems |
| MATH 222 | (3) | Calculus 3 |
| MATH 240 | (3) | Discrete Structures |

**Students who have sufficient knowledge in a programming language do not need to take Comp 202 but can replace it with an additional computer science complementary course.*

Complementary Courses (15 credits)

3-6 credits from:

| | | |
|----------|-----|-----------------------|
| MATH 223 | (3) | Linear Algebra |
| MATH 318 | (3) | Mathematical Logic |
| MATH 323 | (3) | Probability |
| MATH 324 | (3) | Statistics |
| MATH 340 | (3) | Discrete Structures 2 |

At least 3 credits from:

| | | |
|----------|-----|---------------------------------------|
| COMP 330 | (3) | Theoretical Aspects: Computer Science |
| COMP 350 | (3) | Numerical Computing |
| COMP 360 | (3) | Algorithm Design Techniques |

At least 3 credits from:

| | | |
|----------|-----|-----------------------------------|
| COMP 302 | (3) | Programming Languages & Paradigms |
| COMP 303 | (3) | Software Development |

The remaining credits selected from:

| | | |
|----------|-----|-----------------------|
| COMP 230 | (3) | Logic & Computability |
|----------|-----|-----------------------|

And Computer Science courses at the 300-level or above (**except COMP 364, COMP 396, COMP 400, COMP 431**) and ECSE 508.

Supplementary Minor Concentration in Computer Science (18 credits)

This Minor may be taken only by students registered in the Major Concentration in Computer Science.

There are no overlap in credits taken for this Supplementary Minor Concentration and the Major Concentration in Computer Science. Taken together, these constitute a program very close to the Major in Computer Science offered by the Faculty of Science.

Complementary Courses (18 credits):

*Selected from Computer Science courses at the 300-level or above (**except Comp 364, Comp 396, Comp 400, Comp 431**) and ECSE 508.*

At most 3 credits can be selected from:

| | | |
|----------|-----|-----------------------|
| Math 223 | (3) | Linear Algebra |
| Math 318 | (3) | Mathematical Logic |
| Math 323 | (3) | Probability |
| Math 324 | (3) | Statistics |
| Math 340 | (3) | Discrete Structures 2 |

6.8) Major Concentration in Software Engineering (36 -37credits)

This Major Concentration in Software Engineering focuses on the techniques and methodology required to design and develop complex software systems.

The major concentration provides a program of study that covers the subject commonly known as "Software Engineering". This may be used to satisfy part of the requirements for a B.A. degree.

This program does **not** lead to certification as a Professional Engineer.

Required Courses (30 credits)

| | | |
|----------|-----|----------------------------------|
| COMP 202 | (3) | Introduction to Computing 1 |
| COMP 206 | (3) | Introduction to Software Systems |
| COMP 250 | (3) | Introduction to Computer Science |
| COMP 251 | (3) | Data Structures & Algorithms |
| COMP 302 | (3) | Program. Languages & Paradigms |
| COMP 303 | (3) | Software Development |
| COMP 304 | (3) | Object-oriented Design |
| COMP 421 | (3) | Database Systems |
| MATH 223 | (3) | Linear Algebra |
| MATH 240 | (3) | Discrete Structures 1 |

Complementary Courses (6-7 Credits)

3 credits selected from:

| | | |
|--------------------|-----|--------------------------------------|
| COMP 335 | (3) | Software Engineering Methods |
| Or ECSE 321 | (3) | Introduction to Software Engineering |

3-4 credits selected from:

| | | |
|-----------|-----|--|
| COMP 322 | (1) | Introduction to C++ |
| COMP 361 | (3) | Systems Development Project |
| COMP 529 | (3) | Software Architecture |
| COMP 533 | (3) | Object-Oriented Software Development |
| Or | | any Computer Science course at the 300 level or above, excluding COMP 364, COMP 396 & COMP 431. |