Sh*t I Wish I Knew

2020 Edition
University Program Information
**What is CGPA?**
- **Cumulative Grade Point Average**
- Calculated by taking the weighted average GPA of each of your courses
- Can be broken down by year (annual CGPA)
  - Annual term: Summer to Winter

**Carleton uses a 12-point GPA system**
- See chart (right) for details
- Grade points assigned intervallically
  - Highest possible grade point in a course: any grade >= 90%

---

<table>
<thead>
<tr>
<th>Grade points for 1.0 credits</th>
<th>Grade Points for 0.5 credits</th>
<th>Percentage Equivalency</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+ = 12.0</td>
<td>A+ = 6.0</td>
<td>90-100%</td>
</tr>
<tr>
<td>A  = 11.0</td>
<td>A  = 5.5</td>
<td>85-89%</td>
</tr>
<tr>
<td>A- = 10.0</td>
<td>A- = 5.0</td>
<td>80-84%</td>
</tr>
<tr>
<td>B+ = 9.0</td>
<td>B+ = 4.5</td>
<td>77-79%</td>
</tr>
<tr>
<td>B  = 8.0</td>
<td>B  = 4.0</td>
<td>73-76%</td>
</tr>
<tr>
<td>B- = 7.0</td>
<td>B- = 3.5</td>
<td>70-72%</td>
</tr>
<tr>
<td>C+ = 6.0</td>
<td>C+ = 3.0</td>
<td>67-69%</td>
</tr>
<tr>
<td>C  = 5.0</td>
<td>C  = 2.5</td>
<td>63-66%</td>
</tr>
<tr>
<td>C- = 4.0</td>
<td>C- = 2.0</td>
<td>60-62%</td>
</tr>
<tr>
<td>D+ = 3.0</td>
<td>D+ = 1.5</td>
<td>57-59%</td>
</tr>
<tr>
<td>D  = 2.0</td>
<td>D  = 1.0</td>
<td>53-56%</td>
</tr>
<tr>
<td>D- = 1.0</td>
<td>D- = 0.5</td>
<td>50-52%</td>
</tr>
<tr>
<td>F  = 0.0</td>
<td>F  = 0.0</td>
<td>0-49%</td>
</tr>
</tbody>
</table>
1st-Year Courses

**COMP 1405** - Intro to CS I: variable types, branching and looping structures, arrays, functions, sorting and searching (in Python)

**COMP 1406** - Intro to CS II: object-oriented programming, basic data structures, recursion, efficiency, debugging (in Java)

**COMP 1805** - Discrete Structures I: logic, proof techniques, set theory, graph theory, asymptotic analysis of algorithms

**MATH 1007** - Calculus I: limits, derivatives and differentiation, max/min optimization, basic integrals (similar to Grade 12 Calculus)

**MATH 1104** - Linear Algebra I: systems of linear equations, matrix algebra, vector spaces, eigenvalues, complex numbers
2nd-Year Courses

**COMP 2401** - Intro to Systems Programming: memory management, pointers, process management (in C)

**COMP 2402** - Abstract Data Structures: stacks, queues, lists, trees, graphs

**COMP 2404** - Intro to Software Eng.: object-oriented software development (in C++)

**COMP 2406** - Web Applications: HTML/CSS, JavaScript programming, database querying, web technologies

**COMP 2804** - Discrete Structures II: counting, probability, recurrence relations, randomized algorithms

**STAT 2507** - Intro to Stat Modelling I: random variables, probability distributions, distribution of sample mean, hypothesis testing
3rd-Year Courses

**COMP 3000** - Operating Systems: Linux OS and file system, low-level C programming

**COMP 3004** - Object-Oriented Software Eng: group project class, UML, computer ethics

**COMP 3005** - Database Management Systems: ER modelling, SQL, relational algebra, normalization

**COMP 3007** - Programming Paradigms: functional and logical programming (Haskell, Lisp/Scheme, Prolog)

**COMP 3804** - Design and Analysis of Algorithms I: divide-and-conquer, dynamic programming, shortest path, NP-completeness
Importance of Grades

- Annual CGPA of 10.00 (A- average) required to renew entrance scholarship
- CGPA of 10.00 in first year required to qualify for DSRI
- Overall CGPA of 8.00 (B average) required to be in Co-op
- Some courses require a minimum grade in a prerequisite course
  - Minimum grade of C- in COMP 1406 required to register for COMP 2401, 2402, and 2406
  - Minimum grade of C- in COMP 1805 required to register for COMP 2804, 3005, and 3007
Free Electives vs. Breadth Electives

Free electives are courses that are:

- **Not** included in your **major**
- **Not** on the list of **prohibited courses**:
  - BUSI 2402, BUSI 3400, COMP 1001, COMP 1004, MATH 1009, MATH 1119, MATH/ECON 1401, MATH/ECON 1402
  - All courses in BIT, IMD, IRM, MPAD, NET, OSS, PLT and ITEC **except for the following**:

Breadth electives are courses that are:

- **Not** included in your **major**
- **Not** on the list of **prohibited courses** (see list on the left)
- **Not** in any of the following:
  - School of Computer Science
  - School of Mathematics and Statistics
  - Faculty of Engineering and Design
Major vs. Honours vs. Streams

- Carleton offers a BCS Major as well as a BCS Honours degree
  - The Honours degree requires COMP 3804 and a project/thesis
  - The Honours degree has higher CGPA requirements to be in good standing/to graduate
  - Many universities require an Honours degree to get into graduate studies

- What is a stream?
  - A 2.0-credit concentration in a particular CS field, and a related Honours project/thesis
  - Gives priority access to certain optional COMP courses

- Important notes:
  - You must be in the Honours program to have a stream
  - Streams are unique to the CS program at Carleton - may not be recognized at other institutions/in the workplace
Academic Advice
Time Management

- Start your assignments as early as possible
  - Even if you don’t finish, you’ll have something to hand in
  - You will be able to think about the assignment while you aren’t working on it

- Schedule specific times for schoolwork/studying/free time
  - Separating work and play helps you increase productivity while working and enjoy yourself more during free time

- Use an agenda/calendar
  - Writing things down helps you to remember them
  - You will have an organized list of everything you need that you can refer back to

- Set reminders for important dates and deadlines
  - E.g. alarms, push notifications, sticky notes
  - You don’t want to risk waking up late for a test, or missing an assignment deadline
Hofstadter's Law: It always takes longer than you expect, even when you take into account Hofstadter's Law.
Work Ethic

● Don’t procrastinate
  ○ The sooner you do your work, the sooner you can fix the bugs in your code

● Treat school like a full-time job
  ○ (except you are being paid in knowledge and grades)

● Attend all lectures
  ○ And take your own notes, even if the prof posts their notes as well
  ○ The more classes you miss, the more likely it will be for you to fall behind

● Don’t skip assignments!
  ○ Anything is better than 0%; every percent counts!

● Know when to stop
  ○ Burnout is real
  ○ Don’t keep pushing yourself to the limit at the expense of your mental/physical health
Resources

- **Learn to Google**
  - If you run into a problem, chances are thousands of others have as well
- **Consult your instructor and TAs**
  - There is no shame in asking questions
  - They are there to help you learn
- **Join a study group**
  - Your peers can help hold you accountable for your schoolwork
  - Help and support each other (as long as you don’t violate academic integrity)
- **Make use of Carleton’s resources**
  - [Science Student Success Centre](#)
  - [Paul Menton Centre](#)
  - [Carleton Computer Science Society](#)
  - [MacOdrum Library](#)
Quick Intro to Research with MacOdrum
Why do you need to know this?

➔ Electives are a necessary part of your degree
  ◆ 5.0 credits in breadth electives → 10 courses that are not COMP, STAT, or MATH
  ◆ Categories
    ● Culture and Communications
    ● Humanities
    ● Social Science
    ● Science, Engineering, and Design

➔ Minors
  ◆ Anything else you’re passionate about?
Omni

→ Main search engine MacOdrum offers
→ Searches entire collection
→ Pulls from different databases
  ◆ JSTOR
  ◆ Project MUSE
  ◆ DOAJ
  ◆ HathiTrust*
Subject Guides

➔ Curated resources by a Librarian
➔ Detailed guides include how-tos on writing and citation
➔ Quick guides offer a small sample of sources
Library Services

➔ Writing Services
  ◆ Will help with all written assignments
➔ Off-campus database access
  ◆ Search databases directly
Career Opportunities
Getting Started

- Make a resume even if you have no work experience or side projects
- Start applying now
  - You will start to learn how to market yourself
  - You will get used to rejection early on
- Go to info sessions to learn more about companies hiring from Carleton
- Apply even if you don’t meet all the requirements
- Don’t stress yourself out - you don’t HAVE to get a job in first year
Internship Opportunities

- **Dean’s summer Research internship (DSRI)**
  - Process starts December/January
  - You must find the placement by reaching out to profs

- **Federal Student Work Experience Program (FSWEP)**
  - Partially a lottery process
  - Internships

- First-year-specific internships exist (e.g. Google STEP Program)
Internships Vs Co-op

- Some job postings ask for you to be in a registered co-op program
- Coop through universities is often subsidized by the government
- Coop at Carleton University provides you with a job board
- Career Services Office available to all students - will provide resume reviews and mock interviews!
- It’s not necessary to be in the coop program to get a great internship!
  - You can always apply externally
- But it is a lot more work to find jobs yourself - takes effort and commitment
Finding Jobs

- Keep an eye on the Careers page of the companies you’re interested in!
- Indeed
- LinkedIn Jobs
- Networking Events
  - You can register for coop job fairs and networking events in first year
  - Net Night (1 per semester)
  - Career Fairs in University Center on campus
- Going to Hackathons, Tech Meetups
Resume Tips

- You can use MS Word or online resume builder templates
- A little bit of colour and style can make you stand out!
- Two columns is a popular style
- Verify your resume can be parsed (CTRL-F Test)
- Get all your friends and mentors to review your resume

- What do you put on your resume if you don't have any compsci work experience?
  - Projects!
  - Hackathons
  - Organizations you’re involved in
  - Volunteer experience
  - Non-CS work experience
  - School/uni projects
**Interview Tips**

- Go to mock interview and technical interview workshops
  - Ask your friends to interview you!
- Confidence and friendliness goes a LONG way, smile!!!!
  - Culture fit is very important and companies are looking for positive, passionate candidates
- Research the company - culture, products, tech, etc.
- Ask questions!
  - Indicates interest and will make you more familiar with the company
  - Example question: “what’s a typical day like as a developer on your team?”
- Own the things you don’t know but emphasize how much you want to learn
- Dress up professionally (even helps you feel prepared for phone interviews)
- Practice technical interview problems on HackerRank and LeetCode and brush up on CS concepts
- Follow up with a thank-you email
Learning Outside the Classroom
Personal Projects

- Personal Projects
  - Can showcase your knowledge of certain skills
    - Languages, frameworks, paradigms, etc.
- They show you go beyond just doing classwork
- Group projects are good too!
  - Join hackathons to work on something exciting for a competition
  - Start a side project with your friends
  - Find project buddies @ CCSS Dev Club
Personal Projects

- Make something you’re passionate about - a fun game, a digital planner, a website
- Remake something you use but optimized for you e.g. Notes App!
- Make something to help you learn something new e.g. mobile development
- Remake a school assignment and add more features such as a full GUI
- If you really can’t think of anything... make a [resume website](#)
- Look at projects you’re inspired by on [Devpost](#)

Examples:
- [First-year Project](#)
- [Recent Project](#)
Github

- Having a public portfolio that shows your activity and interests is a big plus
- Free Github Student Developer pack
Self-Directed Learning

- Self Directed Learning Strategies
  - Absorb Context
  - Watch videos, read articles, or ask others to get a gain context
  - Context will help you make sense of new information

- Follow a road map
  - You can't just jump into something that requires other base knowledge
  - Understanding context helps you develop a road map
Self-Directed Learning

- Strategies for finding interests within CompSci
  - Look at positions and checkout the technical requirements
  - Research the technologies required to build something you would like to make
- No one knows everything... CS is vast
- Don't be discouraged by information overloads, everyone is gonna have a niche
Sources

https://carleton.ca/academicadvising/cgpas/
https://carleton.ca/awards/scholarships/entrance-scholarships-for-new-students/
https://calendar.carleton.ca/undergrad/regulations/co-operativeeducation/
https://calendar.carleton.ca/undergrad/undergradprograms/computerscience/
https://carleton.ca/scs/current-students/undergraduate-students/help-selecting-courses/electives-and-prohibited-courses-2/
https://carleton.ca/scs/current-students/undergraduate-students/streams-and-programs/major-vs-honours-2/
https://science.carleton.ca/students/undergraduate-resources/deans-summer-research-internships/
https://carleton.ca/career/
https://roadmap.sh/