

**Course Information:**

- **Instructor:** Jason Hinek (Office 5332HP, Email: [jason.hinek@carleton.ca](mailto:jason.hinek@carleton.ca))
- **Office Hours:** TBA
- **TAs:** Available here: <http://people.scs.carleton.ca/~mjhinek/COMP1006>
- **Class:** Wednesdays and Fridays from 4:05pm – 5:25pm in room 103 Steacie Building

**Course Description:**

This is a second course in programming. It assumes that you are familiar with programming and builds on that knowledge by introducing object-oriented programming concepts. You will practice designing and testing various objects as well as use various abstract data types in a recursive and non-recursive manner. You will also learn to develop complete applications with graphical user interfaces using various Java FX window components and event handling. File I/O is also introduced as well as Exception Handling. Depending on time, other concepts such as network programming and graphics may also be discussed.

**Topics Covered:**

- The Java programming language
- Java Memory Management
- Defining object behavior
- Class hierarchies, inheritance, polymorphism
- Abstract Data Types (Lists, Queues, Stacks, Sets, etc)
- Graphical User Interfaces (using Java FX)
- The Model/View/Controller paradigm
- Menus and Dialog boxes
- Recursion with objects
- Exception handling & File I/O

**Course Objectives:**

This course will teach you how to make complete applications in JAVA. It builds on concepts that you learned from COMP1005. By the end of the course you should have a solid background in programming in JAVA, having covered all the fundamentals. Note that a **grade of C- or above must be earned in this course** in order to advance to 2<sup>nd</sup> year COMP courses. Please see the 2017-18 Undergraduate Calendar for prerequisites: <http://calendar.carleton.ca/undergrad/>.

**Course Notes:**

There is no textbook assigned to this course. Instead, there is an in-depth set of course notes which are available online.

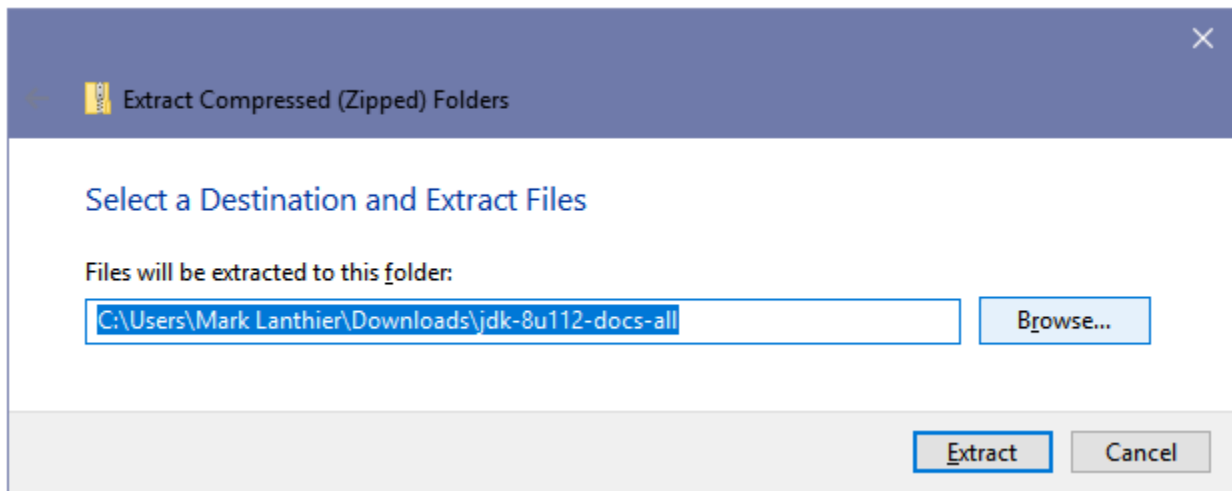
**Evaluation:**

| Component      | Weight     | Details                 | Due Date  |
|----------------|------------|-------------------------|---|
| Tutorials      | <b>10%</b> | best 10 out of 11       | Weekly, starting Monday, January 15 <sup>th</sup>             |
| Assignments    | <b>42%</b> | 7 at 6% each            | Weekly, due Mondays at 12 <b>noon</b>                         |
| In-Class Tests | <b>20%</b> | best 2 of 3 at 10% each | Friday Jan 25, Friday Feb 15, Friday Mar 22                   |
| Final Exam     | <b>28%</b> | no flexibility, do well | (to be announced February 16 <sup>th</sup> by the University) |

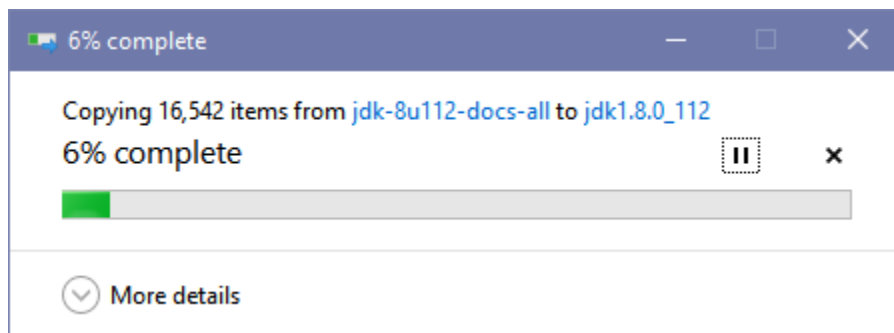
## Laboratory Software

You will be programming using the "Java Development Kit (JDK) SE 8u151" which is available from Oracle (although there may be a more recent update by the time you read this, so the 151 number may be different). A recent version of this software is installed on all laboratory computers so if you work at school ... you do not have to do anything tricky. If you work at home, you will probably want to download this free software on your machine there. Go to the following website to download the software:

1. Go to <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
2. Accept the license agreement and then download the file that corresponds to your operating system. For windows-based PCs, you can download either the 64-bit version (**jdk-8u151-windows-x64.exe**) or the 32-bit version (**jdk-8u151-windows-i586.exe**). Running the file will install JAVA onto your hard drive. Let it install to the default location.
3. Go to this webpage <http://docs.oracle.com/javase/8/docs/>. Scroll down the left side and select **JDK 8 Documentation** from the **Downloads** section. Click on the **Accept License Agreement** radio button under the **Java SE Development Kit 8u151 Documentation** section. Then click on the zip file called: **jdk-8u151-docs-all.zip** and select Save File. It will go to your downloads folder on your computer by default. Click on the **Accept License Agreement** radio button under the **JavaFX API Documentation** section at the bottom. Then click on the zip file called: **javafx-8u151-apidocs.zip** and select Save File. Both files will go to your downloads folder on your computer by default.
4. Unzip the **jdk-8u151-docs-all.zip** file to the directory that JAVA was installed into. To do this, when the following dialog box comes up, select the **Browse** button (note that version number will be different than in picture):

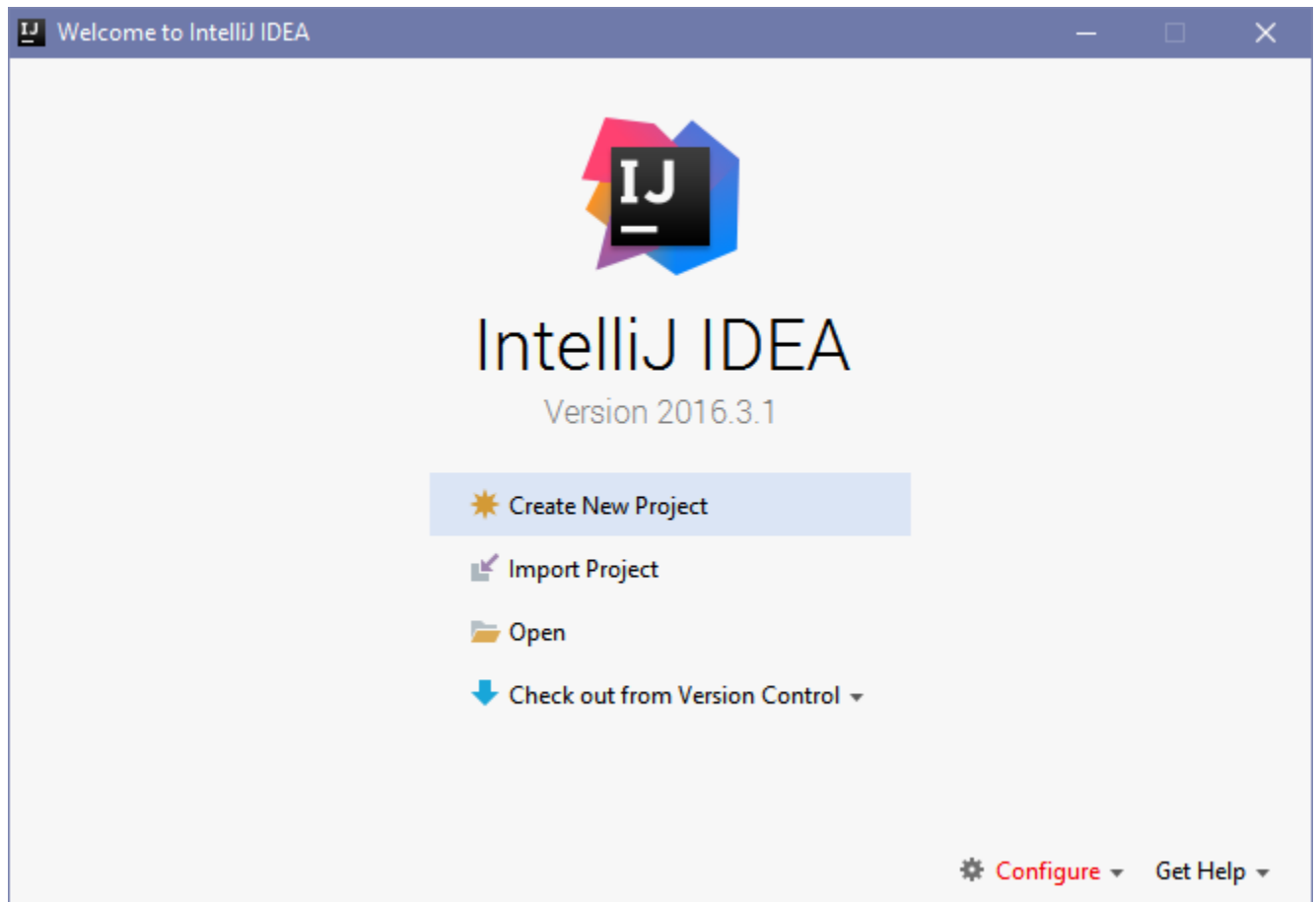


5. From the window that pops up, locate the folder that JAVA was installed into. On your windows machine, it will likely be: **C:\Program Files\Java\jdk1.8.0\_151** (see below ... version number will be different than in picture.). If you installed the 32-bit version, it will likely be this directory: **C:\Program Files (x86)\Java\jdk1.8.0\_151** instead. Let the files be extracted (it may take 10-15 minutes). You might have to click **Continue** when prompted for permission to install.



6. In the same way that you unzipped the java documentation file, unzip the **javafx-8u151-apidocs.zip** file to the directory that JAVA was installed into (i.e., after selecting Browse button, choose this directory **C:\Program Files\Java\jdk1.8.0\_151**). It will unzip much quicker to a folder called **api**.

7. Now any time that you want to browse through the JAVA classes, you can go to the **api** subfolder of the **docs** folder (i.e., **C:\Program Files\Java\jdk1.8.0\_151\docs\api**) and select the **index.html** file, which will open in your browser. You can bookmark it in your browser for quick access during the term. Similarly, you can browse the JavaFX classes by selecting the **index.html** file from the **api** folder that you unzipped that to (i.e., **C:\Program Files\Java\jdk1.8.0\_151\api**). You may want to bookmark that as well in your browser.
8. The above software is simply the JAVA compiler and interpreter (and other tidbits). We will also be using an editor to write/compile/run our JAVA code. We have chosen a specific editor (called IDE - Integrated Development Environment) for use in this course that you **MUST** use, since it has a provision for exporting and importing projects that the TAs will need in order to mark your assignments. The IDE that we will use is called IntelliJ IDEA and is freely available for download here: <https://www.jetbrains.com/idea/>. Click the **Download** button. The **Community** version is free and has everything that you will need for this course, so I suggest using that version. The software is available for Windows/MacOS and Linux. For windows, the file was called **ideaIC-2017.3.exe** (at the time that I wrote these instructions). Run the downloaded file to install it. Let it install to the default location (i.e., **C:\Program Files (x86)\JetBrains\IntelliJ IDEA Community Edition 2017.3**). Select the **64-bit launcher** or **32-bit launcher**, depending on which JAVA version you installed in step 2 above. Select **.java** under the **Create Associations** area. **DO NOT** select the checkbox for downloading the 64-bit JRE as shown at the bottom. Click **Next**, then click **Install ...** and wait, then click **Finish**.
9. You should have a desktop icon that you can run (looks something like what is shown here on the right). You can setup various plugins (if you know what you are doing), but you simply just need to select the **Skip All and Set Defaults** button. In time, you should see a window as shown below (version number will be different). Select **Configure** at the bottom right of the window. Then select **Settings**.



10. On the left side of the Default Settings dialog box, scroll down to the section called **Build, Execution, Deployment**. Select the **Java Compiler** option under the **Compiler** dropdown list. For **Project bytecode version**, select **1.8** from the dropdown list. Click the OK button. You can now close the program. In class (and during the first tutorial) you will learn how to create a project and write some JAVA code.

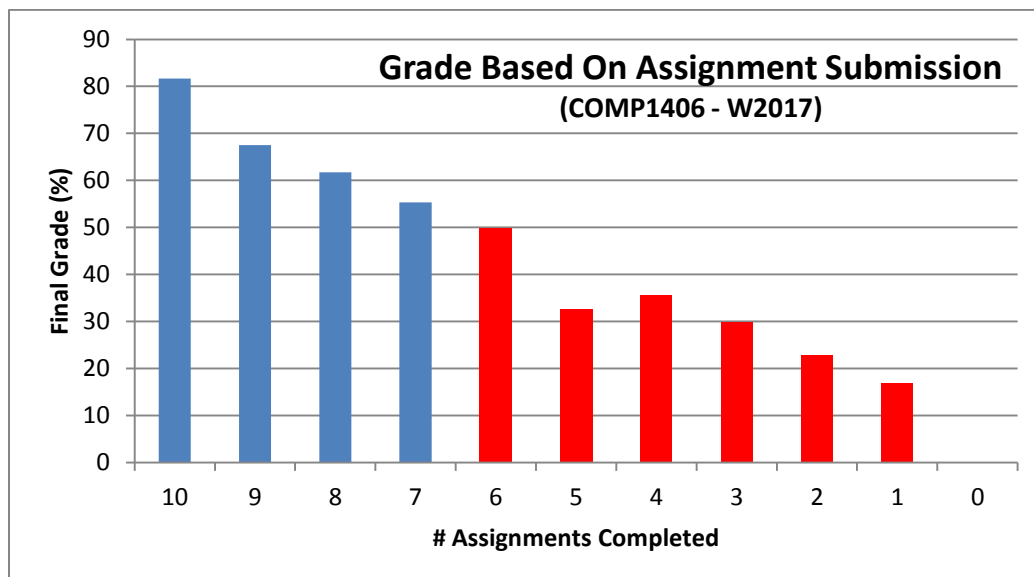
## In-class Tests

In class tests will be 40 minute closed-book and will occur in the 2nd half of a lecture. Each test must be handed back immediately when completed in order to be graded.

## Assignments

There will be assignments in this course which will be available on the course web page, which is not CULearn, but is here: (<http://people.scs.carleton.ca/~lanthier/teaching/COMP1406/>) . Do your best to complete all of the assignments, since this is where you actually learn the course material. Assignments must be handed in before or on the due date and time. **NO LATE ASSIGNMENTS WILL BE ACCEPTED.** You will be using Carleton's [CULearn](#) system to submit your assignments and view your grades throughout the term. For more information about how to do this, please go to <http://carleton.ca/culearnsupport/instructors/getting-started/>. Always keep a backup of your work, perhaps on a USB flash drive or by sending yourself an email with your assignment attached. That way you can access your code from anywhere. You should take the time to ensure that assignments are neat, legible and easy to understand. Any instructions required by the teaching assistants (for example any assumptions you made about the assignment) should be clearly indicated on a separate README.TXT file, included with the assignment. Remember, it is YOUR responsibility to demonstrate that you have understood and completed the assignment. A portion of your grade for assignments will be given for the readability of them and for your demonstration that you have completed the assigned tasks. The CULearn system also allows you to view your marks on-line. You should ensure that the posted marks are correct. Any complaints regarding assignment marks should be brought to the attention of the TA who marked it (only if the TA does not address the problem to your satisfaction should you bring the matter to the instructor). This should be done **no later than two weeks after the assignment has been graded**. After this time, no remarking will be done. Being "sick" on the day an assignment is due is not an excuse for not doing it. You have about 10 days to do each assignment. Start early and feel free to submit partially completed versions. That way, if you get sick, your partially completed version will be marked, and you will not get 0. **DO NOT** email your assignments to any TAs. If you have problems submitting, it is likely due to the fact that you missed the deadline by a few minutes. You will get 0, despite your hard work. So please, submit at least 15 minutes **BEFORE** the assignment is due ... consider yourself forewarned. If you are at home trying to submit and you cannot get it to work, again, you will be late with your assignment and it will not be accepted. If you are sick for an extended period of time, please inform the instructor (not the TAs) immediately. You will need to have official documentation of illness. At this point, it will be up to the instructor as to how to handle the situation. In regards to the in-class tests, you must attend and write them. If sick, you must inform the instructor via email the day before and you will need official documentation as well. If you miss one test, the other two will count for your grade.

As shown here, evidence from the past shows that your final grade is directly related to your submission of assignments:



## Cheating and Plagiarism (University Policies)

Sadly, every term, students are caught cheating on assignments (and sometimes tests) in this course. Copying on an assignment or a test is considered plagiarism:

***Plagiarism** - As defined by Senate, "plagiarism is presenting, whether intentional or not, the ideas, expression of ideas or work of others as one's own".*



### **Student Academic Integrity Policy:**

Every student should be familiar with the Carleton University student academic integrity policy. A student found in violation of academic integrity standards may be awarded penalties which range from a reprimand to receiving a grade of F in the course or even being expelled from the program or University. Some examples of offences are: plagiarism and unauthorized co-operation or collaboration. Information on this policy may be found in the Undergraduate Calendar.

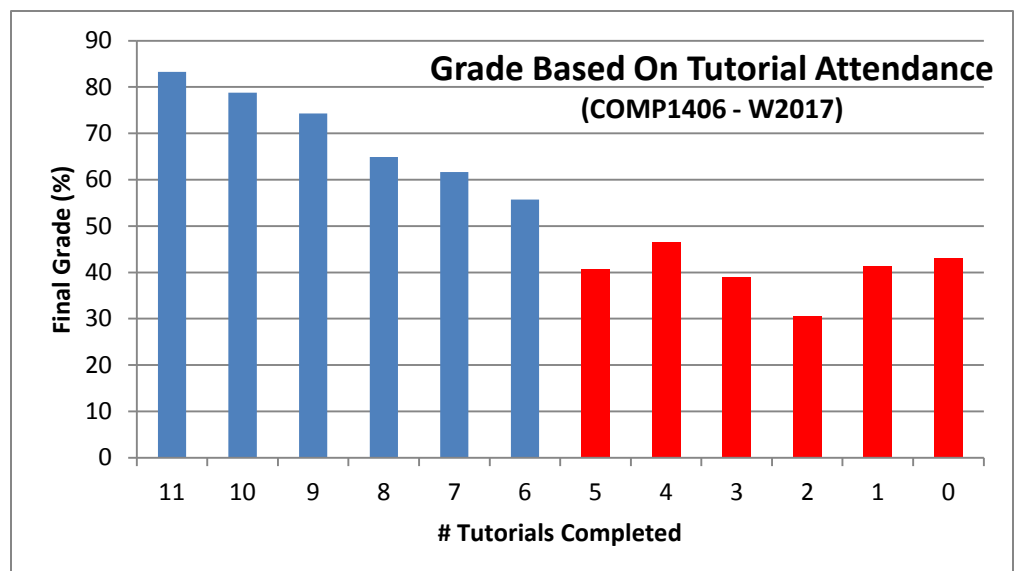
To be as clear as possible:

- You are not allowed to copy & edit any portion of another student's code.
- You are not allowed to give your code (or portions of it) to another student.
- You are not allowed to post full or partial assignment solutions on discussion boards or websites (e.g., facebook, etc..).
- You must work on your assignment on your own without collaboration with other students. If you need help, please see a TA or your instructor.

Your assignment will be compared with others in the class and if you are found copying (or if you give a copy of some of your code to someone else), such reported offences will be reviewed by the office of the Dean. In most cases, you will receive zero, have a permanent mark on your record and your final grade will likely be delayed by a few months, (making it difficult for you to register for the next term).

## Tutorials

There will be mandatory tutorials in this course which will be counted towards your final grade. You will be assigned 1% of your final grade each time you **attend AND participate** in the tutorial for the full 1.5 hours. While in the tutorial, you must work on the tutorial work provided. You may not work on your assignment in that lab while the tutorial is going on. Also, anyone who is not working on the tutorial must leave the room. Near the end of the tutorial, the TA will make a note of who is working on the tutorial and who is not. If the TA finds that you did not spend your time doing the tutorial work, then you will not get your grade, even if you are there for the full 1.5 hours. Also, you are not allowed to get a hold of the tutorial beforehand to work on it or complete it before your tutorial session. If you show up to the tutorial with it partially or completely done, you will not get your grade. So, to put it simply, come to the tutorial each week, do the work while you are there and then leave. You should plan to stay the full 1.5 hours each week. You **MUST** attend the tutorial that you are registered in. **Tutorials take place in 4155HP. Tutorials will begin on Monday, January 15th** (see course website for complete schedule). As shown here, evidence from the past shows that your tutorial attendance will affect your final grade:



## Undergraduate Academic Advisor

The Undergraduate Advisor for the School of Computer Science is available in Room 5302C HP, by telephone at 520-2600, ext. 4364 or by email at [undergraduate\\_advisor@scs.carleton.ca](mailto:undergraduate_advisor@scs.carleton.ca). The undergraduate advisor can assist with information about prerequisites and preclusions, course substitutions/equivalencies, understanding your academic audit and the remaining requirements for graduation. The undergraduate advisor will also refer students to appropriate resources such as the Science Student Success Centre, Learning Support Services and the Writing Tutorial Services.

## Equity Statements

You may need special arrangements to meet your academic obligations during the term. For an accommodation request the processes are as follows:

- **Pregnancy obligation:** write to me with any requests for academic accommodation during the first two weeks of class, or as soon as possible after the need for accommodation is known to exist. For more details visit the Equity Services website: <http://www2.carleton.ca/equity/>
- **Religious obligation:** write to me with any requests for academic accommodation during the first two weeks of class, or as soon as possible after the need for accommodation is known to exist. For more details visit the Equity Services website: <http://www2.carleton.ca/equity/>
- **Academic Accommodations for Students with Disabilities:** The Paul Menton Centre for Students with Disabilities (PMC) provides services to students with Learning Disabilities (LD), psychiatric/mental health disabilities, Attention Deficit Hyperactivity Disorder (ADHD), Autism Spectrum Disorders (ASD), chronic medical conditions, and impairments in mobility, hearing, and vision. If you have a disability requiring academic accommodations in this course, please contact PMC at 613-520-6608 or [pmc@carleton.ca](mailto:pmc@carleton.ca) for a formal evaluation. If you are already registered with the PMC, contact your PMC coordinator to send me your **Letter of Accommodation** at the beginning of the term, and no later than two weeks before the first in-class scheduled test or exam requiring accommodation (*if applicable*). **Requests made within two weeks will be reviewed on a case-by-case basis.** After requesting accommodation from PMC, meet with me to ensure accommodation arrangements are made. Please consult the PMC website ([carleton.ca/pmc](http://carleton.ca/pmc)) for the deadline to request accommodations for the formally-scheduled exam (*if applicable*).
- You can visit the Equity Services website to view the policies and to obtain more detailed information on academic accommodation at [carleton.ca/equity/](http://carleton.ca/equity/)

## **Medical Certificate**

The following is a link to the official medical certificate accepted by Carleton University for the deferral of final examinations or assignments in undergraduate courses. To access the form, please go to [carleton.ca/registrar/forms/](http://carleton.ca/registrar/forms/)