CS 120: Introduction to Computing
Fall Semester 2014
http://www.cs.clarku.edu/~cs120/

Mathematics and Computer Science Department

Class Meetings @ BP326/310 (TBA):
Section 01: Tue, Thurs 10:25-11:40am
Section 02: Tue, Thurs 4:15-5:30pm
Section 03: Tue, Thurs 6:00-7:15pm

Richard Mitchell, Instructor, Section 1
rimitchell@clarku.edu
Office hours @ BP334:
Tue 1-4pm, Thurs 9:30-10:30am
and by appointment

Frederic Green, Instructor, Section 2
fgreen@clarku.edu
Office hours @ BP334:
Tue 1-4pm, Thurs 10:30am-12:30pm
and by appointment

Kevin Ankiewicz, Instructor, Section 3
kankiewicz@clarku.edu
Office hours @ BP335:
Tue 5-6pm, Thurs 7:20-9:20pm
and by appointment

Lab Meetings @ BP310:
Section 01: Tue 2:50-4:05pm
Section 02: Thurs 2:50-4:05pm
Section 03: Tue 7:20-8:35pm

Teaching Assistants
TBA
Please include “CS120” in the subject.
Tutoring hours held at CS lab BP 310 Mon-Thurs nights.

Course Description
This course will be an introduction to computer science and object-oriented programming through the vehicle of programming in the Java language and with Alice. The course will focus on the skill of programming, to be sure, but will also attempt to introduce the student to the discipline of computer science. It serves as a stand-alone, rigorous sample of computer science, suitable for math and science majors, and also as the first course in the CS concentration.

Our emphasis in teaching programming in this course centers on the following:
1. Developing elegant and efficient code from an abstract specification;
2. Using building blocks of program development.
3. Literate programming (writing a program that can be read by a human as well as a machine);
4. Programming methodology, which involves
   • Planning out the design of a program using object-oriented design and appropriate features of Java
   • Methodical and efficient development of the implementation using step-wise refinement and incremental testing and debugging (using appropriate debugging tools)
   • Being able to convince yourself of the correctness of the implementation by logical reasoning.

Teaching/Learning Method
Most topics will employ a flipped-classroom model.
You will be responsible for content-acquisition before coming to class, and we will use in-class time for learning through programming assignments.
For each topic, your preparation before coming to class will include:

- Read the relevant textbook sections to be introduced to vocabulary, concepts, and examples.
- Watch one or several short videos that present examples relevant to the topic.
- Re-create the examples on your own, solving any syntax or logical issues you encounter. Bring your questions about the examples for discussion at the start of class.
- Take a short online quiz on Moodle to prove to yourself that you have acquired the basic concepts, vocabulary and syntax.


Textbooks

*Alice in Action, Computing Through Animation* by Joel Adams

*Java, An Eventful Approach* by Kim Bruce, Andrea Danyluk, and Thomas Murtagh

Other online readings and tutorials will be posted to the schedule page.

Software

For the applied parts of the course, we will be using the following software.
All of these are available in the CS computer lab (BP 310):

- Alice – An environment for learning programming through animation.
- Java is a programming language, which we will use inside an environment called BlueJ.

Grading

The following percentages are tentative and may be changed at my discretion at any time:

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attendance, participation, professionalism</td>
<td>5%</td>
</tr>
<tr>
<td>In-class and Lab Assignments</td>
<td>30%</td>
</tr>
<tr>
<td>Programming Projects (2)</td>
<td>10%</td>
</tr>
<tr>
<td>Midterm Exam(s)</td>
<td>20%</td>
</tr>
<tr>
<td>Final Exam (written)</td>
<td>25%</td>
</tr>
<tr>
<td>Lab Exam(s)</td>
<td>10%</td>
</tr>
</tbody>
</table>

Withdrawing from the Course

If you feel that you want to drop or withdraw from the class, please come talk to me about it as early as possible; I want to help you succeed, but you need to ask for help.
Add/Drop period ends Wednesday September 3, 2014 at 11:59pm
The last date to withdraw and receive a “W” grade is Friday, October 31, 2014.

What You Need to Know About Computer Programming

I believe anyone can succeed at learning to program. This is a first course in computer programming, and there are no formal pre-requisites. The only expectation of students’ computer skills before taking this class is to be comfortable with using email, web browsing, and copying and pasting text. In addition, familiarity with high-school level algebra is assumed.
In addition, you will need time, and this is more important than you can imagine. Many people believe that computer programming is extremely difficult, and that code is written in some arcane syntax understandable only by experts. Although some parts of the process are indeed complex, most of the source code required for homework assignments can be easily understood.

So, what makes programming so hard? It’s not the difficulty: It’s the time required to achieve any decent results. The assignments will take time, so make sure you have plenty of it. Adapted from text in “Core Techniques and Algorithms in Game Programming,” Daniel Sanchez-Crespo Dalmau, and Aaron Stevens (BU)

Policies and Miscellaneous

The official administrative business of this class will be conducted by email. Grade questions/disputes, explanation of absence, etc. will be processed via email so that we both have a written record of what was agreed. Feel free to discuss in person but an email follow-up is required for the official record.

Attendance and discussion/asking questions are expected and will be reflected in your grade. If you must be absent, please email me in advance to let me know why you won’t be in class, and to let me know what you will do to keep up with the assignments. CS120 is not a correspondence course.

Computer Use Policy
Most of this course will be hands-on “learning by doing”. You are expected to attend class and work on the assignments for this class during that time. Please do not use this time to work on other homework, play games, check Facebook, or surf the web. Such activities can be distracting to other students.

Programming projects are due on the date stated on the assignment (to be posted on web).

- Projects received within 0-24 hours of the deadline will be accepted with a 10% penalty.
- Projects received within 24-48 hours of the deadline will be accepted with a 20% penalty.
- Projects received more than 48 hours past the deadline will not be accepted or graded.

Since the in-class work is meant to encourage a self-paced learning environment, there is a bit more flexibility on those assignments. However, is is very important to stay up-to-date with your work as best you can, and work on all assignments as soon as they are given so you can ask questions in class and get assistance in the labs and tutoring hours.

Students are responsible for ensuring that assignments are correctly submitted. If you have a question or problem, seek help from CS120 staff immediately.

No special make-up work will be accepted after the end of the semester. Don’t even ask.
In the event of a documented major medical problem, a grade of Incomplete will be given pending the submission of complete work. However, make up work “to improve one’s grade” will not be accepted.

It is the student’s responsibility to retain all papers, quizzes, and exams that have been graded and returned. Should these original documents not be available in the event of a grade dispute, I will need to defer to the own records.

Grades are not negotiable. Don’t even ask – just do the work and you’ll get the grade you deserve. Of course, please bring any clerical grading errors to my attention and I will gladly fix them.
Plagiarism, Collaboration, and Collusion

Although you are strongly encouraged to solve problems independently in classroom assignments, a certain amount of collaboration in in-class assignments and labs is also encouraged, since you can learn a great deal from each other, and it fosters a team spirit.

On the other hand, when specified, CS120 assignments are independent work. This will be true of the projects and lab exams, and it is important to understand what this means.

It is the student’s responsibility to know and understand the Clark University Academic Integrity policy, which is within the Academic Advising Handbook (The Blue Book) available at the Academic Advising Center.

In addition to the definition of plagiarism in the handbook, with respect to CS120, plagiarism is specifically defined to include (but is not limited to) the following:

- collaboration on the solutions/code you write
- copying any part of someone else’s assignment/program, even if you have permission and/or have modified the code
- sharing or giving your assignment/code or even a subset of your assignment/code to another student to review
- reviewing another student’s solution (including from past semesters)
- reviewing solutions on the internet

The University takes acts of cheating and plagiarism very seriously, and suspected cases are submitted directly to the Dean of Academic Advising; violators may be suspended or fail the course.

What is acceptable cooperation?
Cooperation is recommended in understanding programming concepts and system features. You are encouraged to discuss the labs, the homework problem statements and expected output, and to seek and receive help with Alice, the Java Programming Language, BlueJ IDE, and other tools.

However, when doing work that is designated as independent, each student must write his or her own solution/code and other deliverables independently.