snap-syllabus

COR100 Snap! Programming Syllabus

Marcus Birkenkrahe

July 3, 2022

Contents

1 General Course Information

- Meeting Times: Tuesday from 11:00-11:50 am

- Meeting place: Derby Science Building Computer Lab 209

- Professor: Marcus Birkenkrahe

- Professor's Office: Derby Science Building 210

- Phone: (870) 307-7254 (office) / (501) 422-4725 (private)

- Office hours: Monday through Friday after class (see Canvas)

- Textbook: Joshi A B (2018). Learn CS Concepts with Snap!

About reading the book / further studies

There aren't a lot of books about Snap! because the language is easy to learn and needs to be experienced rather than read about. As is common for programming languages, there is a detailed reference manual available from the Berkeley support site (`https://snap.berkeley.edu/`).

Snap! looks very similar to Scratch for which many courses and books are available, most of them pitched for kids and young adult learners. Snap! is in a different class - it's a full-fledged programming language ready for sophisticated data science and media informatics work.

For further studies, I recommend the Snap! courses offered for free by openSAP, the online university of SAP, a large German software company that employs Snap!'s main developer and supports the development of the language[1].

Objectives

In this course, you will complete a series of game and animation projects using the visual, drag-and-drop programming language Snap! You will learn basic computer and data science principles, and understand how computers help us control the world around us. You will graduate from mere consumer and user to powerful programmer while playing around with blocks on the screen. Acquiring programming skills will help you develop your critical thinking. We will meet weekly for an hour only of practical problem solving and interactive exercises. In between classes, you will work on small assignments, and you will read short textbook chapters or watch short instructional videos. You'll get to present your own work and (optionally) publish your finished projects for the whole world to see, and if things go well, perhaps we'll go to Snap!Con in California next year!

Target audience

This course is for you if you're curious about visual programming, if you don't know if working with computers is for you, or if you're interested in media, the arts, data, or in developing games. Snap! is also a great stepping stone to move on to higher languages like Python, R, C or C++. If you already have some knowledge of computing, programming or media computation, you'll

---

[1]I wonder why! SAP's main business is in Enterprise Resource Planning software - these are massive systems that support whole companies and all their processes. But SAP is actively pursuing data science and predictive AI, which is why they "acquired" Snap! (not really, because Snap! is FOSS - Free and Open Source Software).

be right at home and can add visual languages to your skillset. The course also serves as an easy entry into computer or data science degree programs.

Student Learning Outcomes

Students who complete YearOne COR 100 "Snap! Programming Playground", will be able to:

- Create exciting games, animation and media computation

- Learn computer and data science principles by playing with data

- Acquire basic sequential programming skills

- Understand the relationship of humans and machines better

- Develop your critical thinking skills

- Know how to effectively present assignment results

Course requirements

No prior knowledge required. Some knowledge of, and experience with computers is useful but not critical. Curiosity is essential. You will gain data literacy skills by taking this course. The course will prepare you for further studies in computer and data science, or in other disciplines that use modern computing, i.e. every discipline, from accounting to zoology).

Grading system

DESCRIPTION	IMPACT
Quizzes	25%
Class assignments	25%
Programming assignments	25%
Independent study project	25%

You should be able to see your current grade at any time using the Canvas gradebook for the course.

Grading table

This table is used to convert completion rates into letter grades. For the midterm results, letter grades still carry signs, while for the term results, only straight letters are given (by rounding up).

%	Midterm Grade	Final Grade
100-98	A+	
97-96	A	A (passed -
95-90	A-	very good)
89-86	B+	
85-80	B	B (passed -
79-76	B-	good)
75-70	C+	
69-66	C	C (passed -
65-60	C-	satisfactory)
59-56	D+	
55-50	D	D (passed)
49-0	F	F (failed)

Quizzes (25%)

- Complete in class

- Recall last chapter or section of class

- Read relevant textbook chapters

Class assignments (25%)

- Complete assignments in class (participation)

- Upload completed assignments (homework)

- Be ready to present your results in class

Programming assignments (25%)

- Solve programming problems in or outside of class

- Complete assignments outside of class if necessary

- Be ready to present your solutions

Independent study project (25%)

- Study a problem of your choice (options given)

- Present prototype results for midterm grades

- Present final result at end of course

2   Standard Policies

Honor Code

All graded work in this class is to be pledged in accordance with the Lyon College Honor Code. The use of a phone for any reason during the course of an exam is considered an honor code violation.

Class Attendance Policy

Students are expected to attend all class periods for the courses in which they are enrolled. They are responsible for conferring with individual professors regarding any missed assignments. Faculty members are to notify the Registrar when a student misses the equivalent of one, two, three, and four weeks of class periods in a single course. Under this policy, there is no distinction between "excused" and "unexcused" absences, except that a student may make up work missed during an excused absence. A reminder of the college's attendance policy will be issued to the student at one week, a second reminder at two weeks, a warning at three weeks, and notification of administrative withdrawal and the assigning of an "F" grade at four weeks. Students who are administratively withdrawn from more than one course will be placed on probation or suspended.

Disabilities

Students seeking reasonable accommodations based on documented learning disabilities must contact Interim Director of Academic Support in the Morrow Academic Center at (870) 307-7016.

Harassment, Discrimination, and Sexual Misconduct

Title IX and Lyon's policy prohibit harassment, discrimination and sexual misconduct. Lyon encourages anyone experiencing harassment, discrimination or sexual misconduct to talk to Danell Hetrick, Title IX Coordinator

and Interim Vice-President for Student Life, or Sh'Nita Mitchell, Title IX Investigator and Associate Dean for Students, about what happened so they can get the support they need and Lyon can respond appropriately. Lyon is legally obligated to respond to reports of sexual misconduct, and therefore we cannot guarantee the confidentiality of a report, unless made to a confidential resource (Chaplain, Counselor, or Nurse). As a faculty member, I am required to report possible Title IX violations and must provide our Title IX coordinator with all relevant details. I cannot, therefore, guarantee confidentiality.

Details

Details specific to this course may be found in the subsequent pages of this syllabus. Those details will include at least the following:

- A description of the course consistent with the Lyon College catalog.

- A list of student learning outcomes for the course.

- A summary of all course requirements.

- An explanation of the grading system to be used in the course.

- Any course-specific attendance policies that go beyond the College policy.

- Details about what constitutes acceptable and unacceptable student collaboration on graded work.

3   Course specific information

Assignments and Honor Code

There will be several assignments during the summer school, including programming assignments and multiple-choice tests. They are due at the beginning of the class period on the due date. Once class begins, the assigment will be considered one day late if it has not been turned in. Late programs will not be accepted without an extension. Extensions will **not** be granted for reasons such as:

- You could not get to a computer

- You could not get a computer to do what you wanted it to do

- The network was down

- The printer was out of paper or toner

- You erased your files, lost your homework, or misplaced your flash drive

- You had other coursework or family commitments that interfered with your work in this course

  Put "Pledged" and a note of any collaboration in the comments of any program you turn in. Programming assignments are individual efforts, but you may seek assistance from another student or the course instructor. You may not copy someone else's solution. If you are having trouble finishing an assignment, it is far better to do your own work and receive a low score than to go through an honor trial and suffer the penalties that may be involved.

  What is cheating on an assignment? Here are a few examples:

  - Having someone else write your assignment, in whole or in part
  - Copying an assignment someone else wrote, in whole or in part
  - Collaborating with someone else to the extent that your submissions are identifiably very similar, in whole or in part
  - Turning in a submission with the wrong name on it

    What is not cheating? Here are some examples:

    * Talking to someone in general terms about concepts involved in an assignment
    * Asking someone for help with a specific error message or bug in your program
    * Getting help with the specifics of language syntax or citation style
    * Utilizing information given to you by the instructor

  Any assistance must be clearly explained in the comments at the beginning of your submission. If you have any questions about this, please ask or review the policies relating to the Honor Code.

  Absences on Days of Exams:

  Test "make-ups" will only be allowed if arrangements have been made prior to the scheduled time. If you are sick the day of the test, please e-mail me or leave a message on my phone before the scheduled time, and we can make arrangements when you return.

Attendance policy

You must attend a minimum of 80% of non-cancelled meetings, and no consecutive meetings without risking a fail. This means that you can miss 3 meetings (not in consecutive weeks) without any issues. If you miss 4 meetings, you fail the class. Any missed meetings result in an "Early Alert" report.

Schedule and session content

NO	DATE[2]	TESTS[3]	TEXTBOOK[4]	ASSIGNMENT[5]
1	Tue-16-Aug		1.1 First look at snap	
2	Tue-23-Aug	Quiz 1	1.2 Sequence and sounds	Program 1
3	Tue-30-Aug	Quiz 2	1.3 Looping and costumes	Program 2
4	Tue-06-Sept	Quiz 3	1.4 XY geometry and motion	Program 3
5	Tue-13-Sept	Quiz 4	1.5 Animation mini project	Program 4
6	Tue-20-Sept	Quiz 5	1.6 More motion and broadcasting	Program 5
7	Tue-27-Sept	Quiz 6	1.7 Project animation	Program 6
8	Tue-11-Oct	Quiz 7	2.1 Events, reset scripts, concurrency	Program 7
9	Tue-18-Oct	Quiz 8	2.2 Keyboard interaction, conditionals	Program 8
10	Tue-25-Oct	Quiz 9	2.3 Project helicopter	Program 9
11	Tue-01-Nov	Quiz 10	3.1 Mouse interaction	Program 10
12	Tue-08-Nov	Quiz 11	3.2 Variables	
13	Tue-15-Nov	Quiz 12	3.3 Project game of maze	
14	Tue-22-Nov	Quiz 13		Presentations
15	Tue-29-Nov			Presentations

4 References

- Huegle J/Moenig J (2018). Get coding with Snap!. URL: open.sap.com.

- Huegle J/Moenig J (2020). From media computation to data science. URL: open.sap.com.

---

[2] Fall break: 1-4 October. Last day of fall classes: 2 Dec.

[3] Each quiz tests your understanding of the previous lesson.

[4] Joshi, Learn CS Concepts with Snap! (2018).

[5] Weekly in-class assignments supplemented by program assignments for home completion and upload at your snap.berkeley.edu project location. The last two classroom sessions are reserved for presentation of your projects.

- Joshi A B (2018). Learn CS Concepts with Snap!. URL: abhayjoshi.net.

- Joshi A B (2020). Adventures in Snap! Programming. URL: abhayjoshi.net.