CS111 Lab Extra Credit Assignments (Spring’2016)

The purpose of this set of extra credit exercises is not just to let you have a chance to make up for the points in the lab. I am hoping that even without the incentive of extra credits, you will try most if not all of them out.

Purposes:
1) Explore different resources available in learning and know where and how to appropriately ask for help in forums or online communities
2) Get connected to the bigger IT communities in NYC and participate some of the meetup events
3) Provide extra information on some of the upcoming topics that you will come across in this class or further down the road for your degree
4) Provide you a guide on how to pick up new skills as a lifetime learner 😊

Xunzi (340 - 245 BC) “ru xiao pian” 儒效篇
Before we start...

At the beginning of every semester, professor Ryba gives an introduction to the course stating that “every student can get an A in the course by learning the materials.” But it hasn’t happened yet in all the years in his teaching for the course. So let’s help him to see that becomes reality as much as possible by striving to do our best.

Even if you did not get a decent first mid-term score, or you never see yourself as an A student. I want you to not just pass but to excel in the course. But in order to get there, we need to get some of you to believe that it is possible. So let’s have a little breakthrough by watching a video by Tony Robbins below:

Tony Robbins 2016. Start your New Life NOW!! - YouTube

If you never see yourself as an A student, you will definitely need to sign up for the course below and get yourself equipped with the right tools and study techniques. Even if you are an A student already, I think it doesn’t hurt to get a refresher. 😊

Learning How to Learn: Powerful mental tools to help you master tough subjects - University of California, San Diego | Coursera

Instructors

Barbara Oakley
University of California, San Diego
@barbaraokley

Terrence Sejnowski
University of California, San Diego
@sejnowski
CS111 Lab EC1

1) Register for the Google Code Jam and participate in the qualification round on Apr 8, 2016. [https://code.google.com/codejam](https://code.google.com/codejam)

2) Take a screenshot to prove that you have participated (include your login name and the actual progress of the contest (top participants are going to show up on the left side of the menu bar.)
CS111 Lab EC2
1) Register for the self-paced CS50 course at edx.org and watch the introduction video from professor Malan.
   CS50: Introduction to Computer Science | edX

2) Watch Nate’s video on binary, if not too familiar:
   https://www.youtube.com/watch?v=hacBFrgtQjQ

3) And next with Nate’s video on ASCII:
   https://www.youtube.com/watch?v=UPlR4eMMCmI

4) Learn how to do bubble sort:
   https://www.youtube.com/watch?v=8Kp-8OGwphY

5) Answer the questions in EC2 in Blackboard. (0.5 points)
CS111 Lab EC3

Step 1: Go through the tutorial on Linux commands written by Ryan Chadwick. Pay special attention to section 4, 8, 9, 10, 11, 12 and section 13 discussion on “Why the ./?“.

Linux Tutorial

Step 2: Read up a little bit about Bash Scripting Tutorial and awk

Step 3: Answer the questions in EC3 in CUNY Blackboard. (1 point)
CS111 Lab EC4 (maximum 1 point)

Part 1) Check out some of the videos on:
https://www.khanacademy.org/computing/computer-programming/meet-the-computing-professional

Part 2) Check out some of the IT companies culture on:
http://devpost.com/teams

Part 3) Suggest at least one new resource that covers similar kind of information above on piazza forum

Part 4) Join a meetup (CS related) and attend an event (need to provide a proof that you have attended by getting a picture of yourself attending the event)
CS111 Lab EC5

Step 1: Learn how to use a version control system called Git by doing the exercises from

or

b) [https://www.lynda.com/Git-training-tutorials/1383-0.html](https://www.lynda.com/Git-training-tutorials/1383-0.html)

Note: You can get a free access to lynda.com training videos through [NYPL](https://www.nypl.org) or [Brooklyn Public Library](https://www.bpl.org).

Step 2: Answer questions in EC5 in CUNY Blackboard. (1 pt)

Below, I have included a screenshot from a presentation by Václav Rajlich on [ACM SIGSOFT Webinar: "Teaching Future Software Developers".](https://www.acm.softeam.org/sigsoft) Keep this in mind as you learn additional tools in the future.

### Technologies portfolio

- **Core language** (C++, Java, …)
- **IDE** (Visual studio, Eclipse, …)
- **GUI** (Swing, MFC, …)
- **Database** (SQL, Hibernate, …)
- **Testing** (Abbot, JUnit, …)
- **Modeling** (UML, XML, …)
- **Intra-team communication** (Wiki, …)
- **Version control** (CVS, Subversion, …)

© 2015 Václav Rajlich
Name your file **ec6.cpp**

1. Your program asks the user for six integers and stores them in an array. Then, it asks for six characters and stores them in a second array.
2. The program then prints a bar chart made of the characters provided (with the heights corresponding to the integers).
3. Lastly, a sorted bar chart (lowest to highest) should be printed.

Use arrays and functions. Recursion is not necessary.

Download a copy of **ec6.out** and run it to generate sample output and understand how the program works.

**Feel free to use any of the programs on Professor Ryba’s site as reference, though almost all would need to be modified to work correctly in this program.**

Here are the title lines of the functions I used in my version:
void printChart(int h[], char c[], int cap, int max);
void swap(int h[], char c[], int a, int b);
void sort(int h[], char c[], int cap);
int minIndex(int h[], int low, int high);
int getMax(int h[], int cap);
Sample Run:
Please enter six non-negative integers.
Integer #0: 3
Integer #1: 5
Integer #2: 2
Integer #3: 1
Integer #4: 7
Integer #5: 6
Please enter six characters.
Character #0: a
Character #1: b
Character #2: c
Character #3: f
Character #4: e
Character #5: d
Unsorted bar chart
   e
   d
   b   e   d
   b   e   d
a b   e   d
a b c   e   d
a b c f e d
Sorted bar chart
   e
   d   e
   b d e b
   d e a
   b d e
   c a b d e
f c a b d e
CS111 Lab EC7 (0.5 point for submission **accepted by the online judge**)

Step 1: Sign up for an account at [https://leetcode.com](https://leetcode.com)

Step 2: Write your solution for the following problem

**Palindrome Number | LeetCode OJ**

```cpp
class Solution {
public:
    bool isPalindrome(int x) {
        //Start writing your solution in here
    }
};
```

Test and see how your code is doing against a test case

When you are ready, you can submit and see if it passes all test cases

Step 3: Once your solution passes the test, submit a copy of your code in Blackboard as an attachment. The name of the file should be `yourname_ec7.cpp`. For example, I will call my submission `TseChi_ec7.cpp`. 
CS111 Extra Credit#8 (2 points)

Name your file ec8.cpp
1. Your program reads in the data from ec8.txt.
2. The program then calculates CS111 lab score using the following formula:

   \[ \text{labScore} = \text{hwSum} + \text{quizSum} \]

   - \( \text{hwSum} \) is the sum of the best 10 out of 13 homework scores
   - \( \text{quizSum} \) is the sum of the best 10 out of 12 quiz scores

Note: If the final lab score is over 30, it should set the final lab score as 30.

3. Lastly, name and final lab score should be saved to a file named labScore.txt.

Output saved in labScore.txt:
Student1 21.98
Student2 22.86
Student3 9.83
Student4 19.06
Student5 21.16
Student6 22.73
Student7 24.68
Student8 18.41
Student9 14.56
Student10 27.24

EC8.txt input file columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Test Scores (12 of them)</th>
<th>Homework Scores (13 of them)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student1</td>
<td>0.8 0.75 0 0.25 0 0.27 0.4 0.4 0.53 0.58 0 0.6</td>
<td>2 1 1.7 1.9 1.5 1.5 1.8 1.3 0 1.7 1.8 1.6 1.9</td>
</tr>
<tr>
<td>Student2</td>
<td>0.9 0 0.7 0.66 0.4 0.69 0.55 0 0.8 0.63 0.53 0.1</td>
<td>2 1.8 1.9 2 1.2 0 1.7 1.5 0 0 1.3 1.8 1.7</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>