



California State University, Sacramento  
College of Engineering and Computer Science

## Computer Science 35: Introduction to Computer Architecture

Spring 2018 – Lab 5 – *Magic Alphabet Soup*

---

### Overview

Ah! You enjoy every moment of being a student at the famous **Hogwarts School of Witchcraft and Wizardry**. This noble school has produced some of the finest magical students the World has ever seen.

At Hogwarts, everything is magical – from the paintings, to the walls, to even your nightly soup. The elves, in the Hogwarts kitchens, have prepared a wonderful version of the classic **alphabet soup**. However, there is a catch. Before you can enjoy your dinner, you have to guess a secret letter. If you don't, the soup won't let you eat it!



### Guessing the Letter

This game is also quite easy. The soup (computer) will select a lowercase letter from **a** to **z**. Then the student will attempt to guess it. After each guess, the soup tells the student whether their guess is too high or too low. Once they get the letter correct, the game is complete... and bon appetite!

Basically, you are going to write a loop that will continue until the guess is equal to the correct answer. Inside the loop, you need to check if the correct answer is too high or too low and display a message. The exact wording is up to you. Make sure to print a third message when the loop is complete.

### Part 1

For the first part, you want to generate the secret letter. Letters are just ASCII characters which, in turn, are just bytes – nothing more. You can compare two characters with each other just like any other number.

#### **Read the documentation!**

The csc35.o object library contains a subroutine called "random". Pass the range of numbers into %rax. It will return a random number from 0 to n-1 into %rax. Also look at an ASCII table to figure out the value of a lowercase **a**. Then, see if you can generate a random number between **a** and **z**.

You must figure this out on your own. I will not provide any help except to open the CSC35 library documentation. Make sure this part works before proceeding to Part 2.

## **Part 2**

This is the main game.

1. Loop until the user guesses the secret letter. The loop must check their guess against the correct answer.
2. If the user guesses too low or too high, print text telling the user.
3. When the game is over, print text congratulating the user.

## **Example**

Your solution doesn't to look exactly like the example below. User input is displayed in **blue** (this is for readability, you don't have to make the input blue). For this example, the system selected **g** as the secret letter.

```
I am your delicious Hogwarts Alphabet Soup.  
Before you can eat me, you must guess my letter!  
  
Guess: c  
You are too low  
Guess: m  
You are too high  
Guess: h  
You are too high  
Guess: f  
You are too low  
Guess: g  
  
Correct! You may now eat me!
```

## **Requirements**

You must think of a solution on your own. The requirements are as follows:

1. Display an introductory message.
2. Create a random letter
3. Loop until they enter the correct answer
4. Display a message if they are too high or too low in there guess.
5. Display a message when they guess correctly

## **Extra Credit**

For an extra 5 points, make it so the program can handle both uppercase and lowercase input. So, an uppercase "A" should match a lowercase "a". This is not difficult, if you plan ahead.

## UNIX Commands

### Editing

Action	Command	Notes
Edit File	<b>nano</b> <i>filename</i>	"Nano" is an easy to use text editor.
E-Mail	<b>alpine</b>	"Alpine" is text-based e-mail application. You will e-mail your assignments it.
Assemble File	<b>as -o</b> <i>objectfile asmfile</i>	Don't mix up the <i>objectfile</i> and <i>asmfile</i> fields. It will destroy your program!
Link File	<b>ld -o</b> <i>exefile objectfiles</i>	Link and create an executable file from one (or more) object files

### Folder Navigation

Action	Command	Description
Change current folder	<b>cd</b> <i>foldername</i>	"Changes Directory"
Go to parent folder	<b>cd</b> <b>..</b>	Think of it as the "back button".
Show current folder	<b>pwd</b>	Gives a file path
List files	<b>ls</b>	Lists the files in current directory.