



California State University, Sacramento
College of Engineering and Computer Science

Computer Science 35: Introduction to Computer Architecture

Fall 2022 – Lab 6 – *Ravenclaw Tower*

Overview

The Sorting Hat has been sitting on your head for what feels like an eternity. The Hat, befuddled, has been in arguing with itself for several minutes and becoming more irritated with every passing moment.

"Okay, okay, okay.... let me start over... this is a good mind... yes... so Ravenclaw... "

"But, but... a programmer. Yes, a programmer. Sneaky... a Slytherin.

"But... programmers have to do lots of hard work in assembly... and they like it... how odd these muggles. So, Hufflepuff, yes."

"Oh, but they did decide to study computer science in the first place! So brave! So stupid... that's Gryffindor."

What to do? What to do?" The Hat groans to itself and starts clicking its tongue impatiently.

But you are taken aback. "Wait... um... Mr. Hat... how are you clicking your tongue? You don't have a tongue?"

The hat constricts uncomfortably around your head.

"Oh, do you think it's funny that I don't have a tongue, do you? Think yourself a smarty-pants, don't you? Fine, so be it."

Then, to the entire room, that Hat yells "RAVENCLAW!". Before you realize what just happened, the Hat is yanked off your head and you join the Ravenclaw table to applause... albeit less enthusiastic than other students received. After the feast, you are cornered by the Ravenclaw Prefect. You can tell they are a prefect by the highly polished "P" on the robes.

"Hello, my name's Joe Gunchy... a pleasure to meet you." Joe extends a hand and shakes yours vigorously. "I'm the prefect for Ravenclaw House. So, I'm in charge of keeping everyone in-line and other managerial tasks. The hat took quite a bit of time on you."

You get a sinking feeling in your stomach. Did it take that long? Are other Ravenclaws, and other students for that matter, talking behind your back? Joe notices your pained look and continues.

"Okay, you seem like a muggle-born... ah yes... Okay, let me give you the lowdown. Each of the houses has a common room and attached dormitories. That's where you will be staying. Each common room has some... well... security to get in. The Hufflepuff Common Room is located down near kitchens. They have a special knock. The Slytherins and Gryffindors use a password. They are down in the dungeons and up in a tower, respectively, that is. Ours, is far more interesting."



You take a moment to let the Joe's long, rambling, description to sink it. "Ah, I think I got it. Why is ours interesting"?

"Well," Joe continues enthusiastically, "the Ravenclaw Common Room is quite different. Like Gryffindor, it is located in a tower. The highest tower, in fact. However, we don't use a password. Too simple. Instead, the door challenges your ability to think. Like our house motto: wit beyond measure is man's greatest treasure!"

You shrug. "Oh, great".

Joe smiles. "The test can range from math problems to abstract riddles. It's really fond of math problems at the moment. It's using some muggle game called the 'Secret Number' or something like that."

Your Task

Your task to write a game that will challenge the Ravenclaw student's ability to think. This game is also quite easy. The Door (computer) will select a random number from 1 to 100. Then the student will attempt to guess it. After each guess, the Door tells the student whether their guess is too high or too low. Once they get it correct, the swings open.

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.

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 the make the input blue). For this example, the Door selected **42** as the secret number.

```
Greetings Ravenclaw student!
Before you may proceed... within 1 and 100, a number I need.

Guess: 50
Alas, that is too large.

Guess: 20
Regrettably, that is too small.

Guess: 40
Regrettably, that is too small.

Guess: 45
Alas, that is too large.

Guess: 42

Well-reasoned! You may now enter!
```

Tips

Work on each of the requirements below one at a time. You will turn in the final program, but incremental design is best for labs.

- First make sure your random number is working. You can print it to the screen for testing.
- Second, make your While Loop work. Don't worry about the If Statements.
- Finally, add the If Statements.

Creating a Random Number

For the first part, you need to generate a random number. The csc35.o object library contains a subroutine called "random". Pass the range of numbers into **rdx**. It will return a random number from 0 to n-1 into **rdx**.

For example, if you store 100 into rdx and call the function, rdx will contain 0 to 99. So, how do you make the range 1 to 100? Perhaps you can add 1.

Program Pseudocode

```
Print a greeting
Generate a random secret number between 1 and 100 (store it somewhere)
Loop while their guess ≠ secret number
    Input their guess
    If the guess > secret then print their guess is too high
    If the guess < secret then print their guess is too low
End Loop
Print a message telling them that they are correct
```

Requirements

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

Submitting Your Lab



This activity may only be submitted in Intel Format.

Using AT&T format will result in a zero. Any work from a prior semester will receive a zero.

To submit your lab, you must run Alpine by typing the following and, then, enter your username and password.

```
alpine
```

To submit your lab, send the assembly file (do not send the a.out or the object file to:

```
dcook@csus.edu
```



UNIX Commands

Editing

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

Folder Navigation

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

File Organization

Action	Command	Description
Create folder	mkdir <i>foldername</i>	Folders are called directories in UNIX.
Copy file	cp <i>oldfile</i> <i>newfile</i>	Make a copy of an existing file
Move file	mv <i>filename</i> <i>foldername</i>	Moves a file to a destination folder
Rename file	mv <i>oldname</i> <i>newname</i>	Note: same command as "move".
Delete file	rm <i>filename</i>	Remove (delete) a file. There is no undo.