

CSE 1321L: Programming and Problem Solving I Lab

Assignment 2

Module 2

What students will learn:

- Reinforcing the concept of Boolean expressions and conditionals.
- Implementing selection statements within repetition structures (loops).

Content

- Overview
- Assignment2A: Guitar Tuner
- Assignment2B: Owl World - Ticketing System

Overview:

For this assignment, you are going to practice implementing selection structures within repetition structures. Thus, reinforcing the use of selection statements but also learning how to create more complex problems by adding repetition into your programs.

Final note: *Do not cheat*

If your temptation is to look online, don't. Come see us instead and ask questions – we are here to help. Remember, you are going to have to write code in your future job interviews, so learn it now to secure a high-paying job later.

Assignment2A: Guitar Tuner

A guitar will usually have 6 strings, each of those strings will be tuned to a specific note/frequency. The note or frequency of a guitar can be tuned by loosening or tightening the string.

For this assignment, you are going to create a program that asks the user for the frequency (Hz) of a string and output to the user if it is tuned, too low, or too high.

As a reference, the program will tune the guitar in the “Standard Tuning”:

- E String (6th) is 82.41 Hz
- A String (5th) is 110.00 Hz
- D String (4th) is 146.83 Hz
- G String (3rd) is 196.00 Hz
- B String (2nd) is 246.94 Hz
- e String (1st) is 329.63 Hz

Requirements

- o Your program should have a single `while` loop which will keep looping until all the six strings of the guitar are tuned in the Standard Tuning.
 - Your solution should not use any other loop other than this one.
- o The program should start by checking the tuning of the Low E String (6th string).
- o The program should evaluate string by string, only evaluating the next string until the current one is correctly tuned.
- o For each string of the guitar, the program should prompt the user for the current frequency of said string.
 - Print if it is too low, too high, or perfect.
 - Review the sample output for the wording of the print statements.
- o If the frequency is too low or too high, the program should ask the user again to input another frequency.
- o If the frequency is perfect, move on to the next corresponding string.

Hint

- o Your program should be “aware” of the status of each string, which can be either in tune or not tuned. Tracking the status of each string will help you figure out what string to evaluate next.

Sample Output #1:

Let's tune your guitar!

What is the frequency of the sixth string?: **80**
Too Low! Tighten the string.

What is the frequency of the sixth string?: **83**
Too High! Loosen the string.

What is the frequency of the sixth string?: **82.41**

Perfect! You are in tune, let's move on to the next string...

What is the frequency of the fifth string?: **110**

Perfect! You are in tune, let's move on to the next string...

What is the frequency of the fourth string?: **146.83**

Perfect! You are in tune, let's move on to the next string...

What is the frequency of the third string?: **196**

Perfect! You are in tune, let's move on to the next string...

What is the frequency of the second string?: **246**

Too Low! Tighten the string.

What is the frequency of the second string?: **246.90**

Too Low! Tighten the string.

What is the frequency of the second string?: **246.94**

Perfect! You are in tune, let's move on to the next string...

What is the frequency of the first string?: **329.63**

Perfect! You are in tune, let's move on to the next string...

Your guitar is tuned and ready to use!

Assignment 2B: Owl World - Ticketing System

“Owl World” a KSU-themed theme park, is about to open. But they are missing their ticketing system which you are in charge in developing.

Owl World charges \$28.00 to adults and \$24.00 to children for a 1-day entry ticket. While for customers with an account they get a \$6.00 discount per ticket, charging \$22.00 to adults and \$18.00 to children.

With your program, the user should be able to buy tickets but also be able to register and sign in to be able to get the discount.

Requirements

- o Your program should have a single `while` loop which will keep looping until the user chooses to exit the program.
 - Your solution should not use any other loop other than this one.
- o The program's main menu options will be different based on whether the user has sign-in into the system or not.
 - If the user has not signed in, the program should feature the options:
 - 1. *Buy Tickets*
 - 2. *My Account*
 - 3. *Exit*
 - If the user is signed in, the program should feature these options:
 - 1. *Buy Tickets*
 - 2. *Sign Out*
 - 3. *Exit*
 - Moreover, if the user is signed in, the program should add a welcome message to the user placed above of the menu selections.
- o For this menu, use `IF` statements.
- o If the user chooses option 1, the program should display the corresponding ticket prices (discounted or not) and prompt the user how many adults and children tickets they would like to purchase. Lastly, print a receipt.
- o If the user chooses option 2 without having signed in, display the user another menu:
 - 1. *Sign-in*
 - 2. *Register*
 - If the user chooses option 1, ask for the username and password and authenticate.
 - If the user chooses option 2, ask for the username and password and save it.
 - For this menu, use `MATCH` statements.
- o If the user chooses option 2 while being signed in, sign out the user.

Regarding Sign-in and Registration

- o We have not covered sequences and data structures yet, therefore the Sign-in and Registration functionality for this assignment will be simplified by only keeping track of one user at a time.
- o To be able to track the user's username and password you are going to need to initialize a username and password variable **before** the main program loop starts. Initialize them as **empty** strings.

- o When the user “registers”, you are going to overwrite the username and password with the user input.
- o When the user “sign in”, you are going to verify and authenticate using the variables mentioned above.

Hint

- o Like Assignment2A, the program will have to keep track whether the user has signed in or not.

Sample Output #1

```
[Owl World]
Choose an option:
1. Buy Tickets
2. My Account
3. Exit
> 1

Ticket Prices:
Adult: $28.00, Children: $24.00

How many Adult tickets?: 2
How many Children tickets?: 3

Processing...

2x Adult Tickets ---- $56.00
3x Children Tickets ---- $72.00
Total: $128.00

Choose an option:
1. Buy Tickets
2. My Account
3. Exit
> 3

Program Terminated
```

Sample Output #2

```
[Owl World]
Choose an option:
1. Buy Tickets
2. My Account
3. Exit
> 1

Ticket Prices:
Adult: $28.00, Children: $24.00

How many Adult tickets?: 2
```

How many Children tickets?: **3**

Processing...

2x Adult Tickets ---- \$56.00
3x Children Tickets ---- \$72.00
Total: \$128.00

Choose an option:

1. Buy Tickets
2. My Account
3. Exit

> **2**

Choose an option:

1. Sign-in
2. Register

> **2**

Username: **jd0e01**

Password: **strongPassword**

Registration Successful!

Choose an option:

1. Buy Tickets
2. My Account
3. Exit

> **2**

Choose an option:

1. Sign-in
2. Register

> **1**

Username: **Jd0e01**

Password: **strongPassword**

Incorrect Username/Password

Choose an option:

1. Buy Tickets
2. My Account
3. Exit

> **2**

Choose an option:

1. Sign-in
2. Register

> **1**

Username: **jd0e01**

Password: **password**

Incorrect Username/Password

Choose an option:

1. Buy Tickets
2. My Account
3. Exit

> 2

Choose an option:

1. Sign-in
2. Register

> 1

Username: **jd0e01**

Password: **strongPassword**

Login Successful!

Welcome jd0e01!

Choose an option:

1. Buy Tickets
2. Sign Out
3. Exit

> 1

Ticket Prices:

Adult: \$22.00, Children: \$18.00

How many Adult tickets?: 2

How many Children tickets?: 3

Processing...

2x Adult Tickets ---- \$44.00

3x Children Tickets ---- \$54.00

Total: \$98.00

Welcome jd0e01!

Choose an option:

1. Buy Tickets
2. Sign Out
3. Exit

> 3

Program Terminated

Submission Instructions:

- o Programs must follow the output format provided. This includes each blank line, colons (:), and other symbols.
- o Programs must be working correctly.
- o Programs must be written in Python.
- o Programs must be submitted with the correct .py format.
- o Programs must be saved in files with the correct file name:

- Assignment2A.py
- Assignment2B.py
- o Programs (source code files) must be uploaded to Gradescope by the due date.