

# CSE 1321L: Programming and Problem Solving I Lab

## Assignment 3 – 100 points Solving Problems

What students will learn:

- 1) Problem solving
- 2) Write code that includes if/else statements
- 3) Design programs that leverage match/case logic
- 4) Using logic within loops
- 5) Nested loops (loops inside of loops)
- 6) Solve problems of increasing complexity

Follow the same conventions for class names and file names for your source code. For the Java folks, remove the “package” statement if you have one. Make sure to follow the [FYE Submission Guidelines](#). Finally, we don’t mean to lecture, but we want to remind you [not to cheat](#). This is the core of what a lot of you will be doing for a living, so master it now.

### **Assignment 3A: *Mathletes Game!***

With the introduction of loops, we now know enough about programming to begin creating simple games. Our first game (sponsored by the KSU College of Science and Mathematics) will test your ability to solve multiplication problems.

Your game will first ask the player if they want to play on easy or hard mode. Save their response in an appropriate variable (you will reference it later). Then, you will randomly generate two numbers and display them to the player in the form of a multiplication problem ([Refer to the online lab videos for help with random numbers](#)). The possible numbers will range between -255 and 255. You will ask the player to solve the equation. You will keep creating multiplication questions until the player wins or loses.

Based on whether the player chose “hard” or “easy” mode, the game will play a little differently. Refer to the following rules for the difference:

<b>Easy Mode</b>	<b>Hard Mode</b>
Player only has to answer <b>3</b> questions	Player has to answer <b>6</b> questions
Player can try again if they get a question wrong (up to two times)	Player loses instantly if they get a question wrong

Call the file name Assignment3A.py and the class name Assignment3A. Example outputs are shown below. Note that due to using a random number generator, you will not get the same numerical results – but the formatting and wording should be the same.

User input is indicated in **bold**.

Sample Output #1:

[Mathletes Game]

Choose a game mode (0=Easy, 1=Hard): **0**

Playing on easy mode, huh? Okay!

Q1.  $3 * 4 = ?$

**12**

Correct!

Q2.  $-3 * 7 = ?$

**0**

Incorrect! Try again.

**-21**

Correct!

Q3.  $-1 * -2 = ?$

**3**

Correct!

You win!

Sample Output #2:

[Mathletes Game]

Choose a game mode (0=Easy, 1=Hard): **1**

So, you want a challenge? Okay!

Q1.  $-12 * 7 = ?$

**-84**

Correct!

Q2.  $-3 * 77 = ?$

**539**

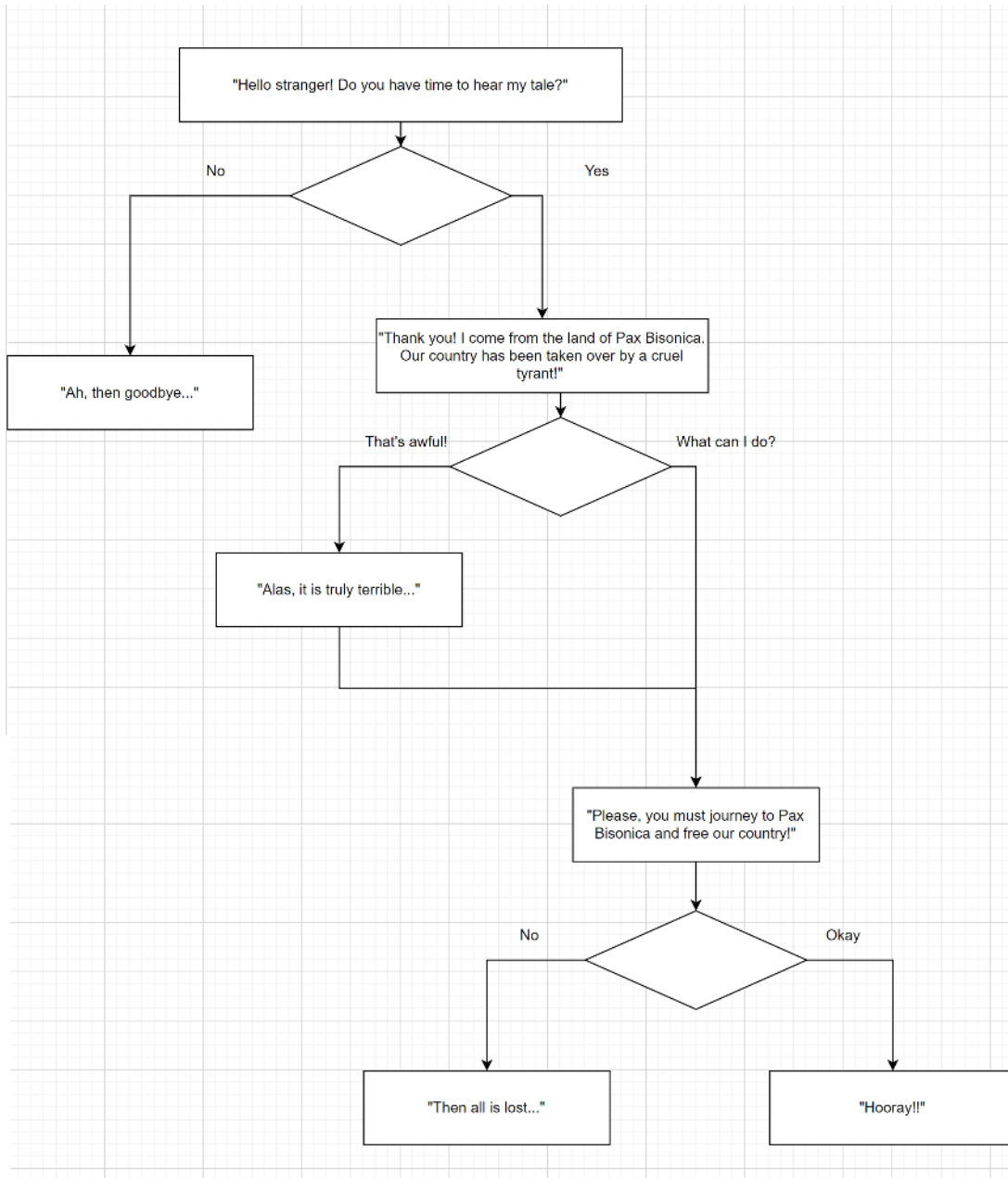
Incorrect!

Game over...

### Assignment3B:

*Decision Trees:* Many video games and interactive stories use “decision trees” to allow for different dialogue based on the player’s choices. For this assignment, you will create a (basic) dialogue system using **NESTED IF** statements. The diagram on the next page shows the different choices you can make, and the appropriate responses.

**Note:** The following diagram is a simplified form of UML – Unified Modeling Language. It is one way to display an algorithm so that people can understand it and implement it in code. For our purposes, the boxes show the printed prompts and responses. The diamonds indicate decisions after each prompt. For example, if I respond “No” to the first prompt, my program should print the response shown at the end of the “No” line on the diagram.



You must call the class **Assignment3B** and put it in a file called "Assignment3B.py",

Make sure to use **nested IF statements** to complete this assignment. Your program should produce the following output based on different user input.

Sample Output #1:

```
[Epic Quest Simulator]
Hello stranger! Do you have time to hear my tale?
1) Yes
2) No
2
Ah, then goodbye...
```

Sample Output #2:

```
[Epic Quest Simulator]
Hello stranger! Do you have time to hear my tale?
1) Yes
2) No
1
Thank you! I come from the land of Pax Bisonica. Our country has been
taken over by a cruel tyrant!
1) That's awful!
2) What can I do?
1
Alas, it is truly terrible...
Please, you must journey to Pax Bisonica and free our country!
1) Okay
2) No
1
Hooray!
```

Sample Output #3:

```
[Epic Quest Simulator]
Hello stranger! Do you have time to hear my tale?
1) Yes
2) No
1
Thank you! I come from the land of Pax Bisonica. Our country has been
taken over by a cruel tyrant!
1) That's awful!
2) What can I do?
2
Please, you must journey to Pax Bisonica and free our country!
1) Okay
2) No
2
Then all is lost...
```



Sample Output #2:

```
Enter the size of the square: 4
Enter the size of the border: 3
Enter the color of the square: 2
Invalid input!
Enter the color of the square: 1
Enter the color of the border: 0
PBM File Contents:
P1
10 10
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 1 0 0 0
0 0 0 1 1 1 1 0 0 0
0 0 0 1 1 1 1 0 0 0
0 0 0 1 1 1 1 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
```

### **Submission:**

1. You will submit 3 separate files containing source code – one for each of the assignments above. Make sure the files are named correctly and to include the header with your name and course section listed.
2. Upload all 3 files (simultaneously) to the assignment submission folder in Gradescope.
3. We'll continue to work with you on this assignment if something messes up, so long as you submit by the due date. However, start early because we work during the weekday.

## **APPENDIX**

To actually see your image, copy and paste the contents of the PBM file (starting with the P1 line) into a text editor like Notepad or TextEdit. Save it with a .pbm extension (making sure to save as "All Files" rather than as a Text file).

If you did this correctly, the file should have a blank icon (or at least, not look like a text file). You can then view the image by dragging it onto this page:

<http://paulcuth.me.uk/netpbm-viewer/>

If you'd like to edit it further and make your own PBM files, you can use image software like the free [GNU Image Manipulation Program](#).