ASSIGNMENT 3: Almost a Riddle
DEPARTMENT AND COURSE NUMBER: COMP 1010
COURSE TITLE: Introduction to Computer Science 1
TERM: Summer 2023

Assignment 3: Almost a Riddle

DUE DATE: JULY 3RD , 2021 11:55 PM

Notes:

- Name your sketches using your name, the assignment number, and the question number, *exactly* as in this example: `LastnameFirstnameA3Q1`.
- Your programs must run upon download to receive any marks.
- Submit one PDE file for each question.
- Assignments must follow the programming standards document published on the course website.
- After the due date and time assignments may be submitted but will lose 2% of marks per hour late or portion thereof.
- You may submit a question multiple times, but only the most recent version will be marked.
- These assignments are your chance to learn the material for the exams. *Code your assignments independently*. We use software to compare all submitted assignments to each other, and pursue academic dishonestly vigorously.

Q1: Riddle

Your Riddle assignment:

In Question 1, you will implement a very simple and basic Riddle program. The program starts by showing a series of words (set in a constant at the top of the program) on the screen. Then the user is prompted to enter an answer in a dialog box. Finally, the program determines whether the entry matches the hidden answer or not.

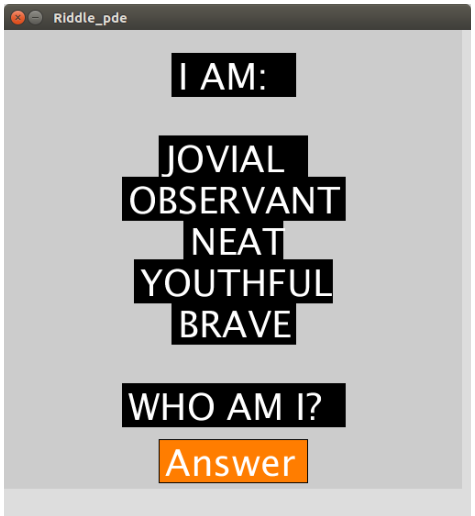

Illustration 1: Figure 1 shows an example of a riddle using the above rules. The answer in this case would be "JONYB"

Here are the rules for the riddle:

- There needs to be a first line to introduce the riddle, such as "I am:", "The thief had:", or something else.
- There needs to be an end line to close the Riddle such as "Who am I?", "Who is the thief?", "Where is the thief?", etc...
- There should multiple words compromising the riddle (**minimum 4**, preferably more).
- These words should be declared in a **single String constant** at the top of the program, each word should be separated by a comma and a space (", ").
- To make it easy for the user to read the words, make all the words printed on the screen are upper case, **even if they are not upper case in the constants**.
- The words should be centered on the X axis, no matter their length, with a box behind them as highlighting; you may choose the colours yourself. You will need to use a loop.
- You need to store a hidden answer based on the words, the answer will be a String made consisting of all the first letters of the words.

- You need an Answer button at the bottom of the screen, clicking it should make a dialog box appear (use JOptionPane) where the user may enter a word to try to solve the riddle.
- If the answer is correct, the screen should go black.
- You will need to use String functions for this assignment such as String.equals(), String.indexOf(), String.length(), String.substring(), and String.toUpperCase(). Read up on these and see how they could be of use to you.

Figure 1 shows an example of a riddle using the above rules. The answer in this case is "JONYB"

As the program starts:

- First draw the beginning of the Riddle on the canvas "I am:" or something else.
- Separate the Riddle into the individual words using a loop.
- As you separate each word:
  - You should add its first letter to your "secret" String.
  - You should then draw the word, centered on the canvas.
- Draw the "Answer" Button
- Think about whether you need to redo all of this every frame.
- When the user clicks on the Answer button, they should be prompted to enter the text.
- Finally, you need to compare the two Strings, the user input and the "secret" String you created earlier.
- If the answer is correct, the screen should go black.

Write your program using several small functions that do specific portions of the job. You may choose your own way to split up the program into functions.

Use DemoA3Q1.mp4 as reference.

Assignment 3 Marking Guide

Q1: Riddle (17)

- The program draws the Riddle. (1)

- The program draws the Answer box. (1)
- The words are separated from the Riddle string using a loop (2)
- Each word of the Riddle is centered on the Canvas (2)
- Each successive word is drawn on a new line and properly spaced out (2)
- Each word is contained in it’s own box (2)
- The Secret string is formed from the first letter of each word(2)
- Clicking the Answer box opens a JOptionPane allowing the user to answer the Riddle (1)
- The program compares the two Strings and and responds as required (2)
- The program still works if the Riddle String is changed (2)

Check Code (10)

- Constants, variables, and functions are used and named correctly (2)
- Proper commenting (2)
- Functions are defined as suggested, in both questions (2)
- Code is properly organized and easy to follow (2)
- All other programming standards are followed (2)

Grade: /27