TAYLOR'S SCHOOL OF COMPUTING AND IT

Bachelor of Computer Science (Hons)
Bachelor of Software Engineering (Hons) &
Bachelor of Computer Science (Hons) - Computer Security & Forensics
Bachelor of Information Technology (Hons) – Internet Technologies

ITS66704
Advanced Programming Sem 0821
Assignment Task 1 – 20 marks (20%)
Online (6 hours)

Objectives / Module Learning Outcomes (MLO 2)

The objective of this special assessment is to enable the students to:
2. Apply problem solving skills to evaluate and solve specific topics in advanced object-oriented problem and programs.

INSTRUCTIONS

1. **Complete all of the following tasks (Google "PacMan" for game knowledge).**
2. **Write your name, student id, module code as comment on top of your main program (starting java file).**
3. **Zip your java files, screenshot, and sound files and name it as XXXXXX_AP_Task1.zip where XXXXXX is your student id.**
4. **Submit your answer through Times BEFORE the cut-off time at 1pm.**

Based on the PacMinah game template (Normal Java Swing Application) given to you, modify the game to adhere to the following specifications:

1. Change the **player class name (Minah.java)** and its instantiated object "minah" to your short name for example "Joyah.java" and "joyah" if your name is Joyah.
2. Create a superclass called "Item" to make Minah and Monster to become its subclasses.
3. Enhanced "hunting" function. Current hunting function moves monster randomly.
4. Add restart function.
5. Add Minah and Monster collision detection. If they collided, the game is over.
6. Add two (3) more monster for a total of 4 monsters.
7. Customized Minah image to a minimum of 8-point polygon in a 30x30 pixels square.
8. Customized the welcome sound in the game. The wav file created must be generated by your own vocal cord only (speech) and sounds close to the original opening sound. Your name must be identifiable in the sound. Rename the files to your short name for example "minah.wav".
9. Change the original color/font/positioning to make it unique with better playability.
10. Basic OOP concepts of Instantiation, Encapsulation and Inheritance should be present by the time all tasks completed.

BONUS

1. Add the keyboard arrow keys to control the player's movements.
2. Customized monsters' images to minimum of 10-point polygon in a 30x30 pixels square.

SUBMISSION

Save a screen shot of your game in running mode. Upload the image together with your zipped program and the sound file to Times (Test2 Submission) before 1pm today. Your java files should include Item.java, Minah.java (change to your short name), Monster.java, GamePanel.java, PacMinah.java (change to your short name), and minah.wav (change to your short name).

--- The End ---

MARKING SCHEME

Basic Specifications (20 marks)

Items	Marks
Base game setup	2
Requirement No 1 to No 7 (1½ marks each)	10½
Sound Requirements (No 8)	2
Program Aesthetic and Uniqueness (Requirement No 9)	2
Smooth execution	1
Screen shot (Print screen)	1½
Submission conformance	1
TOTAL	**20**

Bonus (max 3 marks)

Items	Marks
Requirement 1 to 2 (1 mark each)	2
TOTAL	**2**

Note : Marks are capped at maximum of 20 even if total of basic and bonus is greater than 20.

MARKING RUBRICS

For EACH criterion of marks allocated, the following rubrics will be applied:

100% of allocated marks	75% of allocated marks	50% of allocated marks	25% allocated marks	0% allocated marks
• Complete understanding of the problem • A plan that could lead to a correct solution with no algorithmic errors • Correct solution	• Misinterprets minor part of the problem • Substantially correct solution with minor omission or procedural error	• Misinterprets major part of the problem • Partially correct solution but with major fault • Computational error, partial solution for problem.	• Completely misinterprets the problem • Substantially inappropriate solution	• No attempt • No answer or wrong answer based upon an inappropriate solution