| **Section 1 – Practice Assessment Task Overview and Description** |
| --- |

## Case Study

A beginner programming institution has requested your team to build fun and interactive games for beginner programmers to try and replicate in their own time. The team leader has tasked you with building a number guessing game and has provided you with the below requirements.

* The game must be developed in *Python 3.7* using *Visual Studio Code*
* The game will take **between 1 and 7 guesses** from the player
* The game must use the **console** as an input and output method
* A **Flowchart or Pseudocode algorithm** must be developed for the students to use as a reference
* **Internal documentation** must be present within the python code

The team leader has provided you with a mock-up of the game, and sent you a screenshot

|  |  |
| --- | --- |
| Loosing | Winning |

You have already done a brief breakdown on what you will have to develop as follows:

Generate a random number between 1 and 25. Allow the user to input a guess then display if the guess was greater or less than the random number. Take at most 7 guesses from the player **(Task 1)**. If the guess equals the random number, stop the game and display the output **(Task 2)**. All guesses need to be saved, and outputted at the end of the game **(Task 3)**

Please complete the following tasks within the **time restraint provided by your teacher**.

*It is highly recommended you read every task before starting.*

## Tasks

*Task 0 – Develop an Algorithm*

Develop a Flowchart or Pseudocode algorithm outlining the flow of this program, including all loops, if statements and variables used. Paste a screenshot of this below.



*Task 1 – Taking Guesses*

Take guesses from the user and display if its greater / less than the random number. **Ensure to document your code.**

*Task 1.1*

Allow the user to enter up to 7 guesses.

If the user has guessed the number, stop asking for more guesses.

*Task 1.2*

Show the user if their guess is less than or greater than the random number

*Task 2 – Outputting the result*

Take guesses from the user and display if its greater / less than the random number. **Ensure to document your code.**

*Task 2.1*

If the user has guessed the number, display how many guesses it took them

*Task 2.2*

If the user could not guess the number, call them a fool and show them the number

*Task 3 – Logging guesses*

Display all guesses the user made after they have finished guessing

| **Section 2 – Practice Assessment Task Submission Information** | |
| --- | --- |
| **Submission Details** | **Due date:** |
| This is a practice assessment and does not count to your final grade. Please use this assessment as a **guide only** on what will be on the actual assessment.  You may submit this assessment through canvas; however, it will not be marked. If you require feedback on your practice submission, please contact your teacher. |

| **Summary of Evidence to be Submitted** |
| --- |
| * This document |
| * Flowchart or Pseudocode Algorithm in screenshot form |
| * Python Source Code as an |
| The task will be assessed as satisfactory when all of the required evidence listed has been satisfactorily demonstrated.  \* If applicable, for graded units, the task must be satisfactorily completed before marks will be allocated. Refer to your unit outline for more information. |

| Section 3 – Practice Assessment Task Criteria and Outcome | |
| --- | --- |
| *All items/criteria must be demonstrated satisfactorily to achieve this task. The items/criteria for this activity will be assessed as S – Satisfactory or US – Unsatisfactory.* | |
| Items/criteria | |
| 1. | A functional algorithm in either Pseudocode or a Flowchart that covers all the tasks |
| 2. | A functional python script that covers all the tasks |
| 3. | Usage of at least 1 (one) while loop |
| 4. | Usage of at least 2 (two) if, elif and / or else statements |
| 5. | Usage of at least 1 (one) list or array |

|  |  |
| --- | --- |
| **Section 4 – General Practice Assessment Information** | |
| **Decision Making Rules** | Each activity in the assessment task must be satisfactorily completed for the task to be assessed as satisfactory.  Every task must be satisfactorily completed to be assessed as competent in the unit.  *\* For graded units, competence must be demonstrated before a mark can be given.* |
| **Plagiarism** | There are serious penalties for plagiarism that may include repeating a new assessment task or being withdrawn for the unit / course.  Students must ensure that all assessments are their own work (or group work and clearly noted as such).  Please refer to [www.swinburne.edu.au/corporate/registrar/plagiarism/index.html](http://www.swinburne.edu.au/corporate/registrar/plagiarism/index.html) |
| **Reasonable Adjustment** | Students may request reasonable adjustment for assessment tasks.  Reasonable adjustment usually involves varying:   * the processes for conducting the assessment (eg: allowing additional time, varying the venue) * the evidence gathering techniques (eg: oral rather than written questioning, use of a scribe, modifications to equipment)   However, the evidence collected must allow the student to demonstrate all requirements of the unit.  If you have any other issue that may impact your ability to undertake the assessment, please discuss with your teacher. |
| **Re-submission** *(where tasks are not satisfactorily completed)* | Assessment tasks that are not satisfactory can be resubmitted up until the end of the unit as scheduled on the Unit Outline. The timing on this may depend on the equipment required for this assessment task.  Resubmissions received after the scheduled unit end date may not be accepted unless approved by the teacher prior to the end date.  Note: Assessment tasks submitted for the first time after the unit end date as scheduled in the Unit Outline will not be assessed and the student should re-enrol into the unit. |
| **Special consideration** | Students may apply for Special Consideration where personal circumstances have adversely affected their task result or ability to undertake an assessment. A Special Consideration form can be completed prior to, but no later than 3 days after, the date of assessment and submitted to the relevant Manager. |
| **Work Health & Safety** | Activities may require the use of equipment or participation in group exercises. If the teacher identifies any unsafe activity or potentially dangerous situations, the teacher can stop the assessment at any time. |