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

# Student / Class Details

**Full name:** Matilda Goodison

**Student ID:** 101889190

**Teacher:** Anh Nyugen

**Date / Time started:** 13/06/2020 12:00 AM

GitHubLink: <https://github.com/Matilda578/BingoTime/tree/master>

By checking the box below, you agree that penalties exist for plagiarized work and that all work submitted is your own. Please refer to the bottom of the document for more information.

I agree and confirm that all work in this assessment is mine

Introduction  
The Swinburne Bingo club has decided to update their practices and join the digital world. They have requested a program to be made that is able to draw the numbers for a Bingo game.

They have already outsourced the creation of the bingo playing cards to some students. They only require the number drawing to be created.

However, the average age of the Swinburne Bingo club is 110 and they are uncomfortable with using anything more than a console program.

# Tasks

Write a program that draws bingo numbers. When the program runs, it will randomly choose a number between 1 and a user inputted upper limit. The program should start by requesting the upper limit from the user.

It should then go to a main menu that looks like:

A picture containing black, white

Description automatically generated

## Pass Tasks

1. Upon starting the program should ask for the upper limit of numbers to be drawn
   1. Program should not accept non-numeric input
   2. Program should not accept negative numbers
2. Upon pressing “1” a new number is drawn
   1. No duplicate numbers should be drawn
   2. No negative numbers should be drawn
3. Upon pressing “2” all drawn numbers should be printed
   1. Provide an option to print all numbers in the order that they were drawn
   2. Provide an option to print all numbers in numerical order
4. Upon pressing “3” user is prompted to enter numbers one by one to check if they have been drawn
5. Upon pressing “4” the program will exit

## Extension (optional) (as above, but include):

1. Program will not accept non-numeric input and handle
2. Program will not accept negative numbers and handle
3. Create a new menu option that allows a list of numbers to be entered. These will all be checked to see if they have been drawn.

## Extra Extension (optional):

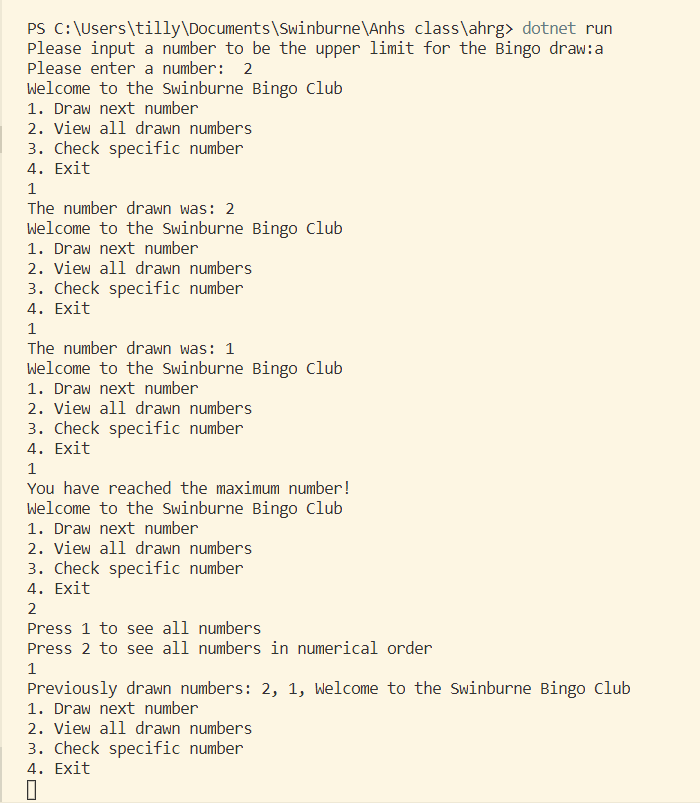
1. Create a new menu option that returns some statistics
   1. Total of numbers drawn thus far
   2. Average of numbers drawn thus far
2. Upon exiting the program will write the drawn numbers to a text file called “drawn\_numbers.txt”

Concepts to keep in mind:

* Random
* Integer
* Print
* While Loops
* If Statements
* Lists
* Functions (useful)

| **Section 2 – Assessment Task Submission Information** | |
| --- | --- |
| **Submission Details** | **Due date:** |
| 1. The assessment task must be submitted via Canvas or directly to the teacher with an assessment cover sheet. 2. Ensure to include on the front page or in the header or footer of your assessment:    * your name    * student ID    * your teacher’s name    * the unit code/s and title/s    * the assessment task title 3. The program must be observed and checked off during the assessment time 4. Submissions received after the submission date must be approved by your teacher. |

| **Summary of Evidence to be Submitted** |
| --- |
| * This document with the relevant parts filled in |
| * GitHub link of Source Code in Canvas Comments |
| * Screenshot of the program functioning |
| * A completed observation checklist |
| 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 – 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. | Loops for the UI |
| 2. | Arrays/List used for storing and calculating |
| 3. | If/Switch statements used for UI |
| 4. | Appropriate datatypes used |
| 5. | Functions used for decomposition |
|  |  |
|  |  |

|  |  |
| --- | --- |
| **Section 4 – General 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. |