# Sarawak Campus

Assignment Cover Sheet

# (for individual and group assignments)

This cover sheet is to be attached to all assignments, both hard copy and electronic format

COS10009 3

**DECLARATION**

Date Received

Unit Code Tutorial/Lab Group Assignment Title

Due date

Unit Title Lecturer/Tutor Name

**ASSIGNMENT DETAILS**

Custom program 12/11/2022

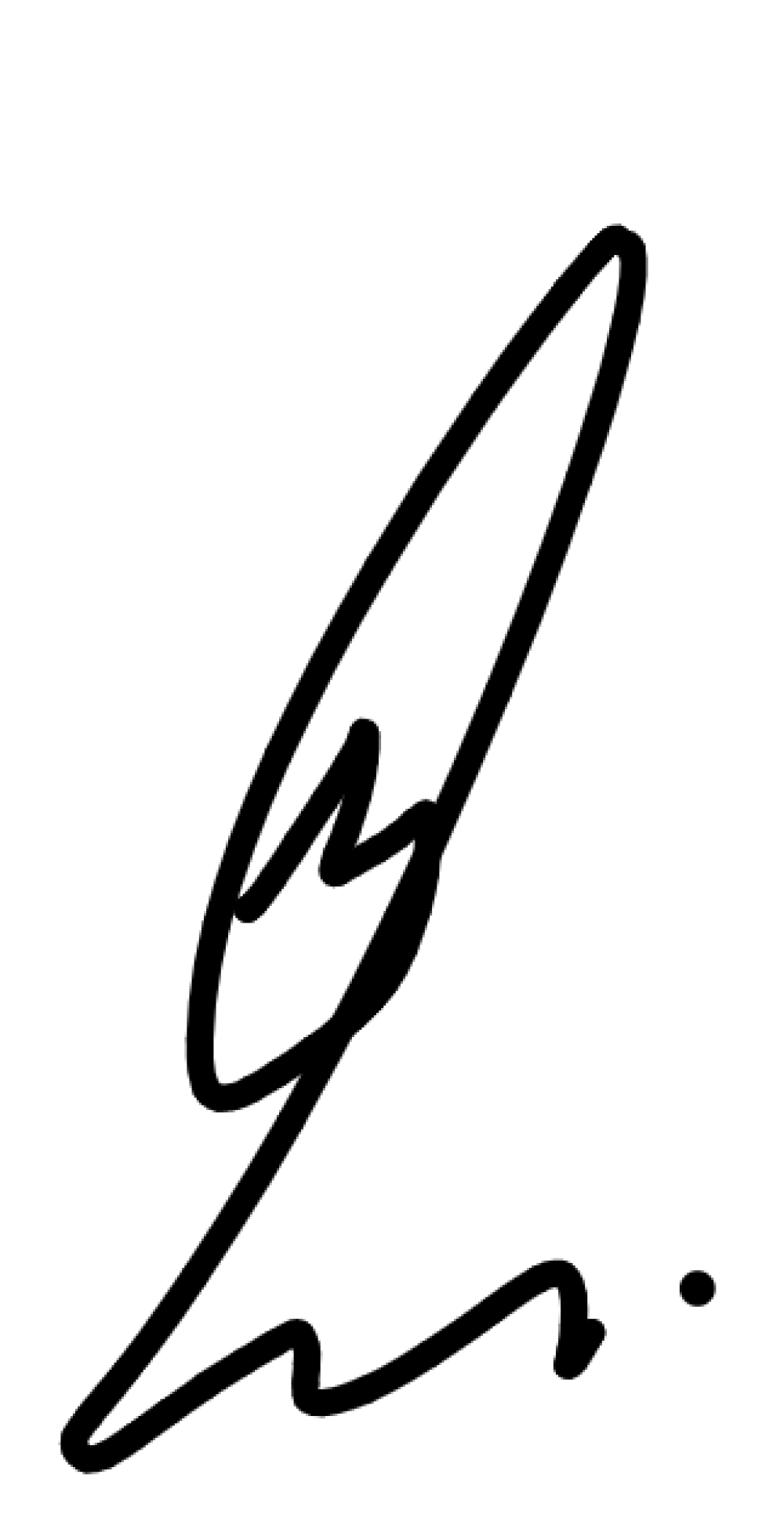
Introduction To Programming

Dr. Jason Yong

For both individual and group assignments, in the case of assignment submission on behalf of another student, it is assumed that permission has been given. The University takes no responsibility for any loss, damage, theft, or alteration of the assignment.

To be completed if this is an individual assignment

I declare that this assignment is my individual work. I have not worked collaboratively, nor have I copied from any other student’s work or from any other source/s, except where due acknowledgment is made explicitly in the text, nor has any part been written for me by another person.



101233250

Munieb Awad Elsheikhidris

Student 1

Student Signature

Student Name

Student ID Number

Student Details

To be completed if this is a group assignment

We declare that this is a group assignment and that no part of this submission has been copied from any other student's work or from any other source except where due acknowledgment is made explicitly in the text, nor has any part been written for us by another person.

|  |  |  |
| --- | --- | --- |
| Student Details Student ID Number(s) | | Student Name(s) Student Signature (s) |
| Student 1  Student 2  Student 3  Student 4  Student 5 |  | |
| **MARKER’S COMMENTS** | | |

**EXTENSION CERTIFICATE**

Date

Marker’s Signature

Total Mark

This assignment has been given an extension by

Unit Convenor

Extended due date

Date Received

Version 4, 2 August 2016. Owner: The Academic Board, Sarawak.

This cover sheet is a live document available on the Swinburne Sarawak intranet; a print copy may not be the latest version

**CUSTOM PROGRAM DESIGN REPORT**

**Overview of Custom Program**

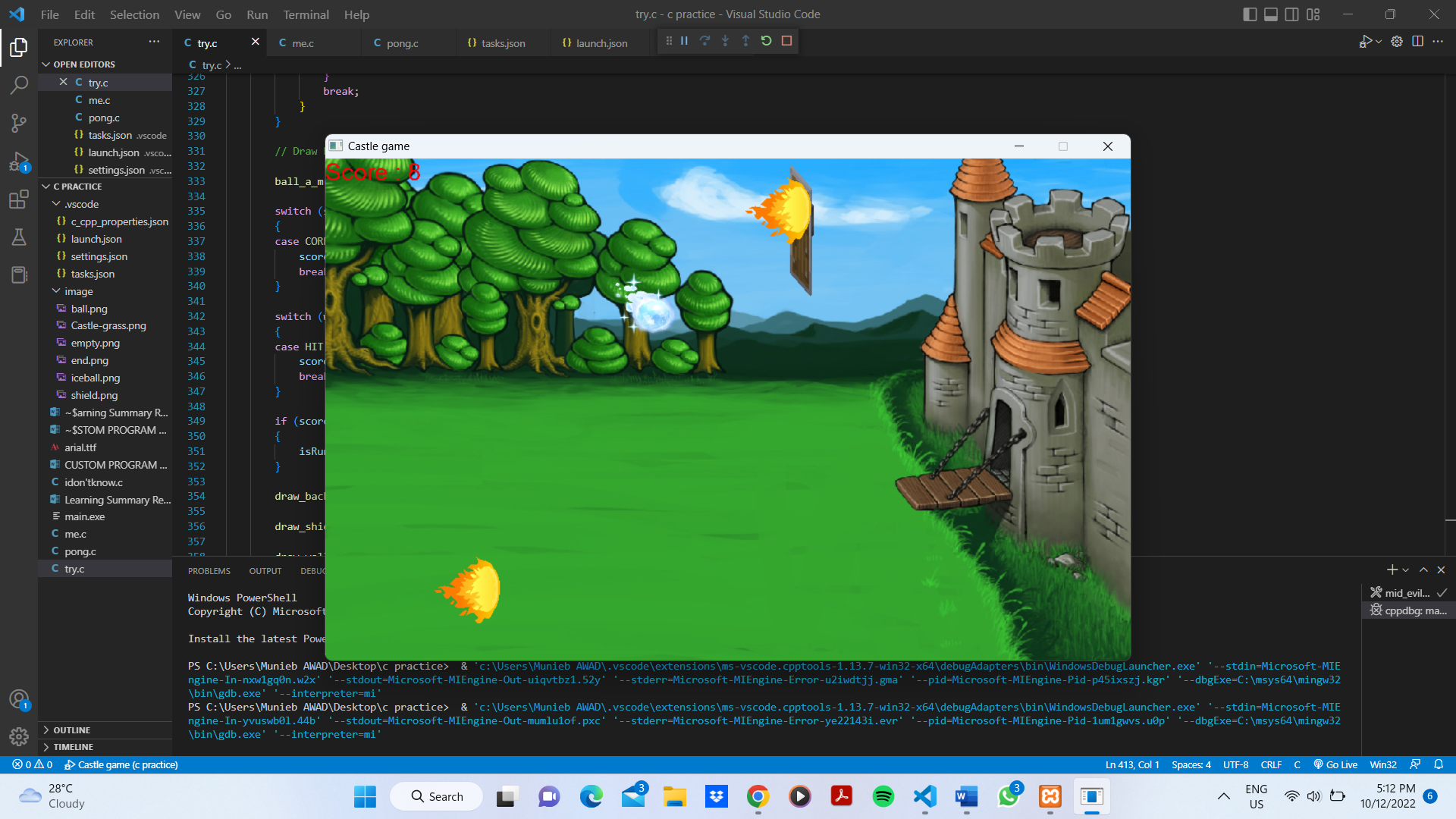
My custom program is going to a Castle game on which the player needs to defend the castle from the fire and ice ball being shut at the. The player will have a shield that will help him deflect the balls. Whenever the player deflects an ice ball he gains a point, on the other hand when a fire or an ice ball hits the castle the player will lose 5 points. When your score goes under 0 you lose the game. The game has been made using SDL user interface, the player will only have only 2 ways of input to the game up and down to control the shield movement.

**FLOWCHART**

Diagram

Description automatically generated

**GAME INTERFACE**



For this game I have made the Width 850 and the height 530. An image of grass and a castle have been rendered in the background using the function ‘draw\_background’. Then a shield image has been rendered using the function ‘draw\_shield’ which can be moved up and down using (up-arrow, w) for up and (down-arrow, s) for down. The balls have been put in an array because there are two types of images (fire and ice ball) as shown in the function ‘draw\_ball’ which is moving from the left to the right using the function ‘ball\_a\_m’. The function ‘shield\_ball’ have been adding to have collection between the ball and the shield and when the player stop a ball he game a point which is shown in the top left corner, and when the ball passes the shield the player should lose 5 points so I rendered an empty image at x = 650 from y = 0 to y = 530 and when the ball collied with the empty wall using the function ‘wall\_ball’ the score will be reduced by 5. When the score is less than 0 the isRunning will = to false which end the first loop.

**GAME END**

A picture containing text, screenshot, electronics, display

Description automatically generated

When the main loop ends the second will start that will display game over as shown in the image above which use the same function as the castle ‘draw\_background’ with the image over rendered on it.

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Description | Example of value |
| Shield | Typedef struct | Stores the shield’s index, y and x coordinates, and the texture pointer | int x;      int y;      int shield\_ball\_index;      SDL\_Texture \*texture\_shield; |
| Ball | Typedef struct | Stores the ball’s index, y and dy coordinates, and the texture pointer, and a bool variable that indicates whether the face is there or not. | bool available;      int x;      int y;      int dir\_x;      int ball\_index;      SDL\_Texture \*texture; |
| Wall | Typedef struct | Stores the shield’s index, y and x coordinates, and the texture pointer | int x;      int y;      int wall\_ball\_index;      SDL\_Texture \*texture\_wall; |
| i | Int | Used in the loop | 8 |
| y | Int | Coordinates of the ball, shield and wall | 0-530 |
| x | int | Coordinates of the ball, shield and wall | 0-850 |
| bwall | Typedef | An enumeration struct that will be used as a flag | HIT,      MESS,      NO |
| ballhit | Typedef | An enumeration struct that will be used as a flag | WRONG,      CORRECT,      NOTHING |
| ball\_array | Array | Stores an array of the ball displayed | 8 |
| score | int | Stores the player’s score | 0-infinate |
|  |  |  |  |

|  |  |
| --- | --- |
| Function /Procedure/Method | Description |
| draw\_shield | Draw the shield |
| draw\_wall | Draw an empty wall |
| draw\_ball | Draw the balls |
| draw\_background | Draw the 2 backgrounds |
| ball\_a\_m | Adds ball the program and moves them from 0-650 |
| shield\_ball | The collision function between the ball and the shield |
| wall\_ball | The collision function between the ball and the wall |
| draw\_score | Draws the score |
| main | Will execute the game it has all the rendering tasks and the keyboard events. |