BTEC level 3 Extended Diploma in Information Technology

Harry Caldwell

GD3 – Creating, Testing, Documenting

Contents

[Test table 3](#_Toc106835976)

[Debugging 4](#_Toc106835977)

[User documentation 4](#_Toc106835978)

[Technical documentation 5](#_Toc106835979)

[Variables that are used in the program 5](#_Toc106835980)

[Colour variables 5](#_Toc106835981)

[screen variables 5](#_Toc106835982)

[Snake and speed variable 5](#_Toc106835983)

[positional variables 5](#_Toc106835984)

[feedback 6](#_Toc106835985)

[Improvements that can be made using the feedback 6](#_Toc106835986)

# Test table

|  |  |
| --- | --- |
| **Tests** | **outcome** |
| When is the game launched does it launch in a separate window? | Game launched in a separate window. |
| Does the snake draw correctly | Snake spawned correctly when the game launched. |
| Does the snake move up when up arrow key is pressed | Snake successfully moves up on key press |
| Does the snake move left when the left arrow key is pressed | Snake successfully moves left on key press |
| Does the snake move right when the right arrow key is pressed | Snake successfully moves right on key press |
| Does the snake move down when the down arrow key is pressed | Snake successfully moves down on key press |
| Does the snake grow after eating the fruit | Snake successfully grows when the fruit is eaten |
| Does the score counter increase when the fruit is eaten by the snake | The score counter successfully increases in intervals of one when the fruit is eaten as intended |
| Does the fruit spawn correctly | The fruit spawns correctly when the game is opened and after the snake has eaten the fruit |
| Does the score display correctly when then game is opened and during the duration of the game | The game displays the score counter throughout the game |
| When the snake hits the screen does the game kill the snake and land on the game over screen | The game correctly kills the snake and the game moves onto the game over screen |
| Does the snake die when it runs into itself | The game kills the snake and lands on the game over screen |
| Does the game over screen display when the snake dies | The game over screen successfully displays when the snake dies |
| Do the options on the game over screen work correctly | The game over options works correctly when the correct key is pressed |

**See the screen capture of the game for evidence of the game running and the testing of the game**

# Debugging

Debugging in python was easy to do and very similar to the way of debugging using the C language, when I was debugging when issues were encountered print statements were very important as this allowed me to see what code was being executed and in terms of the movement, I was able to make it show me what it was detecting when key presses occurred.

In addition to this programming in modules allowed me to take some of the program and test if it would work making the debugging process for the overall program much easier.

Furthermore, in addition to the methods, I just spoke about, the use of the IDE screen was also one of the keyways I found issues with the program as it would tell me where the program is having errors, especially if they were simple ones such as missing brackets, quotation marks and incorrect variables that were written into the game by accident.

In addition to this the use of breakpoints were also essential to the debugging of the game because it allowed me to stop the program after a specific line to test larger parts of the program and test whether the code all fits together and works correctly.

Furthermore, the use of the comments that I wrote in the code as I was programming it allowed me to see what each module of the code does, so that I could come back to it with fresh eyes if frustration begins, the comments also allow me to let other people look at the game and provide ideas on what could be going wrong in the program.

# User documentation

The game that has been developed is snake, the objective of the game is to eat the fruit that is spawned on the screen in order to grow your snake and gain score, however if the snake hits itself or runs into the boundary of the window the game is displayed in the snake will die and you will be prevented with a game over screen.

The controls for the game:

* Up arrow key to make the snake move up
* Left arrow key to make the snake move left
* Right arrow key to make the snake move right
* Down arrow key to make the snake move down

The snake will automatically move in the direction that has been pressed by the user meaning the no control of the movement is needed aside from the direction that the snake is going to move in, the score counter will be located in the top left hand of the window and will update every time the snake has eaten a fruit.

When the game over screen appears when the user has failed, it will give you two options these are:

* Press C to play again
* Press Q to quit the game

When the C key has been pressed, the game will reset and the counter will reset back to zero, and the game will allow you to play the game again.

When the Q key has been pressed the game will exit and close the game.

# Technical documentation

## Variables that are used in the program

### Colour variables

I set colour variables at the beginning of the program, and these variables are for the different colours that are in the snake game, this ranges from the background colour to the text colour that is displayed, these variables were set using RGB code. The variables are as follows:

* white = (255, 255, 255)
* yellow = (255, 255, 102)
* black = (0, 0, 0)
* red = (213, 50, 80)
* green = (0, 255, 0)
* blue = (50, 153, 213)

### screen variables

To draw the screen, I had to create variables that gave values to the hight and the width of the display, I did this as a design decision, this is because I believed that having a full screen for snake might unintentionally make the game harder, so I decided on changing this using the variables:

* dis\_width = 600
* dis\_height = 400

Then to create the window, I used the code “dis = pygame.display.set\_mode((dis\_width, dis\_height))” which is used to create the window by using the two variables above. This is also used for the food placement functions which

### Snake and speed variable

To set the speed of the snake and the block in which the snake spawns I used the variables:

* snake\_block = 10
* snake\_speed = 15

### positional variables

for the positions of things such as the snake or the fruit, I used the variables:

* x1 = dis\_width / 2
* y1 = dis\_height / 2
* x1\_change = 0
* y1\_change = 0

the y1\_change and the x1\_change are used for the movement of the snake.

# feedback

The tester I got to play the game was my brother, the feedback that was received from him was that the window for the game was a good size and allowed for both clear view of the game and so the game was well received.

However he also said that the game speed used for the snake seemed like it was too fast which made him miss a lot of the time and said that for most players this could be an issue, in addition to this comments were also made about the size of the snake sprite that was used in the game, adding that because the sprite was so compared to the size of the window, it was hard to play the game well and also made it seem less accessible for people to be able to play.

He also noted the colour scheme was off putting and needed changing to something more easy on the eyes.

## Improvements that can be made using the feedback

In terms of improvement based on feedback, the first of them should be the speed that the sprite of the snake moves at, this will be the first step of making the game easier for players without making it extremely easy, the suggestion would be that the speed is decreased by 15 percent that way the game would not be made to easy.

The second improvement would be to increase the size of the snake and this could be by around 25 percent to make it both easier to see and easier to make contact with the fruits that will be spawned across the map.

Furthermore another improvement would be to change the colour scheme that is being used in the game to make it more easy on the eyes, using neutral colours would make the game easier on the eyes and increase play time, in addition to this changing the font that is used in the game to make it look more retro game like would be a visual improvement.