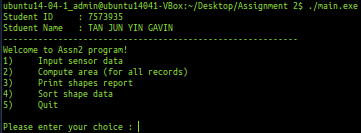
# **Requirements**

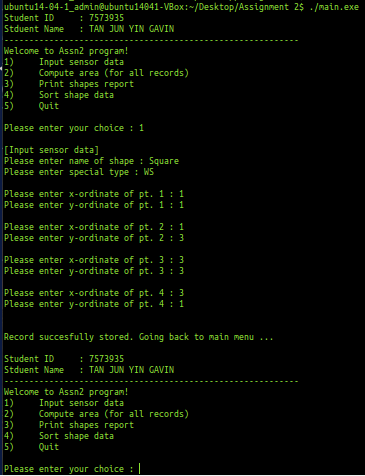
* A start-up dashboard to show the menu and some details (studentID + name)
* Individual modules/functions for each of the 4 menu options
  + To input sensor data through user input
  + Compute the area for all records that were input
  + Print out the shapes report
  + Sort the shape data
* Relevant error prompting and handling wherever needed

# **Program Design & Implementation**

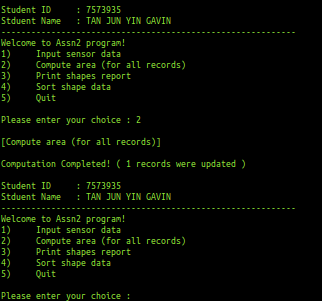
## **Program Flow**



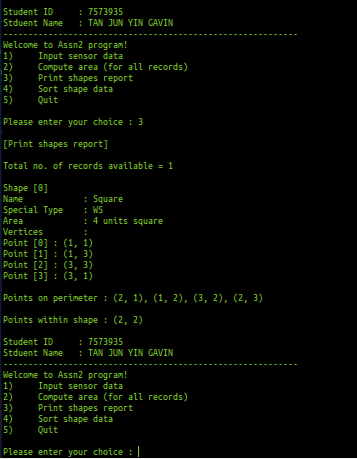
When we first run the .exe file, we will be greeted with a start up dashboard with the student ID and name followed by 5 menu options.



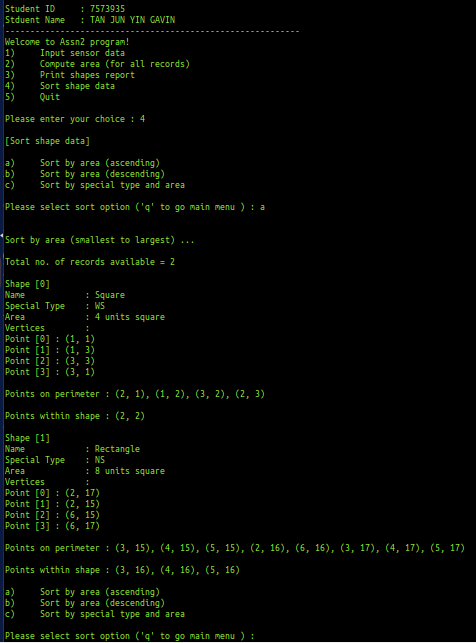
Firstly, we will read in and process the user’s input for sensor data.



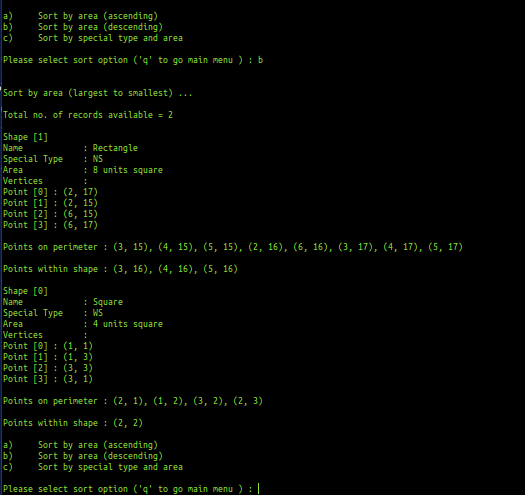
Next we will go through each of the menu options starting with “2” which is to compute the area for all available records. As you can see above, once the computation is completed, it will prompt together with the number of records that were updated.



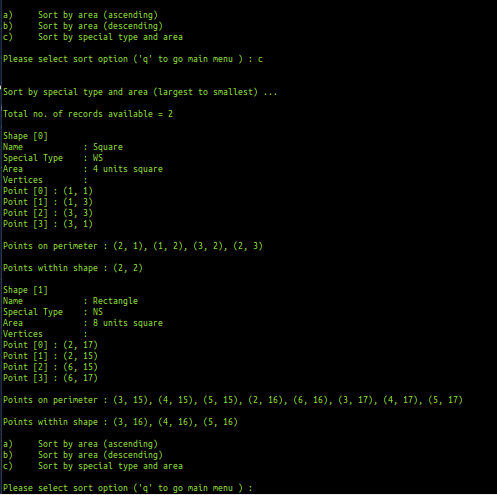
Then we will proceed to option “3” which is to display the shapes report based on the shapes and sensor data that was input. It will provide all the relevant information such as the area of the shape, type (WS or NS), its vertices, points on its perimeter as well as the points within the shape itself.



Then we will proceed to option “4” which is to display the shapes report but with the option to sort the data based on 3 different filters. They are sorted by area (ascending or descending) as well as sorted by special type and area. As you can see above, this is what you will see if you sort by ascending area.



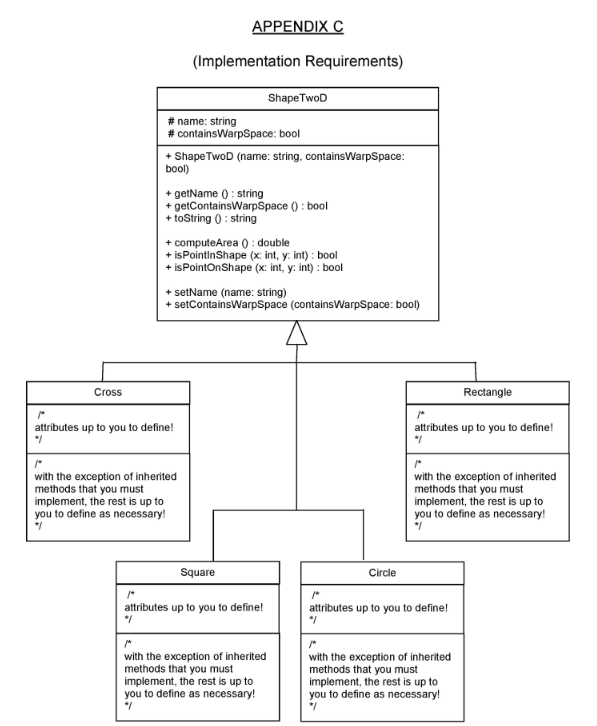
Contrastingly, if you choose sort option b, which is to sort by descending, you will see the mirror image of what was output previously.



Last but not least, we can also sort by special type and area, whereby the area is sorted from largest to smallest with WS shapes being output first followed by NS.

## **Files and Functions**

The file structure was done based on the project’s requirements as shown below:



# **Reflections**

## **Difficulties Faced**

Not being able to visualize and figure out how to work out the logic for WS and NS.

## **What Could Have Been Done Better?**

I believe one area that could still be improved would be to write cleaner code, utilizing better alternatives to help reduce space and time complexity, especially when it comes to sorting and looping. I have a few nested for loops that could have been improved using alternate solutions.

## **What Have You Learned?**

1 key skill/knowledge which I have learned through this assignment is to modulize and break down chunks of code that are being repeated, into reusable functions, thus creating helper functions.