CS23820 Assignment

1 Membership program:

This program was relatively easy to code and I had no problems with the implementation of the tree structure as described the specification. Some of this code was built off of the slides made by Dave Price called Case Study 2 as there is a good implementation of how to build a binary search tree. The basic design uses a do while loop and has functions for each option in the switch statement.

2 Race results program:

This program was a lot more interesting as there were no guidelines on how I was supposed to make the data structures. I used arrays of structs to store both the handicap values and the race results data as they felt like the easiest solution to implement and also had a much faster runtime than implementing binary search trees. As I had already previously implemented the members in the membership program I just copied the files over and used them.

The basic design like the membership program uses a do while loop which consists of 3 options print_races(), print_championship() and exit.

I'm mostly happy with the code that I have written for this assignment however I'm not very pleased with my sort_points() function (which sorts out the total points of each boat) as it creates 2 arrays inside of the function. The first array is used to temporarily store all the data of each boat in each race however this lead to some problems later on as the addition of all the points for each boat and the removal of duplicate ID's in this array proved to be much harder than I originally thought. I solved this issue however by creating a new temporary array which would store the ID's for each boat which competed and I then compared the ID's of the two arrays and if they matched then I would add the points into the correct struct in the array. Once this was completed I then performed qsort on the array so that when I printed the data out at the end all the boats would be in order by points.