/\* DIGITAL SIGNATURE(S):

====================

List the student author(s) of this code below:

Fullname Seneca Email Address

--------------------------- ----------------------------

1)Davinder Verma dverma22@myseneca.ca

+--------------------------------------------------------+

| FILE: menu\_helper.c |

+--------------------------------------------------------+

| 2 0 2 0 ~ S U M M E R |

| I P C : B T P |

| 1 4 4 : 1 0 0 |

| FINAL ASSESSMENT PART - 2 |

| |

| S E N E C A C O L L E G E |

+--------------------------------------------------------+ \*/

1. To make my program more modular and easy for understanding I created several functions, the first function which I created was menu() which involves all the printf arguments in which our program is running in a loop until the user decides to exit.

This function calls other functions like getIntInRange which makes sure that the arguments entered by the user are valid int argument within our desired range.

displayWithdrawn function checks if the skier has withdrawn or not.

displayTime function gives all the time in HH:MM format by converting all the decimals to integer and then converting to HH:MM format.

Other functions which I used is displayTime(double start, double finish, int withdrawn), this function takes double as an input and returns a string the purpose of this function is to convert the decimal value of time into back in the time format.

managerSystem() function gets all the input of the menu and calls the respective function according to the user selected menu.

displayAgeGroup(int age), this function checks if the age is in the given range it will return Junior, Adult, Senior depending upon the int argument received.

readFileRecord() function reads all information from data.txt file.

void sortInfo(), this function performs a bubble sort and sort all the skiers by the time they took the complete the race.

void dislayTopThreeSkiers(strcut Skierinfo[], int realsize), this function call getcategory and sortInfo function and gives us the top 3 skiers in a category selected.

displayInfo function calls getcategory and sortInfo and displays the Skier with their age group and time and if they withdrew or not option.

void dislayLastThreeSkiers(strcut Skierinfo[], int realsize), this function call getcategory and sortInfo function and gives us the last 3 skiers in a category selected.

void displayWinners(), this function gives us the winners in each category.

Menu is selected from the variable selection and the selected option/selection will call the function that is related to menu.

Other than that some functions are being called in multiple other functions like sortInfo so if I would have not made a function I have to put all the lines of code again and again.

1. There are 2 types of categories, the age group and race length.

For age group, we only stored racers' age and when needed, we use displayAgeGroup function in menu\_helper.h to get the text string since it is displayed in multiple reports but not used in any calculation.

For race length, we stored the category as a character and only in winners report, we need to translate the character into text-based category names.

So, I wrote an if statement in displayWinners to do that. A different way would be to store the category name directly as part of the SkierInfo struct.

This way is less efficient because it takes more power to do calculations (char comparison vs string comparison), and it takes more memory to store a string

1. readFileRecord() function read all the information of one skier until the end of the file which it determines in the end by a while loop, which checks if the file has not reached to its end and it’s not a new line character then only the loop will run and this will clear the input buffer, once we have reached to the end of file the loop will break and the function will return the result as true.

For a differing number of cases for a line record where the skiers have an additional character the function checks if there is a char after the last expected filed it will keep reading until it matched "W" and assign it as to withdraw and will continue to read. Another technique we can use is a for loop to get a single character and add inside the loop we can put a check to see if it is a new line.