Name of Project: Schedule Optimizer

Names of Libraries Used:

Course & Section Objects: <iostream>, <string>, <iomanip>, <stdlib.h>, <cstring>, <cstdlib>

FileIO: <string>, <iostream>, <cstdlib>, <fstream>, <stdio.h>, <iomanip>, “FileIO.h”

Preferences: <iostream>, <iomanip>, “Preferences.h”

main.cpp: <iostream>, <string>, <iomanip>

Description of Libraries Used:

<iostream>: “Header that defines the standard input/output stream objects”

<string>: “Strings are objects that represent sequences of characters. The standard string class provides support for such objects with an interface similar to that of a standard container of bytes, but adding features specifically designed to operate with strings of single-byte characters. The string class is an instantiation of the basic\_string class template that uses char (i.e., bytes) as its character type, with its default char\_traits and allocator types (see basic\_string for more info on the template). Note that this class handles bytes independently of the encoding used: If used to handle sequences of multi-byte or variable-length characters (such as UTF-8), all members of this class (such as length or size), as well as its iterators, will still operate in terms of bytes (not actual encoded characters).”

<iomanip>: “Header providing parametric manipulators”

<stdlib.h>: “This header defines several general purpose functions, including dynamic memory management, random number generation, communication with the environment, integer arithmetics, searching, sorting and converting.”

<cstring>: “A C-string, which consists of an array of characters terminated by the null character '\0', and which therefore is different from an ordinary array of characters. There is a whole library of functions for dealing with strings represented in this form. Its header file is <cstring>. In some implementations this library may be automatically included when you include other libraries such as the <iostream> library. Note that the null character may very well not be the very last character in the C-string array, but it will be the first character beyond the last character of the actual string data in in that array. For example if you have a C-string storing "Hello" in a character array of size 10, then the letters of the word "Hello" will be in positions with indices 0 to 4, there will be a null character at index 5, and the locations with indices 6 to 9 will contain who-knows-what. In any case, it's the null character at index 5 that makes this otherwise ordinary character array a C-string.”

<cstdlib>: “This header defines several general purpose functions, including dynamic memory management, random number generation, communication with the environment, integer arithmetics, searching, sorting and converting.”

<fstream>: “Input/output stream class to operate on files. Objects of this class maintain a filebuf object as their internal stream buffer, which performs input/output operations on the file they are associated with (if any).”

<stdio.h>: “Input and Output operations can also be performed in C++ using the C Standard Input and Output Library (cstdio, known as stdio.h in the C language). This library uses what are called streams to operate with physical devices such as keyboards, printers, terminals or with any other type of files supported by the system. Streams are an abstraction to interact with these in an uniform way; All streams have similar properties independently of the individual characteristics of the physical media they are associated with.”

“FileIO.h”: A predefined header file for the FileIO.cpp. “A header files are #included and not compiled, whereas source files are compiled and not #included. The header declares ‘what’ a class (or whatever is being implemented) will do, while the cpp file defines ‘how’ it will perform those features.”

“Preferences.h”: A predefined header file for the Preferences.cpp. “A header files are #included and not compiled, whereas source files are compiled and not #included. The header declares ‘what’ a class (or whatever is being implemented) will do, while the cpp file defines ‘how’ it will perform those features.”

Description of Project:

Our project is labeled “Schedule Optimizer” because our algorithm will make an optimal academic schedule based on user preferences and class needs. Our system is important because it would save user’s time when deciding on their semester schedules and is very applicable in the real world.

The flow and logic of the program can be summarized as follows: the user will input the required parameters about the course information and their schedule preferences. Then the method “addClass” will use these parameters to come up with all the possible combinations of courses. Once we have all the combinations, another method “bestSched” will compare these schedules with the preferences and delete the ones that doesn’t match with the user’s preferences. Then another method “displaySched” will print out the combinations that match the user’s preferences. (Note: If all preferences are not able to be met, the program will display the closet match and list out the preferences that were not met.)

Instruction to Install, Configure and Run The Program: Please download the main.cpp file and it’s associated header file(s). You can either run it on Xcode or the Terminal. If you want to run it Xcode, open up the main.cpp file and click the run button at the top left. In contrast, if you want to compile it on the terminal, the instructions differ for the type of operating system (OS). For Mac OS and Linux, follow these simple instructions:

1) Open up the terminal

2) Type in “ls” By performing this step, you will be informed about your current location in the directory

3) Type in “cd ‘name of folder’” to specify where you want to change the directory too. For example: “cd Documents” will change your current directory to the Documents folder. The command “cd” helps you change directory until you are in the folder which contains the main.cpp source code and has all of the associated header file(s) within the same directory (folder). Another specific example is as follows:

“cd Documents” // Change directory into Documents folder

“cd C++\_Codes” // Change directory from documents folder into C++\_Codes folder

“cd TeamProjectCOP3503” // change directory from C++\_Codes folder into TeamProjectCOP3503 folder

After that last command, you should be able to see main.cpp file(s) and the associated header file(s) if you saved them correctly to the same folder/location in the hard disk.

4) Compile the main.cpp file by using this command “g++ -o main.out main.cpp” in which main.out will be the name of the output file which you can run on the terminal

5) Run the “main.out” file by using this command “./main.out”. The ./“name of output file” is the command to run the output file.

Instruction to Grab The Latest Code: The current code is the latest code. However, if team members decides to continue contributing to this code to enhance it then the latest code can be accessed by emailing us at the following e-mail address: kajirashad@ufl.edu

List of Authors:

Kimberly Price (Team Leader)

Kaji Rashad

Andrew Chabab

Jamal West

Alexander Gonzalez

Alberto Molina

Instruction to Submit Bugs, Feature Request, Submit Patches, Join Mailing List, Get Announcements: If you have any questions or concerns about submitting bugs, feature request, patches, joining the mailing list or to get the latest announcements from our team then please email us at kajirashad@ufl.edu

Other Contact Information:

Kimberly Price (Team Leader): kimberlymprice@ufl.edu

Kaji Rashad: kajirashad@ufl.edu

Andrew Chabab: achabab97@ufl.edu

Jamal West: jwest23@ufl.edu

Alexander Gonzalez: alexglez.09@ufl.edu

Alberto Molina: amolina1995@ufl.edu

Instruction to Grab Documentation: Email us at the following e-mail address: kajirashad@ufl.edu. You can also contact us by phone at 772-494-3513.

© 2016 Schedule Optimizer Inc. Kimberly Price (Team Leader), Kaji Rashad, Andrew Chabab, Jamal West, Alexander Gonzalez and Alberto Molina All Rights Reserved. License and Warranty

Reference List:

http://www.cplusplus.com/reference/string/string/

http://www.cplusplus.com/reference/iostream/

http://www.cplusplus.com/reference/iomanip/

http://www.cplusplus.com/reference/cstdlib/

http://cs.stmarys.ca/~porter/csc/ref/c\_cpp\_strings.html

http://www.cplusplus.com/reference/fstream/fstream/