Name: Jingtao Cheng

PSU ID: 940067494

Date: Feb.17.2018

CS202-Win-2018-Prog3

CS202 Program 3 Design and UML

Goal:

In this project, I was aiming to create an object-oriented program to help people watch their interesting events of 2017 Winter Olympics. This application can help the user find out when a particular sport will be taking place and even when we expect a particular athlete to be competing.

Design:

In this program, I was planning to develop at least five classes to implement all the functions needed. They are **Client**, **A\_node**(node of athletes), **B\_node**(node of BST(balance tree-2 3 4 tree)), **H\_node**(node of historical records) and **Sport** class. First, the client class is used to manage all the main ability of this application and will be used in main function by user. So, the client class will contain the root node of the balance tree. The B\_node class is the basic node of the balance tree which is also a derived class from the Sport class. It has left and right pointers point to itself. It also has a + and += operator which can act as the copy constructor and insert function, the == operator can be used to compare. The sport class has four private string character which stores name, detail, time, channel. It also contains the head of the athletes list. Besides, the class contains the copy constructor which can call the base class’s constructor by the initialization list. And the go\_left, go\_right, set\_left, set\_right function which can get the left and right address and set the passed in reference to left or right address. The A\_node class stores the information about each athletes. It has name, ID, the history records list which is the head pointer point to a H\_node list. It also has go\_next, set\_next function and copy constructor to handle the dynamic memory. The class has the [] operator which can be used to search the athletes and +, += operator which can be used to add In a new athlete or a new history record. The H\_node stores each historical records, which including the time and the event and the medal’s information. It also has a copy constructor to handle the dynamic memory.

Standard requirements:

Basic requirement that I will obey in this program: I will not use statically allocated arrays in my classes or structures, and I will use dynamically allocated arrays instead. I will put all my data member in the private section and use public and private functions to access them. I will not put the input operation in my class, instead, I will put all the input operation in the client program, and I will use several .cpp file to implement the functions and use .h fine to define the functions, finally, using main.cpp to perform the all the client operation. I will not use global variables in my program, if necessary I will define const constant. I will not use the string neither string class, and I realize that I can use the cstring library to use several functions about string, like strlen(), strcmp(), and strcpy(). Every characters and sentences will be stored in the arrays of characters instead.

Requirements:

This program should implement the operators including: =, +, +=, ==, != and []. Those operator may not need to be implement in the same class. The = operator needs to be implemented in all classes that manage dynamic memory. Never make the residual value return a void operator. If the operand isn’t changed, we should make it const. The data structure should be Tree of sports. For each node we should have a LLL of athletes that compete in that sporting event. The athlete’s history of metals should also be supported by a dynamic data structure.

