Name: Jingtao Cheng

Course: CS202

Username: cheng

ID: 940067494

Program3 review&debugger

The mainly purpose for program three is to build a OO application to help manage the training. In this program, has 6 classes. The class include AVL_Node class, AVL_Tree class, Plan class, Plan_Node class, Record class, Record_Node class. All the class combine together to make a AVL tree and linear linked list.

The plan class is used to store the information like plan name, exercise type, and the exercise training frequency, and how many training days left. In this class, I design two copy constructor including passing arguments and pass in temporary object. So, every time I want to pass in the information I can create a new temporary object and pass it in to make sure I will not change any information details. The display function can display all the information store in the class. Compare function is used to compare the member and return Boolean to check is the pass in arguments match the member in class.

The plan node class is used to build a linear linked list of plan list. I design two copy constructor including passing arguments and pass in temporary object. So, every time I want to pass in the information I can create a new temporary object and pass it in to make sure I will not change any information details. The display function will invoke the base class's display function. There has to_go function used to get the next pointer of the node. Set_next function

is used to set the next pointer which is very important for connect a linear linked list. I also implement two operator overloading class. The operator + is used to connect the node to the list. And the != operator is used to match the information in the class, and it will return true or false.

The record class is used to store the information like historical record, and record year. I design two copy constructor including passing arguments and pass in temporary object. So, every time I want to pass in the information I can create a new temporary object and pass it in to make sure I will not change any information details. The display function can display all the information store in the class. Compare function is used to compare the member and return true or false to check is the pass in arguments match the member in class.

The record node class is used to build a linear linked list of record list. I design two copy constructor including passing arguments and pass in temporary object. So, every time I want to pass in the information I can create a new temporary object and pass it in to make sure I will not change any information details. The display function will invoke the base class's display function. There has to_go function used to get the next pointer of the node. Set_next function is used to set the next pointer which is very important for connect a linear linked list. I also implement operator overloading class. The != operator is used to match the information in the class, and it will return true or false.

The AVL node class is used to form the node of the AVL tree. It has head pointer of plan list and record list. I design 3 display function for display all node in tree and all node in plan list and all node in record list. I also design different insert function for those list and node. The

go_left and go_right function is used to return left and right pointer. The set_left and set_right function is used to connect each node in the tree.

AVL tree class is the interface of the program. It has many operations for manage the race and plan and record. Also have display function and etc.

Debugger:

For this program I can change a lot. The gdb help me a lot. I make the tree in my program to be a AVL tree. But when I try to add three different node into the tree. The program segfault. Firstly , I thought that It was my rotation function have some problem. But after using gdb, and use command p to get the value of the left pointer and right pointer's value. I found that the return value of my function to compare the string is reversed. So I change it and solve the problem.

I have many memory leak at first. In three places has double free the pointer and make it wrong. The memory leak is always in the destructor and read in the information repitly. When I read in a new char array , for the dynamic memory, I should to delete the old memory and set it to NULL. At first , I forgot to set each pointer to NULL when I delete it. For the next time I think I can make that set all the pointer to NULL when I delete it. For the double ,memory leak ,is that I am not good at using dynamic memory.

So I will try to familiar will the gdb commands. Each time use it I need to search for it. That waste me a lot of time.