

CS23021 Computer Science I

Project #6

Due by 11:59pm on Monday, November 20, 2006

The assignment contains two parts: (1) implementation of date class containing dynamically allocated members, (2) implementation of set template. For the first part you can reuse the code for Project #5. If your project #5 does not work, contact Mr. Kevin Schaffer. He will provide you with a working version of the project.

Implement dynamic date class and use it for timeline. In this assignment you are to modify the class developed in Project 5 such that the storage for class' member variables is allocated dynamically. The new class is called `Date2`. The class definition is provided in file `date2.h` supplied with the project assignment. Note that the member variable part contains only pointers to variables containing month, day and year of a date. The variables themselves have to be allocated dynamically.

You are to implement all the member functions listed in the class definition. This includes copy constructor, destructor and overloaded assignment. You can inline short functions. You are free to add other member functions to the class if necessary. Place class function definitions in `date2.cpp`

I put file `testdate2.cpp` in Project 6 subdirectory of the subversion repository. This program asks the user to input a date, inputs the date into an object of class `Date2`, then copies the object and prints the contents of the copy. This program continues to do so until the user inputs 0 0 0 Your code should work with this program.

The work of the program for this part of the project is similar to the second part of project #5. It is provided here for your reference.

Write a program that uses your class definition and does the following. Asks the user for today's date. Then asks the user to input a list of dates. The program loops accepting the user input. The user signals the end of input by entering date 0 0 0. You can assume that the user cannot input more than 20 dates. Your program should print the word "Past" and then the past dates (the dates that are earlier than the user input today's date) in sorted order (earliest first). Then, it should print the future dates. Do not output the 0/0/0 date. Note that the user may not input either future or past dates at all.

You should place your program in file `sortdates2.cpp`

Set template. In this part of the project you are to implement a class template for a set of elements. You have to base your implementation on an array. You can assume that the set cannot contain more than 20 elements. The array does not have to be dynamically allocated. Your template's name should be `myset`. Note, that by definition every element of a set is unique. For example, set of integers can hold these numbers: {1, 5, 2, 9, 22}. However, a set cannot hold the following numbers: { 1, 5, 1} because 1 is repeated.

You have to at least implement a member function `add()` that adds an element to the set and overloaded insertion operator that prints the contents of the set. `add()` should silently discard duplicate input. The elements of the set can be printed in the order they were entered. Place your template in `myset.h`. The templated member functions should also be put in `myset.h` Your template has to work with a test program provided in `testset.cpp` as a part of the project assignment.

The files needed for submission are: `date2.cpp`, `date2.h`, `sortdates2.cpp`, and `myset.h`