Gebze Technical University
Department of Computer Engineering
CSE 241/505
Object Oriented Programming
Fall 2022
Homework # 4
Due date Dec 18th 2022

DayofYearSet is a C++ class to represent a set of **DayOfYear** objects (in Turkish DayOfYear kümesi). This class has the following functions and features as well as any other functions that you think are necessary (setters, getters, etc.)

- **DayofYear** is a public inner class with all its public and private members.
  - **DayOfYear** inner class implements the correct date checks properly (Jan takes 31 days, Feb takes 29 days etc).
- **DayofYearSet** class keeps its elements using a **DayofYear** * data member.
- Has a list initializer constructor that takes lists such as **{d1, d2, d3}**, where **d'**s are **DayOfYear** objects.
- Overloaded stream insertion operator **operator<<** will print **DayofYearSet** details.
- Overloaded **operator==** and **operator!=** operators for comparing **DayofYearSet** objects. Two sets are equal if their elements are equal regardless of the keeping order.
- Member function **remove** removes an element from the set.
- Member function **size** returns the number of elements.
- Overloaded binary **operator+** adds an element to the set. Please note that no duplicates are allowed in a set.
- Overloaded binary **operator+** returns the union set.
- Overloaded binary **operator-** returns the difference set.
- Overloaded binary **operator-** removes an element from the set.
- Overloaded binary **operator^** returns the intersection set.
- Overloaded unary **operator!** returns the complement set. Note that there may be 365 days in a year.
- Overloaded binary **operator[]** returns the element at given position.
- Uses keywords **decltype** and **auto** in its implementation.

Your class uses a namespace and separates the class interface from the implementation.

Your driver code will be in a separate file. It will do the following

- Test each function at least 2 times and printing the results. Do not forget to test the constructors.
- Send the class objects to functions using call by value and call by reference and testing the results.
- Writes some sets to text files. Do not forget to include your saved files.
- Submit all your source files and a MAKEFILE that compiles and runs your project.

Notes:

- Do not use any functions from the standard C library (like **printf**)
- Do not use anything that we did not learn in the lectures.
- Check the validity of the user input.
- You should submit your work to the Teams page.
- Hint: Linux utility **valgrind** can test for memory leaks and other heap errors.