

BLG252E - Object Oriented Programming Homework - 1 Report

Compile Commands

```
g++ -o ./test -std=c++11 Time.cpp Citizen.cpp CivilRegistry.cpp main.cpp  
./test
```

Time.h and Time.cpp

- Donstructor** takes integer values and assign them directly.
- Whole **attributes** are private.
- getTime()** function returns const char because there is no need to change it. The function merges strings with append function.
- **< and > operators** works on year>month>day>hour>minute concept. And it checks if values are same, if yes it checks next value, if no it returns if less or bigger than or not.
- **= operator** checks all time attributes are the same or not.
- getDate()** function is used to overload Citizen < operator. It returns integer as(yymmdd)
- Destructor** deletes the object.

CivilRegistry.h and CivilRegistry.cpp

- Constructor** does not do anything.
- Whole **attributes** are private
- insertCitizen()** function push citizen to the proper list by checking hasApp with get_hasApp function to select list.
- removeCitizen()**function pop citizen from the list by taking integer to select list (wApp = 1 wOutApp= others.)
- getter()** function returns lists by selection mode above.
- Destructor** pops whole elements of both lists. It does not delete the object because it is static used on main.cpp

Citizen.h and Citizen.cpp

- Constructor** allocates memory for char* attributes and copies them from parameters. Since time is allocated memory in main.cpp constructor only assign the appTime pointer.
- Custom Copy Constructor is written for Citizen class because it has pointers in it. Copy Constructor allocates memories and copy contents into new attributes.

- getAppTime() function returns Time pointer of appTime.
- get_hasApp() is a getter for hasApp(used for < operator of Citizen);
- get_name() function returns names(used for printing).
- < operator is overloaded to make sort easier. It works different for withApp citizens and withoutApp citizens. For withApp it check date(from getDate of Time header) and then app slot(first check length and then alphabetical order) then hour(since all values equal it checks time from getTime function of Time header.). For withoutApp it only checks if time is less or not.
- Destructor deallocate attributes memory. It does not delete object because it is also used static way on main function.

main.cpp

It opens file(input.txt) and reads it.
It avoids first line of file with getline() function.
The file is read and values are assigned directly.
Time object is created dynamically.
Citizen object is created statically and it uses Time object created below.
"lists" CivilRegister object is created statically.
Whole citizens are insert to the proper lists.

Lists are get with getter function to lists created on main.cpp
Sort() function of list sorts them with overloaded < operator.

Whole list elements printed by popping.