Міністерство освіти і науки України

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»



Звіт

З лабораторної роботи № 6, варіант 8

З дисципліни « ПРОГРАМУВАННЯ, ЧАСТИНА 2 (ОБ’ЄКТНООРІЄНТОВАНЕ ПРОГРАМУВАННЯ) »

Виконав: ст. гр. КІ-15

Гербей О.М.

Перевірив: викладач

Козак Н.Б

Львів – 2020

Завдання: Створити абстрактний базовий клас і похійдний від нього клас, які реалізують модель предметної області згідно варіанту. Кожен клас має мати мінімум 3 власні елементи даних один з яких створюється динамічно, методи встановлення і читання характеристик елементів-даних класу (Set і Get), та мінімум 2 абстрактні методи обробки даних і мінімум 2 методи обробки даних у похідному класі. Крім цього клас має містити перевантаження оператора присвоєння, конструкторів по замовчуванню і копіювання та віртуальний деструктор. Для розроблених класів реалізувати програму-драйвер, яка демонструє роботу класів.

//main.cpp

#include "Header1.h"

#include "Header2.h"

#include "Windows.h"

using namespace std;

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

//Clock MyClock;

Alarm MyAlarm;

int choice, hour, minute, day;

while (true)

{

cout << "Налаштувати Годинник – Натисніть 1" << endl;

cout << "Налаштувати Будильник - Натисніть 2" << endl;

cout << "Показати Дані - Натисніть 3" << endl;

cout << "Вихід – Натисніть Будь Яку Іншу Клавішу" << endl;

cin >> choice;

//system("cls");

/\* if(choice == 1)

{

cout<<"Введіть Годину [0-24]\n";

cin>>hour;

cout<<"Введіть Хвилину [0-59]\n";

cin>>minute;

cout<<"Введіть День [1-7]\n";

cin>>day;

MyClock.SetTime(hour,minute);

MyClock.SetDay(day);

}\*/

if (choice == 2) {

MyAlarm.Start();

cout << "Введіть Годину [0-24]\n";

cin >> hour;

cout << "Введіть Хвилину [0-59]\n";

cin >> minute;

cout << "Введіть День [1-7]\n";

cin >> day;

MyAlarm.SetTime(hour, minute);

MyAlarm.SetDay(day);

cout << "Встановити Будильник?\n";

cout << "Натисніть 1 - Щоб Відповісти Так\n";

cout << "Натисніть Будь Яку Іншу Клавішу - Щоб Відповісти Ні\n";

cin >> day;

if (day == 1) {

MyAlarm.SetWork(day);

cout << "Введіть Годину [0-24]\n";

cin >> hour;

cout << "Введіть Хвилину [0-59]\n";

cin >> minute;

cout << "Введіть День [1-7]\n";

cin >> day;

MyAlarm.SetWakeUp(hour, minute);

MyAlarm.SetWakeUpDay(day);

cout << "Дізнатись Час До Дзвінка Будильника?\n";

cout << "Натисніть 1 - Щоб Відповісти Так\n";

cout << "Натисніть Будь Яку Іншу Клавішу - Щоб Відповісти Ні\n";

cin >> day;

if (day == 1)

MyAlarm.ToWakeUp();

}

else

MyAlarm.SetWork(0);

cout << "Дізнатись Тривалість Сну?\n";

cout << "Натисніть 1 - Щоб Відповісти Так\n";

cout << "Натисніть Будь Яку Іншу Клавішу - Щоб Відповісти Ні\n";

cin >> day;

if (day == 1)

MyAlarm.SleepTime();

MyAlarm.Close();

}

if (choice == 3) {

cout << "Годинник:\n";

//MyClock.Print();

cout << "--------------------------\n";

MyAlarm.Show();

}

if (choice > 3 || choice < 1)

break;

}

return(0);

}

//header1.h

#ifndef GRANDPARENT\_H

#define GRANDPARENT\_H

#include <iostream>

#include <cstdlib>

#include <string>

#include <cstring>

using namespace std;

class Clock

{

protected:

//char\* Time = new char[100];

//char\* Day = new char[100];

string Day;

string Time;

int\* Hour= new int;

int Minute;

int DayNumber;

public:

virtual void SleepTime() = 0;

virtual void ToWakeUp() = 0;

virtual bool Start() = 0;

virtual bool Close() = 0;

Clock()

{

\*Hour = 12;

Minute = 30;

DayNumber = 1;

Day = "Понеділок";

Time = "12:30";

}

virtual ~Clock()

{

// delete[] Day;

// delete[] Time;

}

/\*Clock operator =(const Clock & other){

this->Hour=other.Hour;

return \*this;

}\*/

/\*Clock& operator= (const Clock & obj)

{

cout << endl;

return \*this;

}\*/

void SetTime(int h, int m) {

\*Hour = h;

Minute = m;

}

string GetTime() {

if (\*Hour == 24)

Hour = 0;

if (\*Hour < 10) {

Time = "0";

Time += to\_string(\*Hour);

}

else

Time = to\_string(\*Hour);

Time += ":";

if (Minute < 10)

Time += "0";

Time += to\_string(Minute);

if ((Minute >= 60) || (Minute < 0))

Time = "Error Time";

if ((\*Hour > 24) || (\*Hour < 0))

Time = "Error Time";

return Time;

}

void SetDay(int argumet) {

DayNumber = argumet;

}

string GetDay() {

if (DayNumber == 1)

Day = "Понеділок";

if (DayNumber == 2)

Day = "Вівторок";

if (DayNumber == 3)

Day = "Середа";

if (DayNumber == 4)

Day = "Четвер";

if (DayNumber == 5)

Day = "П'ятниця";

if (DayNumber == 6)

Day = "Субота";

if (DayNumber == 7)

Day = "Неділя";

if ((DayNumber > 7) || (DayNumber < 1))

Day = "Error Day";

return Day;

}

void Print() {

GetDay();

GetTime();

cout << "День = " << Day << endl;

cout << "Час = " << Time << endl;

}

};

//header2.h

#include "Header1.h"

class Alarm : public Clock

{

private:

int WakeUpHour;

int WakeUpMinute;

int WakeUpDay;

bool Work;

string UpDay;

string WakeUp;

int\* Start\_H = new int;

int\* Start\_M = new int;

int Res\_H, Res\_M, Des\_D;

public:

Alarm() :Clock() {

Work = true;

WakeUpHour = 7;

WakeUpMinute = 30;

WakeUpDay = 1;

}

Alarm& operator =(Alarm& other) {

this->Work = other.Work;

this->WakeUpHour = other.WakeUpHour;

this->WakeUpMinute = other.WakeUpMinute;

this->WakeUpDay = other.WakeUpDay;

return \*this;

}

bool Start() {

cout << "Будильник Готовий До Налаштування!\n";

return true;

}

bool Close() {

cout << "Будильник Успішно Завершено\n";

return true;

}

void SleepTime() {

SetStart();

if (WakeUpHour > \*Start\_H) {

Res\_H = WakeUpHour - \*Start\_H;

if (WakeUpMinute > \*Start\_M)

Res\_M = WakeUpMinute - \*Start\_M;

if (WakeUpMinute < \*Start\_M) {

Res\_H--;

Res\_M = 60 - \*Start\_M + WakeUpMinute;

}

}

if (WakeUpHour < \*Start\_H) {

Res\_H = 24 - \*Start\_H + WakeUpHour;

if (WakeUpMinute > \*Start\_M)

Res\_M = WakeUpMinute - \*Start\_M;

if (WakeUpMinute < \*Start\_M) {

Res\_H--;

Res\_M = 60 - \*Start\_M + WakeUpMinute;

}

}

if (WakeUpHour == \*Start\_H) {

Res\_H = 0;

Res\_M = WakeUpMinute - \*Start\_M;

}

cout << "Сон Триватиме = " << Res\_H << "год. " << Res\_M << "хв.\n";

}

void ToWakeUp() {

if (WakeUpHour > \*Hour) {

Res\_H = WakeUpHour - \*Hour;

if (WakeUpMinute > Minute)

Res\_M = WakeUpMinute - Minute;

if (WakeUpMinute < Minute) {

Res\_H--;

Res\_M = 60 - Minute + WakeUpMinute;

}

}

if (WakeUpHour < \*Hour) {

Res\_H = 24 - \*Hour + WakeUpHour;

if (WakeUpMinute > Minute)

Res\_M = WakeUpMinute - Minute;

if (WakeUpMinute < Minute) {

Res\_H--;

Res\_M = 60 - Minute + WakeUpMinute;

}

}

if (WakeUpHour == \*Hour) {

Res\_H = 0;

Res\_M = WakeUpMinute - Minute;

}

cout << "До Дзвінка Будильника Залишилось: " << Res\_H << "год. " << Res\_M << "хв.\n";

}

void SetStart() {

cout << "Введіть Годину Початку Сну:\n";

cin >> \*Start\_H;

cout << "Введіть Хвилину Початку Сну:\n";

cin >> \*Start\_M;

}

void SetWakeUp(int h, int m) {

WakeUpHour = h;

WakeUpMinute = m;

}

string GetWakeUp() {

if (WakeUpHour == 24)

WakeUpHour = 0;

if (WakeUpHour < 10) {

WakeUp = "0";

WakeUp += to\_string(WakeUpHour);

}

else

WakeUp = to\_string(WakeUpHour);

WakeUp += ":";

if (WakeUpMinute < 10)

WakeUp += "0";

WakeUp += to\_string(WakeUpMinute);

if ((WakeUpMinute >= 60) || (WakeUpMinute < 0))

WakeUp = "Error Time";

if ((WakeUpHour > 24) || (WakeUpHour < 0))

WakeUp = "Error Time";

return WakeUp;

}

void SetWakeUpDay(int argumet) {

WakeUpDay = argumet;

}

string GetWakeUpDay() {

if (WakeUpDay == 1)

UpDay = "Понеділок";

if (WakeUpDay == 2)

UpDay = "Вівторок";

if (WakeUpDay == 3)

UpDay = "Середа";

if (WakeUpDay == 4)

UpDay = "Четвер";

if (WakeUpDay == 5)

UpDay = "П'ятниця";

if (WakeUpDay == 6)

UpDay = "Субота";

if (WakeUpDay == 7)

UpDay = "Неділя";

if ((WakeUpDay > 7) || (WakeUpDay < 1))

UpDay = "Error Day";

return UpDay;

}

void SetWork(int k) {

if (k == 1)

Work = true;

else

Work = false;

}

bool GetWork() {

return Work;

}

void Show() {

GetDay();

GetTime();

GetWakeUp();

GetWork();

GetWakeUpDay();

cout << "Будильник:\n";

cout << "Зараз:\n";

Print();

cout << "--------------------------\n";

if (Work == true) {

cout << "Будильник Встановлено\n";

cout << "Час Дзінка:\n";

cout << "День = " << UpDay << endl;

cout << "Час = " << WakeUp << endl;

}

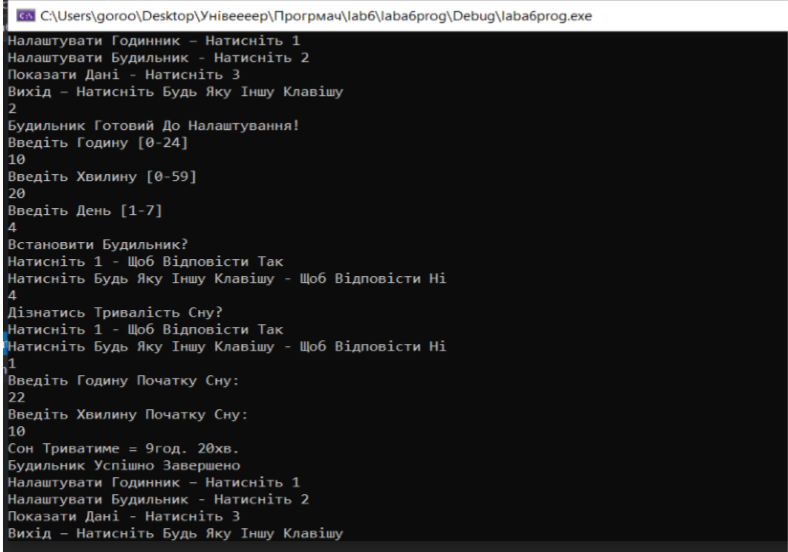
if (Work == false)

cout << "Будильник Не Встановлено\n";

}

};

Приклад виконання програми:



Висновок: впродовж цієї лабораторної роботи я ознайомився із спадкуванням класів.