Задача. Розробити клас «Таймер»

|  |  |
| --- | --- |
| Поля | * поточна кількість годин * на скільки годин встановлюється таймер |
| Методи | * метод визначення, того чи вже спрацював таймер * метод визначення того, скільки залишилось до спрацювання таймера * метод визначення того, скільки годин пройшло з моменту спрацювання таймера |

-------------- Timer.h ------------

#pragma once

#include <string>

class Timer

{

private:

//1.Опис полів

int startHour;

int hoursCount;

public:

//2.Опис методів доступу

void set\_startHour(int value);

int get\_startHour();

void set\_hoursCount(int value);

int get\_hoursCount();

//3.Опис конструкторів

Timer(int startHour, int hoursCount);

//4. Опис інших методів

bool isTimerWorked(int currentHour);

int hoursLess(int currentHour);

int hoursPassed(int currentHour);

std::string toString();#include "stdafx.h"

#include "Timer.h"

void Timer::set\_startHour(int value)

{

if (value < 0 || value > 23)

throw "Time is wrong";

startHour = value;

}

int Timer::get\_startHour()

{

return startHour;

}

void Timer::set\_hoursCount(int value)

{

if (value < 1)

throw "Timer time is wrong";

hoursCount = value;

}

int Timer::get\_hoursCount()

{

return hoursCount;

}

Timer::Timer(int startHour, int hoursCount)

{

set\_startHour(startHour);

set\_hoursCount(hoursCount);

}

bool Timer::isTimerWorked(int currentHour)

{

return currentHour >= startHour + hoursCount;

}

int Timer::hoursLess(int currentHour)

{

if (currentHour < startHour || currentHour >= startHour+hoursCount)

throw "Currennt hours incorrect";

return startHour+hoursCount-currentHour;

}

int Timer::hoursPassed(int currentHour)

{

if (!isTimerWorked(currentHour))

throw "Timer is working";

return currentHour - startHour + hoursCount;

}

std::string Timer::toString()

{

return "Start="+std::to\_string(startHour)+", hours count ="+

std::to\_string(hoursCount);

}

};

--------------- Timer.cpp ---------------

-------------------------- main -----------------

// ConsoleApplication4.cpp: определяет точку входа для консольного приложения.

//

#include "stdafx.h"

#include<iostream>

#include "Timer.h"

using namespace std;

int main()

{

Timer\* timer = new Timer(5,2);

if (timer->isTimerWorked(6))

{

printf("Has fired\n");

}

else

{

printf("Is still working\n");

}

system("pause");

return 0;

}