**МИНОБРНАУКИ РОССИИ**

федеральное государственное автономное образовательное учреждение

высшего образования

«Санкт-Петербургский политехнический университет Петра Великого»

(ФГАОУ ВО «СПбПУ»)

Институт среднего профессионального образования

**ОТЧЕТ**

**по лабораторной работе №1**

по профессиональному модулю ПМ.01 «Разработка программных модулей программного обеспечения для компьютерных систем»

Специальность09.02.07 «Информационные системы и программирование»

Выполнил студент 3 курса 32919/1 группы

Лейко Матвей Александрович

Преподаватель: Ильин Ю. П.

Санкт-Петербург

2023

**Cодержание**

[**Цель работы** 3](#_Toc118149827)

[**Задание** 3](#_Toc118149828)

[**Математическая формулировка методов** 3](#_Toc118149829)

[**Текст программных модулей** 7](#_Toc118149830)

[**Протокол испытаний** 11](#_Toc118149831)

[**Результаты выполнения работы программы** 13](#_Toc118149832)

**Тема: «Простейшие классы»**

**Вариант №9**

# **Цель работы:**

Изучение простейших классов, содержащих скрытые поля, конструкторы с параметрами и без параметров, методы, свойства. Рассмотреть и научиться использовать обработку исключительных ситуаций в программе.

# **Задание:**

**Вариант 9**

Составить описание класса для представления времени. Предусмотреть возможности установки времени и изменения его отдельных полей (час, минута, секунда) с проверкой допустимости вводимы х значений. В случае недопустимых значений полей выбрасываются исключения. Создать методы изменения времени на заданное количество часов, мину т и секунд. Написать программу, демонстрирующую все разработанные элементы класса.

(Заказчик согласовал с нами, что время максимум изменяется на 100 единиц)

# **Математическая формулировка методов:**

1. Конструктор по умолчанию

public Time() { }

Данный конструктор является конструктором по умолчанию и, следовательно, не принимает никаких параметров.

1. Конструктор инициализации

public Time(byte hour, byte minutes, byte seconds)

{

this.hour = hour;

this.minutes = minutes;

this.seconds = seconds;

}

Данный конструктор принимает значения часа, минуты и секунды, а также устанавливает значение полей. Параметры конструктора заданы целочисленным типом byte (переменные hour, minutes, seconds).

1. Метод установки значения времени

public void setTime(byte hour, byte minutes, byte seconds)

{

this.hour = hour;

this.minutes = minutes;

this.seconds = seconds;

}

Данный метод принимает значения часа, минуты и секунды на которые следует изменить текущее время. Параметры метода заданы целочисленным типом int (переменные hour, minutes, seconds).

1. Переопределённый метод для вывода информации об объекте.

public override string ToString()

{

return String.Format("Время {0}:{1}:{2}", hour, minutes, seconds);

}

Данный метод возвращает информацию об объекте (час, минута, секунда)

1. Метод для получения значений времени.

static void inputInfo(out byte hour, out byte minutes, out byte seconds)

{

while (true)

{

try

{

Console.Write("hour = ");

hour = byte.Parse(Console.ReadLine());

if (hour > 24 || hour < 0)

{

throw new Exception("Неправильное значение для часов");

}

Console.Write("minutes = ");

minutes = byte.Parse(Console.ReadLine());

if (minutes > 59 || minutes < 0)

{

throw new Exception("Неправильное значение для минут");

}

Console.Write("seconds = ");

seconds = byte.Parse(Console.ReadLine());

if (seconds > 59 || seconds < 0)

{

throw new Exception("Неправильное значение для секунд");

}

break;

}

catch (Exception e)

{

Console.WriteLine(e.Message);

continue;

}

}

}

Выходные параметры метода заданы целочисленным типом byte. В методе производится ввод значений времени. При вводе времени в неверном формате, будут вызваны исключения. Также генерируются исключения.

1. Метод для изменения часов на заданное количество

public void changeTimeHours(int hour)

{

int minutes = 0;

int seconds = 0;

if (hour > 100 || hour < -100)

{

throw new Exception("Неправильное значение для изменения времени");

}

int totalSeconds = this.hour \* 3600 + this.minutes \* 60 + this.seconds;

totalSeconds += (3600 \* ((hour % 24) + 24) + 60 \* minutes + seconds);

this.hour = (byte)((totalSeconds / 3600) % 24);

this.minutes = (byte)((totalSeconds / 60) % 60);

this.seconds = (byte)(totalSeconds % 60);

}

Данный метод изменяет значение часа на заданное значение единиц. Параметры метода заданы целочисленным значением sbyte. Также генерируются исключения.

1. Метод изменения минут на заданное количество

public void changeTimeMinutes(int minutes)

{

int hour = 0;

int seconds = 0;

if (minutes > 100 || minutes < -100)

{

throw new Exception("Неправильное значение для изменения времени");

}

int totalSeconds = this.hour \* 3600 + this.minutes \* 60 + this.seconds;

totalSeconds += (3600 \* ((hour % 24) + 24) + 60 \* minutes + seconds);

this.hour = (byte)((totalSeconds / 3600) % 24);

this.minutes = (byte)((totalSeconds / 60) % 60);

this.seconds = (byte)(totalSeconds % 60);

}

Данный метод изменяет значение минут на заданное значение единиц. Параметры метода заданы целочисленным значением sbyte. Также генерируются исключения.

1. Метод изменения секунд на заданное количество

public void changeTimeSeconds(int seconds)

{

int hour = 0;

int minutes = 0;

if (seconds > 100 || seconds < -100)

{

throw new Exception("Неправильное значение для изменения времени");

}

int totalSeconds = this.hour \* 3600 + this.minutes \* 60 + this.seconds;

totalSeconds += (3600 \* ((hour % 24) + 24) + 60 \* minutes + seconds);

this.hour = (byte)((totalSeconds / 3600) % 24);

this.minutes = (byte)((totalSeconds / 60) % 60);

this.seconds = (byte)(totalSeconds % 60);

}

Данный метод изменяет значение секунд на заданное значение единиц. Параметры метода заданы целочисленным значением sbyte. Также генерируются исключения.

**Текст программных модулей:**

Библиотечный модуль

using System;

using System.Collections.Generic;

using System.Linq;

using System.Security.Cryptography;

using System.Text;

using System.Threading.Tasks;

namespace Time

{

public class Time

{

private byte \_hour;

private byte \_minutes;

private byte \_seconds;

public byte hour

{

get { return \_hour; }

set

{

if (value > 24 || value < 0)

{

throw new Exception("Неправильное значение для часов");

}

\_hour = value;

}

}

public byte minutes

{

get { return \_minutes; }

set

{

if (value > 59 || value < 0)

{

throw new Exception("Неправильное значение для минут");

}

\_minutes = value;

}

}

public byte seconds

{

get { return \_seconds; }

set

{

if (value > 59 || value < 0)

{

throw new Exception("Неправильное значение для секунд");

}

\_seconds = value;

}

}

public void changeTimeHours(int hour)//метод для изменения значения часа на заданное количенство единиц

{

int minutes = 0;

int seconds = 0;

if (hour > 100 || hour < -100)

{

throw new Exception("Неправильное значение для изменения времени");

}

int totalSeconds = this.hour \* 3600 + this.minutes \* 60 + this.seconds;

totalSeconds += (3600 \* ((hour % 24) + 24) + 60 \* minutes + seconds);

this.hour = (byte)((totalSeconds / 3600) % 24);

this.minutes = (byte)((totalSeconds / 60) % 60);

this.seconds = (byte)(totalSeconds % 60);

}

public void changeTimeMinutes(int minutes)//метод для изменения значения митнут на заданное количенство единиц

{

int hour = 0;

int seconds = 0;

if (minutes > 100 || minutes < -100)

{

throw new Exception("Неправильное значение для изменения времени");

}

int totalSeconds = this.hour \* 3600 + this.minutes \* 60 + this.seconds;

totalSeconds += (3600 \* ((hour % 24) + 24) + 60 \* minutes + seconds);

this.hour = (byte)((totalSeconds / 3600) % 24);

this.minutes = (byte)((totalSeconds / 60) % 60);

this.seconds = (byte)(totalSeconds % 60);

}

public void changeTimeSeconds(int seconds)//метод для изменения значения секунд на заданное количенство единиц

{

int hour = 0;

int minutes = 0;

if (seconds > 100 || seconds < -100)

{

throw new Exception("Неправильное значение для изменения времени");

}

int totalSeconds = this.hour \* 3600 + this.minutes \* 60 + this.seconds;

totalSeconds += (3600 \* ((hour % 24) + 24) + 60 \* minutes + seconds);

this.hour = (byte)((totalSeconds / 3600) % 24);

this.minutes = (byte)((totalSeconds / 60) % 60);

this.seconds = (byte)(totalSeconds % 60);

}

public void setTime(byte hour, byte minutes, byte seconds)//метод для задания значений времени

{

this.hour = hour;

this.minutes = minutes;

this.seconds = seconds;

}

public override string ToString()//метод для преобразования обьектов в строку

{

return String.Format("Время {0}:{1}:{2}", hour, minutes, seconds);

}

public Time(byte hour, byte minutes, byte seconds)//конструктор с парметрами

{

this.hour = hour;

this.minutes = minutes;

this.seconds = seconds;

}

public Time() { }//конструктор по умолчанию

}

}

Программный модуль

// код программы

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Time;

namespace Lab1

{

internal class Program

{

static void inputInfo(out byte hour, out byte minutes, out byte seconds)//Метод для ввода пользовательских значений

{

while (true)

{

try

{

Console.Write("hour = ");

hour = byte.Parse(Console.ReadLine());

if (hour > 24 || hour < 0)

{

throw new Exception("Неправильное значение для часов");

}

Console.Write("minutes = ");

minutes = byte.Parse(Console.ReadLine());

if (minutes > 59 || minutes < 0)

{

throw new Exception("Неправильное значение для минут");

}

Console.Write("seconds = ");

seconds = byte.Parse(Console.ReadLine());

if (seconds > 59 || seconds < 0)

{

throw new Exception("Неправильное значение для секунд");

}

break;

}

catch (Exception e)

{

Console.WriteLine(e.Message);

continue;

}

}

}

static void Main(string[] args)

{

try

{

byte hour, minutes, seconds;

sbyte hour\_, minutes\_, seconds\_;

inputInfo(out hour, out minutes, out seconds);

Time.Time time\_1 = new Time.Time(hour, minutes, seconds);//тест кейс для проверки конструктора с параметрами

Console.WriteLine(time\_1.ToString());

Console.Write("hour = ");

hour = Byte.Parse(Console.ReadLine());

time\_1.hour = hour;

Console.WriteLine(time\_1.ToString());//проверка установки значения часа

Console.Write("minutes = ");

minutes = Byte.Parse(Console.ReadLine());

time\_1.minutes = minutes;

Console.WriteLine(time\_1.ToString());//проверка установки значения минуты

Console.Write("seconds = ");

seconds = Byte.Parse(Console.ReadLine());

time\_1.seconds = seconds;

Console.WriteLine(time\_1.ToString());//проверка установки значения секунды

Console.Write("hour = ");

hour\_ = SByte.Parse(Console.ReadLine());

time\_1.changeTimeHours(hour\_);

Console.WriteLine(time\_1.ToString());

Console.Write("minutes = ");

minutes\_ = SByte.Parse(Console.ReadLine());

time\_1.changeTimeMinutes(minutes\_);

Console.WriteLine(time\_1.ToString());//проверка изменения значения на заданное количество минут

Console.Write("seconds = ");

seconds\_ = SByte.Parse(Console.ReadLine());

time\_1.changeTimeSeconds(seconds\_);

Console.WriteLine(time\_1.ToString());//проверка изменения значения на заданное количество секунд

Console.Write("hour = ");

hour\_ = SByte.Parse(Console.ReadLine());

time\_1.changeTimeHours(hour\_);

Console.WriteLine(time\_1.ToString());//проверка изменения значения на заданное количество часов

Console.ReadLine();

}

catch(Exception e)

{

Console.WriteLine(e.Message);

}

Console.ReadLine();

}

}

}

# **Протокол испытаний:**

*Таблица 1.* Протокол испытаний

|  |  |  |  |
| --- | --- | --- | --- |
| **Проверяемые требования** | **Сообщения программы и вводимые значения** | **Ожидаемые результаты** | **Фактические результаты** |
| Проверка ввода корректных значений для: часов, минут и секунд. | hour=12;  minutes=45;  seconds=12; | Сообщение с выводом, введённого времени в консоли. | hour = 12  minutes = 45  seconds = 12  Время 12:45:12 |
| Проверка ввода некорректного значения для часов. | hour = 25 | Сообщение, вызванное исключением (в сутках всего 24 часа) при вводе часов, и повторный запрос на ввод времени. | hour = 25  Неправильное значение для часов  hour = |
| Проверка ввода некорректного значения для минут. | hour = 13  minute = 78 | Сообщение, вызванное исключением (в часе всего 60 минут) при вводе минут, и повторный запрос на ввод времени. | hour = 13  minutes = 78  Неправильное значение для минут  hour = |
| Проверка ввода некорректного значения для секунд. | hour = 20  minute = 45  second = 112 | Сообщение, вызванное исключением (в минуте всего 60 секунд) при вводе секунд, и повторный запрос на ввод времени. | hour = 20  minutes = 45  seconds = 112  Неправильное значение для секунд  hour = |
| Проверка ввода данных для изменения времени. | hour = 1  minute = 1  second = 1  hour = 10  minute = 10  second = 10 | Сообщение с выводом, заданного времени, а затем, изменённого времени в консоли. | hour = 1  minutes = 1  seconds = 1  Время 1:1:1  hour = 10  minutes = 10  seconds = 10  Время 11:11:11 |
| Проверка ввода отрицательных значений для изменения времени и обновления суток. | hour = 21  minute = 34  second = 51  hour = -24  minute = -11  second = -49 | Сообщение с выводом, заданного времени, а затем, изменённого времени в консоли. | hour = 21  minutes = 34  seconds = 51  Время 21:34:51  hour = -24  minutes = -11  seconds = -49  Время 21:23:2 |
| Проверка ввода данных для изменения времени и обновления суток. | hour = 23  minute = 58  second = 50  hour = 12  minute = 11  second = 56 | Сообщение с выводом, заданного времени, а затем, изменённого времени в консоли. | hour = 23  minutes = 58  seconds = 50  Время 23:58:50  hour = 12  minutes = 11  seconds = 56  Время 12:10:46 |
| Проверка ввода некорректного значения для часов при изменении времени | hour = 11  minute = 1  second = 1  hour = 101 | Сообщение, вызванное исключением (максимально время можно изменять на 100 единиц) при повторном вводе часов для изменения времени. | hour = 11  minutes = 1  seconds = 1  Время 11:1:1  hour = 101  Неправильное значение для изменения времени |
| Проверка ввода некорректного значения для минут при изменении времени | hour = 3  minute = 3  second = 5  hour = 12  minute = 105 | Сообщение, вызванное исключением (максимально время можно изменять на 100 единиц) при повторном вводе минут для изменения времени. | hour = 3  minutes = 3  seconds = 5  Время 3:3:5  hour = 12  Время 12:3:5  minutes = 105  Неправильное значение для минут |
| Проверка ввода некорректного значения для секунд при изменении времени | hour = 22  minute = 11  second = 33  hour = 2  minute = 3  second = 101 | Сообщение, вызванное исключением (максимально время можно изменять на 100 единиц) при повторном вводе секунд для изменения времени. | hour = 22  Время 22:11:33  minutes = 11  Время 22:11:33  seconds = 33  Время 22:11:33  hour = 2  Время 0:11:33  minutes = 3  Время 0:14:33  seconds = -101  Неправильное значение для изменения времени |

**Результаты выполнения работы программы:**

1. Ввод корректных данных(рис.1).

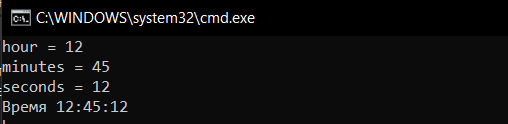


рис.1 Корректный ввод данных.

2. Ввод некорректного значения часов(рис.2)

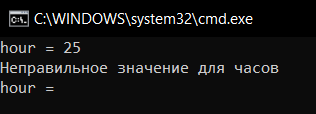


рис.2 Обработанное исключение «Неправильное значение для часов».

3. Ввод некорректного значения минут (рис.3).

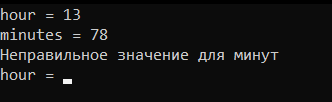


рис.3 Обработанное исключение «Неправильное значение для минут».

4. Ввод некорректного значения для секунд (рис.4).

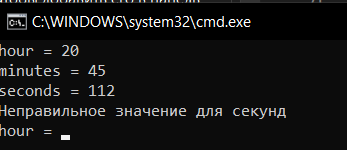


рис.4 Обработанное исключение «Неправильное значение для секунд».

5. Отдельный ввод часов, минут, секунд для изменения времени (рис.5).

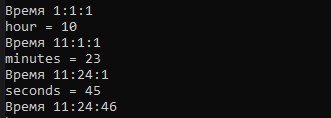


рис.5 Вывод данных с увеличением времени.

6. Ввод данных для уменьшения времени и перехода в новые сутки (рис.6).

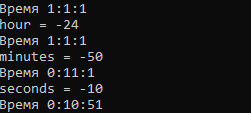


рис.6 Вывод данных об уменьшенном времени и переходе в новые сутки.

7. Ввод данных для увеличения времени и перехода в новые сутки (рис.7).

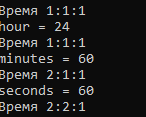


рис.7 Вывод данных об увеличении времени и переходе в новые сутки.

8. Ввод некорректного значения часов для изменения времени (рис.8).



рис.8 Обработанное исключение «Неправильное значение для изменения времени».

9. Ввод некорректного значения минут для изменения времени (рис.9).

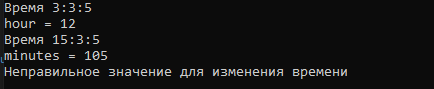


рис.9 Обработанное исключение «Неправильное значение для изменения времени».

10. Ввод некорректного значения минут для изменения времени (рис.10).

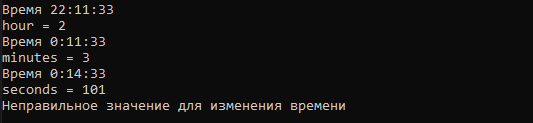


рис.10 Обработанное исключение «Неправильное значение для изменения времени».