

Міністерство освіти і науки України  
Національний технічний університет України «Київський  
політехнічний  
інститут імені Ігоря Сікорського"  
Факультет інформатики та обчислювальної техніки  
Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи №3 з дисципліни  
«Основи програмування 2. Модульне програмування»

«Перевантаження операторів»

Варіант 26

Виконав студент: ІП-15 Поліщук Валерій Олександрович  
(шифр, прізвище, ім'я, по батькові)

Перевірила: Вечерковська Анастасія Сергіївна

( прізвище, ім'я, по батькові)

Київ 2022

## Лабораторна робота № 3

### Перевантаження операторів

#### Варіант 26

**Мета** — вивчити механізми створення класів з використанням перевантажених операторів

#### Постановка задачі

26. Визначити клас "Час" для роботи із часом в межах доби. Членами класу є години, хвилини та секунди. Реалізувати для нього декілька конструкторів, геттери, метод обчислення часу, що залишився до вказаного моменту часу. Перевантажити оператори: префіксний "++" – для збільшення кількості хвилин на 1, постфіксний "++" - для збільшення кількості секунд на 1; ">" – для порівняння моментів часу. Створити три об'єкта часу (T1, T2, T3), використовуючи різні конструктори. Збільшити час T1 на 1 хвилину, а час T2 – на 1 секунду. З'ясувати, який із цих моментів часу (T1 чи T2) є більшим. Для часу T3 визначити час, що залишився до заданого.

#### Розв'язання

C#

Program.cs

```
1 namespace OP
2 {
3     partial class Program
4     {
5         static void Main(string[] args)
6         {
7             Random rand = new Random();
8
9             Time time1 = GenerateFirstObject(rand);
10            Time time2 = GenerateSecondObject(rand);
11            Time time3 = GenerateThirdObject(rand);
12
13            time1--;
14            Console.WriteLine("First object after decrementing (adding 1 min) : " + time1.ToString() + "\n");
15
16            time2++;
17            Console.WriteLine("Second object after incrementing (adding 1 sec) : " + time2.ToString() + "\n");
18
19            if (time1 > time2)
20            {
21                Console.WriteLine("First object time is greater than second ");
22            }
23            else if (time1 == time2)
24            {
25                Console.WriteLine("First object time is equal to the second ");
26            }
27            else
28            {
29                Console.WriteLine("Second object time is greater than first ");
30            }
31            Console.WriteLine();
32            Time end_time = GetEndTime();
33            Console.WriteLine();
34            Console.WriteLine("From the third object time to the specified time left: " + time3.TimeLeftFor(end_time).ToString());
35
36        }
37    }
38 }
```

## Funcs.cs

```
namespace OP
{
    partial class Program
    {
        private static Time GenerateFirstObject(Random random)
        {
            int hours;
            int minutes;
            int seconds;

            hours = random.Next(0, 24);
            minutes = random.Next(0, 60);
            seconds = random.Next(0, 60);

            Console.WriteLine("Random numbers generated for first object :\n");
            Console.WriteLine("hours : " + hours + "\n");
            Console.WriteLine("minutes : " + minutes + "\n");
            Console.WriteLine("seconds : " + seconds + "\n");

            Time time1 = new Time(hours, minutes, seconds);

            Console.WriteLine("first object : " + time1.ToString() + "\n");
            Console.WriteLine();

            return time1;
        }

        private static Time GenerateSecondObject(Random random)
        {
            int hours;
            int minutes;

            hours = random.Next(0, 24);
            minutes = random.Next(0, 60);

            Console.WriteLine("Random numbers generated for second object :\n");
            Console.WriteLine("hours : " + hours + "\n");
            Console.WriteLine("minutes : " + minutes + "\n");

            Time time2 = new Time(hours, minutes);

            Console.WriteLine("second object : " + time2.ToString() + "\n");
            Console.WriteLine();

            return time2;
        }
    }
}
```

```

49 private static Time GenerateThirdObject(Random random)
50 {
51     int hours;
52
53     hours = random.Next(0, 24);
54
55     Console.WriteLine("Random numbers generated for third object :\n");
56     Console.WriteLine("hours : " + hours + "\n");
57
58     Time time3 = new Time(hours);
59
60     Console.WriteLine("third object : " + time3.ToString() + "\n");
61     Console.WriteLine();
62
63     return time3;
64 }
65
66 private static Time GetEndTime()
67 {
68     Time end_time = new();
69     bool success = false;
70
71     do
72     {
73         Console.WriteLine("Enter a time (in format hh:mm:ss) to calculate how much time between the third object and the given time");
74         string string_time = Console.ReadLine();
75
76         if (string_time.Length != 8 || string_time[2] != ':' || string_time[5] != ':')
77         {
78             Console.WriteLine();
79             Console.ForegroundColor = ConsoleColor.Red;
80             Console.WriteLine("Invalid time format! Try again!");
81             Console.ForegroundColor = ConsoleColor.White;
82             Console.WriteLine();
83             continue;
84         }
85
86         try
87         {
88             end_time = new Time(Convert.ToInt32(string_time[0..2]), Convert.ToInt32(string_time[3..5]), Convert.ToInt32(string_time[6..8]));
89         }
90         catch (Exception)
91         {
92             Console.WriteLine();
93             Console.ForegroundColor = ConsoleColor.Red;
94             Console.WriteLine("Invalid time format! Try again!");
95             Console.ForegroundColor = ConsoleColor.White;
96             Console.WriteLine();
97             continue;
98         }
99
100         success = true;
101     }
102     while (!success); ;
103
104     return end_time;
105 }
106
107 }
108
109 }
110

```

```

99
100         success = true;
101     }
102     while (!success); ;
103
104     return end_time;
105 }
106
107 }
108
109 }
110

```

Time.cs

```

1
2 namespace OP
3 {
4     public class Time
5     {
6         public Time(int _hours, int _minutes, int _seconds)
7         {
8             if (_hours < 0 || _hours >= 24)
9             {
10                 throw new ArgumentOutOfRangeException(nameof(hours));
11             }
12
13             if (_minutes < 0 || _minutes >= 60)
14             {
15                 throw new ArgumentOutOfRangeException(nameof(minutes));
16             }
17
18             if (_seconds < 0 || _seconds >= 60)
19             {
20                 throw new ArgumentOutOfRangeException(nameof(seconds));
21             }
22
23             hours = _hours;
24             minutes = _minutes;
25             seconds = _seconds;
26         }
27
28         public Time(int _hours, int _minutes)
29         {
30             if (_hours < 0 || _hours >= 24)
31             {
32                 throw new ArgumentOutOfRangeException(nameof(hours));
33             }
34
35             if (_minutes < 0 || _minutes >= 60)
36             {
37                 throw new ArgumentOutOfRangeException(nameof(minutes));
38             }
39
40             hours = _hours;
41             minutes = _minutes;
42         }
43
44         public Time(int _hours)
45         {
46             if (_hours < 0 || _hours >= 24)
47             {
48                 throw new ArgumentOutOfRangeException(nameof(hours));
49             }
50
51             hours = _hours;
52         }
53     }

```

```

54     public Time()
55     {
56         hours = 0;
57         minutes = 0;
58         seconds = 0;
59     }
60
61
62
63     private int hours;
64     public int Hours { get; set; }
65
66     private int minutes;
67     public int Minutes { get; set; }
68
69     private int seconds;
70     public int Seconds { get; set; }
71
72     public static bool operator >(Time left, Time right)
73     {
74         if (left.hours > right.hours)
75         {
76             return true;
77         }
78
79         else if (left.hours == right.hours)
80         {
81             if (left.minutes > right.minutes)
82             {
83                 return true;
84             }
85
86             else if (left.minutes == right.minutes)
87             {
88                 if (left.seconds > right.seconds)
89                 {
90                     return true;
91                 }
92
93                 else
94                 {
95                     return false;
96                 }
97             }
98
99             else
100             {
101                 return false;
102             }
103         }
104
105         else
106         {
107             return false;
108         }
109     }

```

```
111 public static bool operator <(Time left, Time right)
112 {
113     if (left.hours < right.hours)
114     {
115         return true;
116     }
117
118     else if (left.hours == right.hours)
119     {
120         if (left.minutes < right.minutes)
121         {
122             return true;
123         }
124
125         else if (left.minutes == right.minutes)
126         {
127             if (left.seconds < right.seconds)
128             {
129                 return true;
130             }
131
132             else
133             {
134                 return false;
135             }
136         }
137
138         else
139         {
140             return false;
141         }
142     }
143
144     else
145     {
146         return false;
147     }
148 }
```

```

150 public static bool operator ==(Time left, Time right)
151 {
152     if (left.hours == right.hours && left.minutes == right.minutes && left.seconds == right.seconds)
153     {
154         return true;
155     }
156     else
157     {
158         return false;
159     }
160 }
161
162 public static bool operator !=(Time left, Time right)
163 {
164     if (!(left.hours == right.hours && left.minutes == right.minutes && left.seconds == right.seconds))
165     {
166         return true;
167     }
168     else
169     {
170         return false;
171     }
172 }
173
174 public static Time operator ++(Time time)
175 {
176     Time result = new Time(time.hours, time.minutes, time.seconds);
177     result.seconds += 1;
178     return result;
179 }
180
181 // -- instead of prefix++, because you cant override this operator in c#
182 public static Time operator --(Time time)
183 {
184     Time result = new Time(time.hours, time.minutes, time.seconds);
185     result.minutes += 1;
186     return result;
187 }
188
189 public Time TimeLeftFor(Time end)
190 {
191     int minutes;
192     int hours;
193
194     int start_seconds = this.seconds + this.minutes * 60 + this.hours * 3600;
195     int end_seconds = end.seconds + end.minutes * 60 + end.hours * 3600;
196     int result = end_seconds - start_seconds;
197
198     if (result >= 0)
199     {
200         hours = result / 3600;
201         result %= 3600;
202         minutes = result / 60;
203         result %= 60;
204
205         return new Time(hours, minutes, result);
206     }

```



```

208     else
209     {
210         result += 24 * 3600;
211
212         hours = result / 3600;
213         result %= 3600;
214         minutes = result / 60;
215         result %= 60;
216
217         return new Time(hours, minutes, result);
218     }
219 }
220
221 public override string ToString()
222 {
223     string hours = Convert.ToString(this.hours);
224     string minutes = Convert.ToString(this.minutes);
225     string seconds = Convert.ToString(this.seconds);
226     if (this.hours < 10)
227     {
228         hours = "0" + hours;
229     }
230
231     if (this.minutes < 10)
232     {
233         minutes = "0" + minutes;
234     }
235
236     if (this.seconds < 10)
237     {
238         seconds = "0" + seconds;
239     }
240
241     return hours + ":" + minutes + ":" + seconds;
242 }
243
244 }
245
246

```

## Тестування програми

C#

```
Random numbers generated for first object :  
hours : 4  
minutes : 9  
seconds : 42  
first object : 04:09:42  
  
Random numbers generated for second object :  
hours : 12  
minutes : 31  
second object : 12:31:00  
  
Random numbers generated for third object :  
hours : 0  
third object : 00:00:00  
  
First object after decrementing (adding 1 min) : 04:10:42  
Second object after incrementing (adding 1 sec) : 12:31:01  
Second object time is greater than first  
  
Enter a time (in format hh:mm:ss) to calculate how much time between the third object and the given time  
12:34:33  
  
From the third object time to the specified time left: 12:34:33
```

## Висновок

Я вивчив механізми створення класів з використанням перевантажених операторів