**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**КАФЕДРА САПР**

**ОТЧЕТ**

**по лабораторной работе №3**

**по дисциплине «Объектно-ориентированное программирование»**

**Тема: Использование базовых языковых конструкций**

|  |  |  |
| --- | --- | --- |
| Студентка гр. 9301 |  | Синицкая В. А. |
| Преподаватель |  | Новакова Н.Е. |

Санкт-Петербург

2021

Оглавление

[1 Цель работы 3](#_Toc65194769)

[2 Анализ задачи 3](#_Toc65194770)

[3 Формальная постановка задачи 3](#_Toc65194771)

[3.1 Исходные данные 3](#_Toc65194772)

[3.2 Результирующие (выходные) данные 3](#_Toc65194773)

[4 Используемые классы и методы 3](#_Toc65194774)

[5 Текст программы 5](#_Toc65194775)

[6 Контрольный пример 10](#_Toc65194776)

[7 Полученные результаты 11](#_Toc65194777)

[8 Выводы 11](#_Toc65194778)

[9 Список использованных источников 12](#_Toc65194779)

# Цель работы

Получение знаний и навыков в использовании таких конструкций, как switch, foreach, try-throw-catch и инструкции goto. Практика в обращении с конструкциями enum, if-else и преобразовании типов.

# Анализ задачи

Требуется:

1. Написать программу, включающую перечислимый тип, конструкцию switch, использующую команду break, конструкцию if-else, использующую инструкцию goto, цикл foreach.
2. Внедрить в написанную программу конструкцию try-catch, чтобы обработать неподходящие входные данные.
3. Доработать программу, дополнив ее функциональностью для високосного года.

# Формальная постановка задачи

## Исходные данные

В качестве исходных данных выступает enum, содержащий названия месяцев, массив, содержащий информацию о количестве дней в каждом из месяцев, целое число – год и целое число от 1 до 365 или 366 включительно, обозначающее день года.

## Результирующие (выходные) данные

Пользователю в консоль выводится месяц и число, обозначающее день в указанном месяце, если год високосный, что определяется по его номеру, то 60 будет 29 Февраля.

# Используемые классы и методы

В программах используются методы Console.WriteLine() и Console.ReadLine() - методы класса System.Console. System.Console – это класс для работы с консольным окном, определенный разработчиками стандартных библиотек для языка C#.

Метод Console.ReadLine() сохраняет введенную пользователем строку в заданную переменную.

Метод Console.WriteLine() служит для отображения в консольном окне пользователя строк и других данных, переданных в данный метод в качестве параметров.

В программе определено перечисление MonthName, обозначающее месяцы, две коллекции: DaysInMonths и DaysInLeapMonths, содержащие количество дней в обычном и високосном году соответственно.

Программа содержит единственный класс WhatDay, в котором используются все вышеперечисленные конструкции. Используется метод type.Parse(var\_name) для перевода считанной из консоли строки в нужный тип, используется блок try: в нем заключен код, в котором может возникнуть исключение, в этом же блоке используется throw, чтобы перейти к блоку catch, который обрабатывает данное исключение, то есть в нем описаны инструкции для такого вызова. В этом блоке используются поля Message и StackTree для переменной типа исключения ArgumentOutOfRangeException, которые служат для вывода сообщения, с которым было поймано исключение и для указания местоположения файла и строки кода. Используется оператор if-else, делящий ход выполнения программы в зависимости от того, високосный ли год, выполняя для каждого варианта цикл foreach, который проходит по элементам соответствующей коллекции, уменьшая введенное пользователем число на количество дней в месяце, которые были до текущего, пока очевидным образом не останется дата, соответствующая некоторому месяцу. В программе закомментирован код конструкций if-else, который является аналогом вышеописанной конструкции с foreach, можно раскомменитровать ее, закомментировав первую, и убедиться, что результат будет одинаковым. В тексте программы использована инструкция goto, при обнаружении которой компилятор немедленно переходит к названной строке. Названная строка объявляется в коде позже. В циклах была использована инструкция break, призывающая немедленно прервать текущую итерацию и перейти к следующей, либо завершить текущий цикл.

# Текст программы

1. using System;
2. enum MonthName
3. {
4. January,
5. February,
6. March,
7. April,
8. May,
9. June,
10. July,
11. August,
12. September,
13. October,
14. November,
15. December
16. }
18. namespace OOP3LabaWhatDay
19. {
20. class WhatDay
21. {
22. static void Main(string[] args)
23. {
24. Console.WriteLine("Please, enter the year");
25. string line = Console.ReadLine();
26. int yearNum = int.Parse(line);
27. bool isLeapYear = false;
28. int maxDayNum = 365;
29. if (yearNum % 4 == 0)
30. {
31. isLeapYear = true;
32. maxDayNum++;
33. }
34. Console.WriteLine("Please, enter the day number between 1 and {0}: ", maxDayNum);
35. line = Console.ReadLine();
36. try
37. {
38. int dayNum = int.Parse(line);
39. // dayNum = sum of all days of the months before + now data
41. if (dayNum > 365 || dayNum < 1)
42. throw new ArgumentOutOfRangeException("Day out of Range");
44. int monthNum = 0;
45. if (isLeapYear)
46. {
47. foreach (int daysInLeapMonth in DaysInLeapMonths)
48. {
49. if (dayNum <= daysInLeapMonth)
50. break;
51. else
52. {
53. dayNum -= daysInLeapMonth;
54. monthNum++;
55. }
56. }
57. }
58. else
59. {
60. foreach (int daysInMonth in DaysInMonths)
61. {
62. if (dayNum <= daysInMonth)
63. break;
64. else
65. {
66. dayNum -= daysInMonth;
67. monthNum++;
68. }
69. }
70. }
72. //if (dayNum <= 31) // January
73. //    goto End;
74. //else
75. //{
76. //    dayNum -= 31;
77. //    monthNum++;
78. //}
80. //if (dayNum <= 28 || (dayNum <= 29 && isLeapYear)) // February
81. //    goto End;
82. //else
83. //{
84. //    dayNum -= 28;
85. //if (isLeapYear)
86. //    dayNum--;
87. //    monthNum++;
88. //}
90. //if (dayNum <= 31)  // March
91. //    goto End;
92. //else
93. //{
94. //    dayNum -= 31;
95. //    monthNum++;
96. //}
98. //if (dayNum <= 30) // April
99. //    goto End;
100. //else
101. //{
102. //    dayNum -= 30;
103. //    monthNum++;
104. //}
106. //if (dayNum <= 31)  // May
107. //    goto End;
108. //else
109. //{
110. //    dayNum -= 31;
111. //    monthNum++;
112. //}
114. //if (dayNum <= 30)  // June
115. //    goto End;
116. //else
117. //{
118. //    dayNum -= 30;
119. //    monthNum++;
120. //}
122. //if (dayNum <= 31)  // July
123. //    goto End;
124. //else
125. //{
126. //    dayNum -= 31;
127. //    monthNum++;
128. //}
130. //if (dayNum <= 31)  // August
131. //    goto End;
132. //else
133. //{
134. //    dayNum -= 31;
135. //    monthNum++;
136. //}
138. //if (dayNum <= 30) // September
139. //    goto End;
140. //else
141. //{
142. //    dayNum -= 30;
143. //    monthNum++;
144. //}
146. //if (dayNum <= 31)  // October
147. //    goto End;
148. //else
149. //{
150. //    dayNum -= 31;
151. //    monthNum++;
152. //}
154. //if (dayNum <= 31) // November
155. //    goto End;
156. //else
157. //{
158. //    dayNum -= 30;
159. //    monthNum++;
160. //}
162. //if (dayNum <= 31) // December
163. //    goto End;
164. //else
165. //{
166. //    dayNum -= 31;
167. //    monthNum++;
168. //}
170. //End:
172. //MonthName temp = (MonthName)monthNum;
173. //string monthName = temp.ToString();
175. //string monthName;
176. //switch (monthNum)
177. //{
178. //    case 0:
179. //        monthName = "January"; break;
180. //    case 1:
181. //        monthName = "February"; break;
182. //    case 2:
183. //        monthName = "March"; break;
184. //    case 3:
185. //        monthName = "April"; break;
186. //    case 4:
187. //        monthName = "May"; break;
188. //    case 5:
189. //        monthName = "June"; break;
190. //    case 6:
191. //        monthName = "July"; break;
192. //    case 7:
193. //        monthName = "August"; break;
194. //    case 8:
195. //        monthName = "September"; break;
196. //    case 9:
197. //        monthName = "October"; break;
198. //    case 10:
199. //        monthName = "November"; break;
200. //    case 11:
201. //        monthName = "December"; break;
202. //    default:
203. //        monthName = "not done yet"; break;
204. //}
206. MonthName temp = (MonthName)monthNum;
207. string monthName = temp.ToString();
208. Console.WriteLine("{0} {1}", monthName, dayNum);
209. }
211. catch (ArgumentOutOfRangeException wrong\_data)
212. {
213. Console.WriteLine("Error: {0} \nStack: {1}", wrong\_data.Message, wrong\_data.StackTrace);
214. }
215. }
217. static System.Collections.ICollection DaysInMonths
218. = new int[12] { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
219. static System.Collections.ICollection DaysInLeapMonths
220. = new int[12] { 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
221. }
222. }

# Контрольный пример

Ниже приведены примеры для введенного числа = 60 с обычным годом (Рис. 6.1) и с високосным годом (Рис. 6.2).

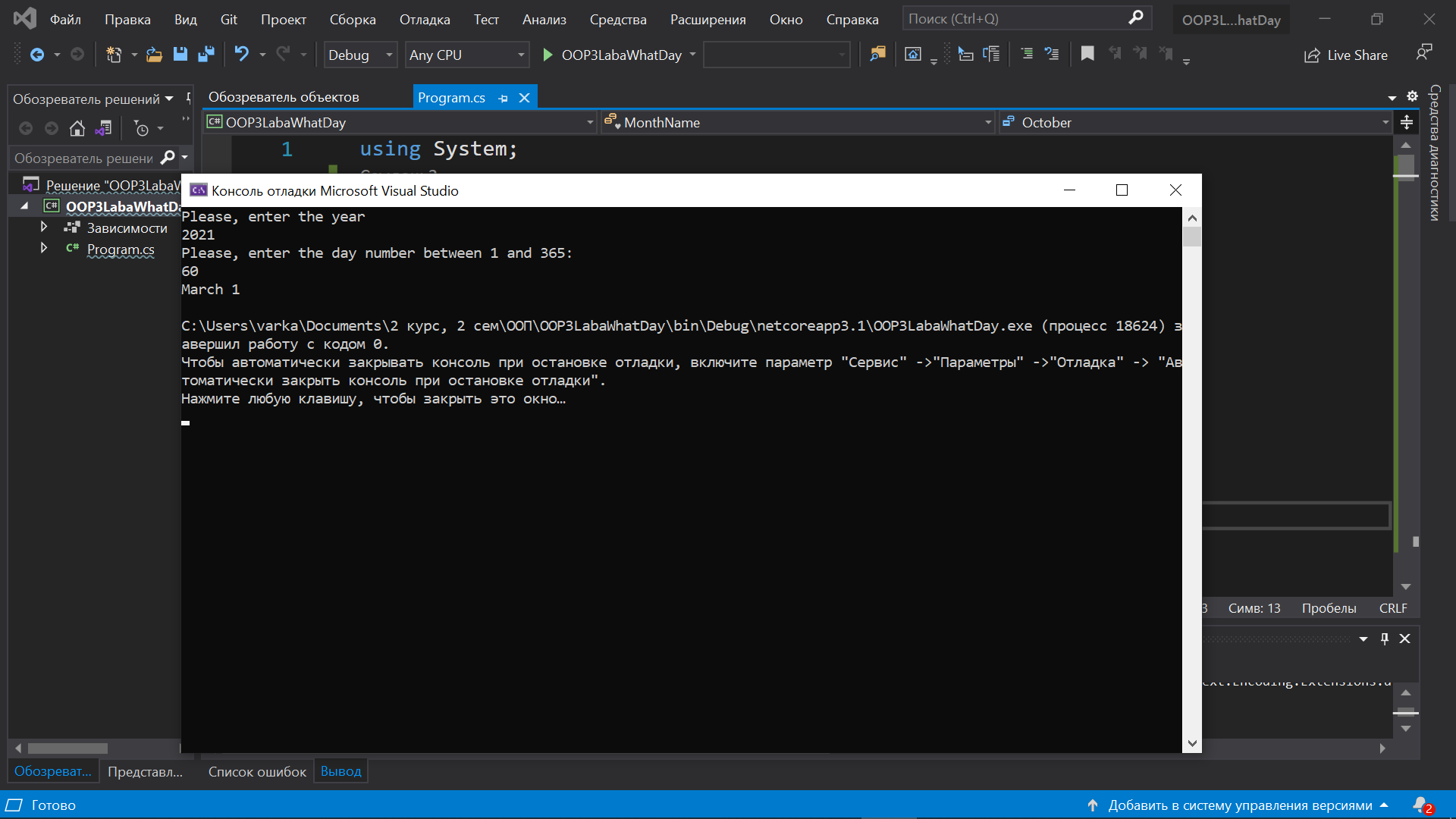


Рис. 6.1 — пример работы программы с обычным годом.

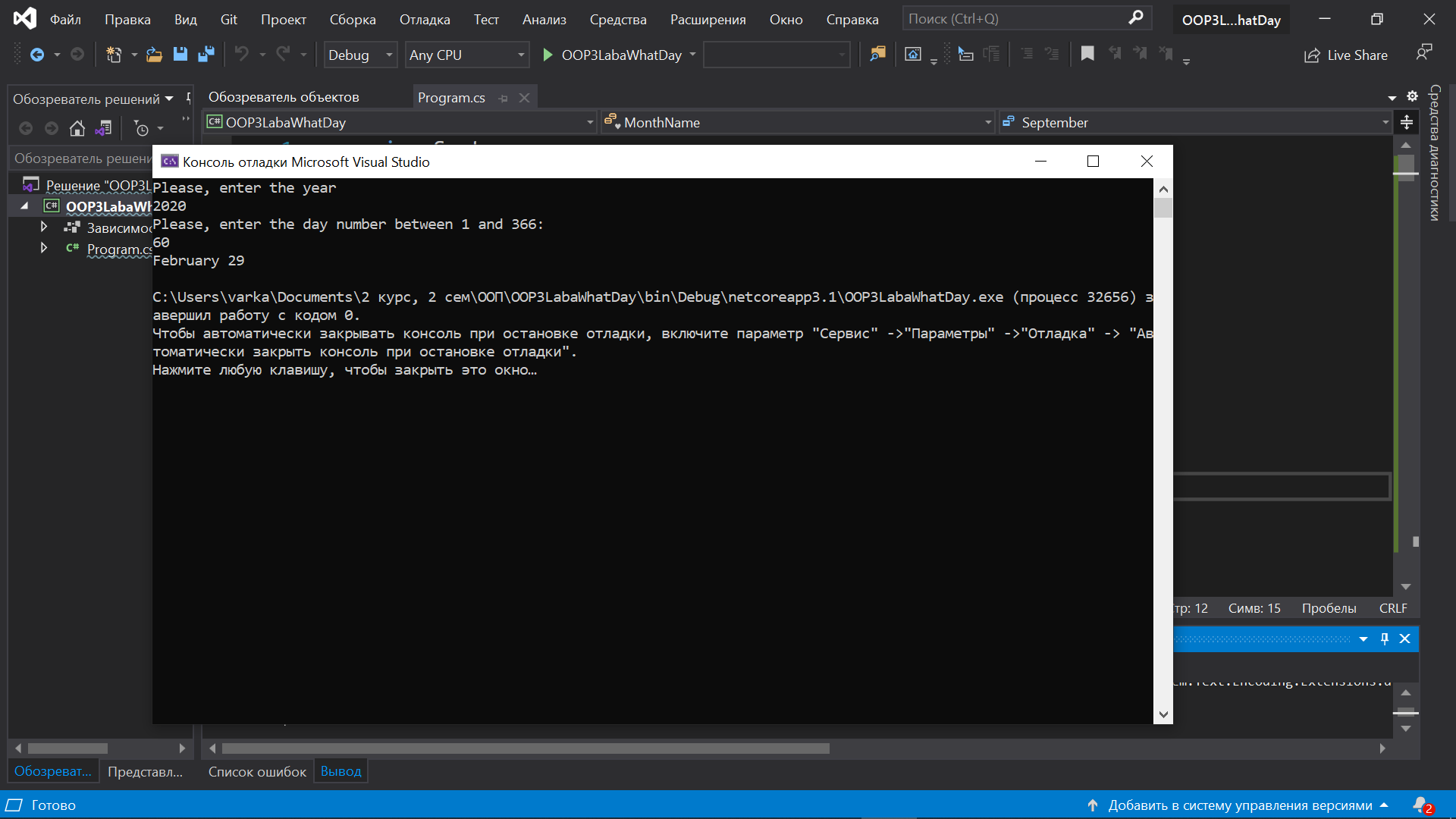


Рис. 6.2 — пример работы программы с високосным годом.

На Рис. 6.3 показана работа программы при неверных входных данных:

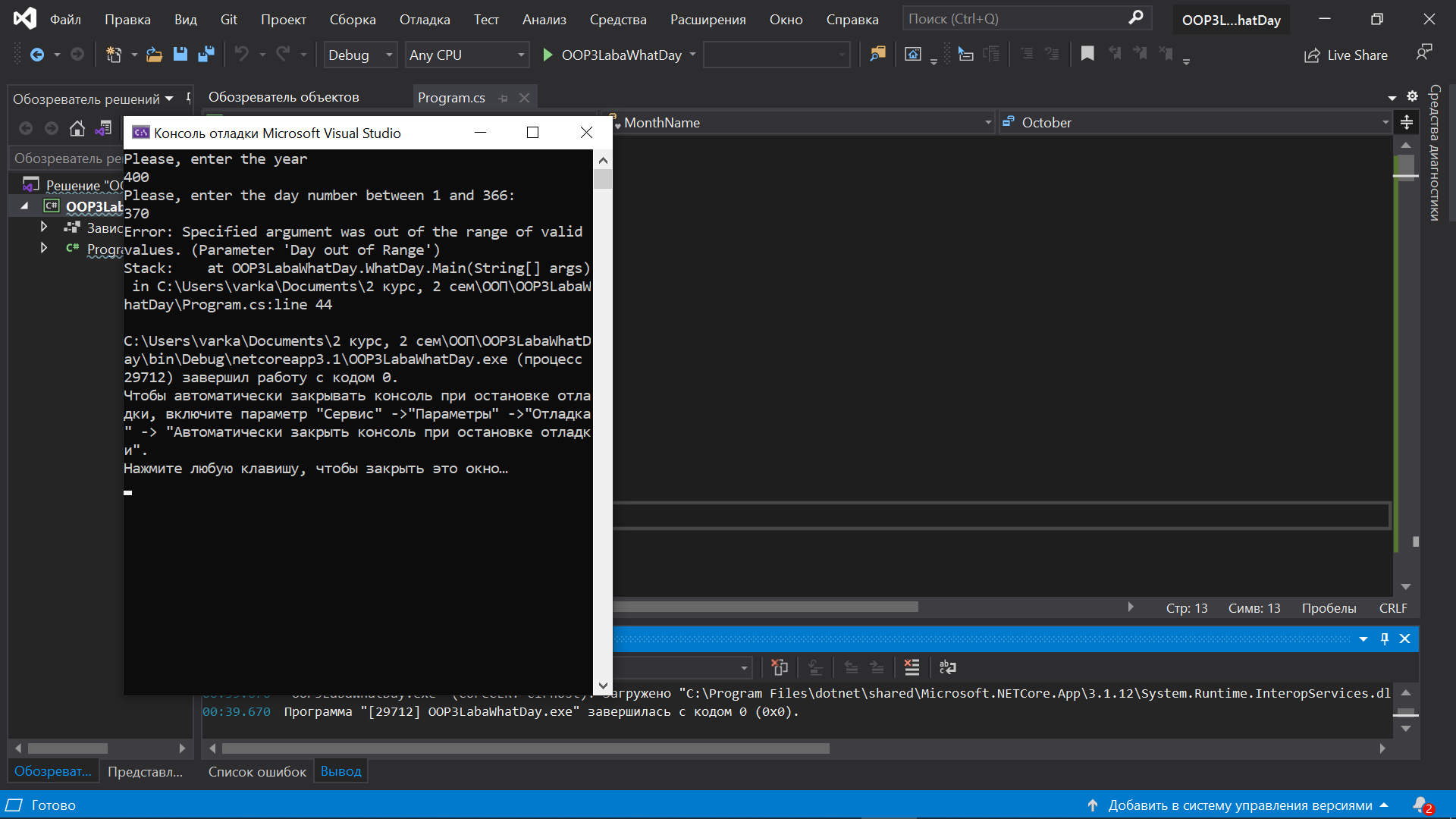


Рис. 6.3 — пример работы программы при неверных входных данных.

# Полученные результаты

В результате работы программы пользователю выводится месяц и день, соответствующие введенным пользователем данным, либо, при некорректном числе, сообщение об ошибке, ее природе и указании на место, где она произошла.

# Выводы

В ходе выполнения данной лабораторной работы:

1. Были получены навыки использования foreach, инструкций break и goto.
2. Были получены знания и навыки написания программ на C#, использующих перечислимые типы, исключения и условные конструкции.
3. Практиковалась работа с типами данных.

# Список использованных источников

Руководство по языку C#: URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/> (дата обращения: 26.02.2021)

Материалы учебного курса по ООП: URL: <https://vec.etu.ru/moodle/course/view.php?id=5512> (дата обращения: 26.02.2021)