

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)
Кафедра экономической математики, информатики и статистики (ЭМИС)

ОБРАБОТКА ИСКЛЮЧИТЕЛЬНЫХ СИТУАЦИЙ
Отчет по лабораторной работе по дисциплине «Объектно-
ориентированное программирование»

Студент группы 549



Баулин С.К.

«__» _____ 2021 г.

Кандидат физико-
математических наук,
доцент кафедры ЭМИС

_____ Шельмина Е. А.

оценка «__» _____ 2021 г.

Томск 2021

Лабораторная работа №10

Обработка исключительных ситуаций

Цель работы: ознакомиться с темой обработки исключительных ситуаций. Выполнить задания по теме.

Теоритические сведения

Обработка исключительных ситуаций

Исключительная ситуация (или исключение) – это ошибка, которая возникает во время выполнения программы. Типичные исключительные ситуации – это «деление на ноль», «достижение конца файла», «переполнение в арифметических операциях» и т.п.

Для реализации механизма обработки исключений в язык C++ введены следующие три ключевых слова: `try` (контролировать), `catch` (ловить), `throw` (генерировать, породить, бросать, посылать, формировать).

Служебное слово `try` позволяет выделить в любом месте исполняемого текста так называемый контролируемый блок: `try { операторы }`

Среди операторов, заключенных в фигурные скобки, могут быть: описания, определения, обычные операторы языка C++ и специальные операторы генерации (выброса) исключений:

`throw выражение_генерации_исключения;`

За этим блоком обязательно находятся один или несколько обработчиков исключений, каждый из которых идентифицируется служебным словом `catch` и имеет в общем случае следующий формат:

`catch (тип_исключения имя) { операторы }`

Чтобы исключение было достаточно информативным, оно должно быть объектом класса, причем класс обычно определяется специально.

Задания

Задание 1. Для каждого варианта необходимо создать три массива `a`, `b` и `c` с размерами соответственно `n1`, `n2` и `n3` (`n1!=n2!=n3`). В массив `a` занести значения функции `f(x)` согласно варианту (при возникновении исключения

заносить нули). Массив *b* заполнить случайными числами (среди них должны быть и отрицательные числа и нули). Массив *c* формируется согласно варианту. Предусмотреть и обработать возникающие при этом исключительные ситуации (деление на ноль, корень из отрицательного числа, арифметическое переполнение, выход за пределы диапазона индексов массива и т.п.). При выводе значений массивов использовать флаги, форматирующие методы и манипуляторы.

$$\ln(x^2 - 1), x \in [0; 4], \Delta x = 0,2, c_i = a_i - 1/b_i$$

Скриншот программы и результат представлен на рисунках 1.1 – 1.2.

```

8 void show(double *a, double *b, double *c, int *size)
9 {
10     cout.setf(ios::fixed);
11     for (int i = 0; true; i++)
12     {
13         try
14         {
15             if (size[0] - 1 < i || size[1] - 1 < i || size[2] - 1 < i)
16             {
17                 throw "Выход за пределы индексов";
18             }
19             cout << setprecision(2) << left << setw(10) << a[i] << "|" << setw(10) << b[i] << setw(5) << "|" << c[i] << endl;
20         }
21         catch (const char *ex)
22         {
23             cout << ex << endl;
24             break;
25         }
26     }
27 }
28
29 double f_a(float x)
30 {
31     try
32     {
33         if (x < 0 || x > 4.01)
34         {
35             throw "Выход за пределы границ";
36         }
37         if (x * x - 1 <= 0)
38         {
39             throw "Основание меньше или равно 0";
40         }
41         else
42         {
43             return log(x * x - 1);
44         }
45     }
46     catch (const char *e)
47     {
48         cerr << e << '\n';
49         return 0;
50     }
51 }

```

Рисунок 1.1 – Скриншот кода программы

```

55
56     try
57     {
58         if (b[i] == 0)
59             throw "Zero Divide";
60         if (n[0] < i || n[1] < i)
61             throw 0;
62         return a[i] - (1 / b[i]);
63     }
64     catch (int ex)
65     {
66         return ex;
67     }
68     catch (const char *excep)
69     {
70         cout << excep << endl;
71         return 0;
72     }
73 }
74
75 int main()
76 {
77     setlocale(LC_ALL, "65001");
78     system("chcp 65001");
79     system("cls");
80     srand(time(NULL));
81
82     int i, size;
83
84     int *n = new int[3];
85     n[0] = 25;
86     n[1] = 25;
87     n[2] = 25;
88
89     double *a = new double[n[0]];
90     double *b = new double[n[1]];
91     double *c = new double[n[2]];
92
93     float x = 0.0;
94
95     for (i = 0; i < n[0]; i++)
96     {
97         a[i] = f_a(x);
98         x += 0.2;
99     }
100     for (i = 0; i < n[1]; i++)
101     {
102         b[i] = rand() % 20 + (-10);
103     }
104
105     for (i = 0; i < n[2]; i++)
106     {
107         c[i] = f_c(a, b, i, n);
108     }
109 }

```

PROБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ ТЕРМИНАЛ КОНСОЛЬ ОТЛАДКИ

2.22	-1.00	3.22
2.36	9.00	2.25
2.48	6.00	2.31
2.60	9.00	2.49
2.71	9.00	2.60
0.00	2.00	-0.50
0.00	7.00	-0.14
0.00	-4.00	0.25
0.00	1.00	-1.00

Выход за пределы индексов
PS C:\Users\seron\Desktop\study\3sem\OOP\laba10> █

Рисунок 1.2 – Скриншот кода и результата работы программы

Задание 2. Осуществить перевод из одной системы счисления в другую.

Написать функцию, которая должна генерировать исключение некорректных

значений параметров. Приведите пример некорректного использования. При выводе значений переменных использовать флаги, форматирующие методы и манипуляторы.

Проверить правильность перевода числа из троичной системы счисления в четырнадцатеричную.

Скриншот программы и результат представлен на рисунках 2.1 – 2.2.

```

11  int _3to10(string item)
12  {
13      int sizeN = item.length();
14      int N = sizeN;
15      int temp, dec = 0;
16      try
17      {
18          for (int i = 0; i < sizeN; i++)
19          {
20              temp = (int)item[i] - 48;
21              if (temp < 0 || temp > 2)
22              {
23                  throw "Число не в троичной системе счисления!";
24              }
25              dec += temp * pow(3, N - 1);
26              N--;
27          }
28          return dec;
29      }
30      catch (const char *e)
31      {
32          cout << e << '\n';
33          return 0;
34      }
35  }
36
37  void _10to14(string input)
38  {
39      string _14, temp;
40      int item = _3to10(input);
41      try
42      {
43          if (item == 0)
44              throw 0;
45          while (item > 0)
46          {
47              temp += to_string(item % 14);
48              item /= 14;
49          }
50          for (int i = temp.length() - 1; i >= 0; i--)
51          {
52              _14 += temp[i];
53          }
54          cout << "Число " << input << " из 3 системы счисления в 14: " << _14 << endl;
55      }
56      catch (int e)
57      {
58          cerr << "Число не может быть преобразовано в 14 систему счисления!" << endl;
59      }
60  }
61

```

Рисунок 2.1 – Скриншот кода программы

```

62  int main()
63  {
64      setlocale(LC_ALL, "65001");
65      system("chcp 65001");
66      system("cls");
67
68      string input;
69      cout << "Введите число в троичной системе счисления: ";
70      cin >> input;
71      _10to14(input);
72
73      return 0;
74  }

```

ПРОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ ТЕРМИНАЛ КОНСОЛЬ ОТЛАДКИ

Введите число в троичной системе счисления: 201
Число 201 из 3 системы счисления в 14: 15

Рисунок 2.2 – Скриншот кода и результата работы программы

Задание 3. В задании 1 из лабораторной работы 7 добавить обработку исключений согласно вашему варианту. При выводе значений переменных использовать флаги, форматирующие методы и манипуляторы.

Учесть исключительную ситуацию, когда стоимость перевозки больше 30000 руб. Написать обработчик этой ситуации и в нём вывести название транспортного средства с пометкой «дорогой переезд».

Скриншот программы и результат представлен на рисунках 3.1 – 3.2.

Задание 4. Даны два выражения Z_1 и Z_2 . Написать функции для вычисления этих выражений с организацией обнаружения нештатной ситуации (деление на ноль) и ее обработки. Передача аргументов в функции – по ссылкам.

$$z_1 = \frac{x^2 + 2x - 3 + (x+1)\sqrt{x^2 - 9}}{x^2 - 2x - 3 + (x-1)\sqrt{x^2 - 9}}, \quad z_2 = \sqrt{\frac{x+3}{x-3}}$$

Скриншот программы и результат представлен на рисунке 4.

```

7  class TransportCompany
8  {
9  protected:
10     double speed, price, distance;
11     string type, city1, city2;
12
13 public:
14     TransportCompany()
15     {
16         speed = price = 0;
17         type = "";
18     }
19     virtual void Write()
20     {
21         cout << "=====" << endl;
22         cout << type << endl;
23         cout << "Скорость " << speed << " км/ч" << endl;
24         cout << "Плата " << price << " руб/ч" << endl;
25     }
26     float Time()
27     {
28         float time = distance / speed;
29         return int(time * 100 + 0.5) / 100.0;
30     }
31     float Cost()
32     {
33         return Time() * price;
34     };
35
36     virtual void CheckCount()
37     {
38         if (Cost() > 30000)
39         {
40             throw "На " + type + " дорогой проезд.";
41         }
42         cout << "=====" << endl;
43         cout << type << endl;
44         cout << "Скорость " << speed << " км/ч" << endl;
45         cout << "Плата " << price << " руб/ч" << endl;
46         cout << "Из города " << city1 << " в город " << city2 << endl;
47         cout << "Путь " << distance << " км," << endl;
48         cout << "Займёт времени " << Time() << " ч." << endl;
49         cout << "Стоимость " << Cost() << " руб." << endl;
50     }
51 };
52
53 You, 7 hours ago | 1 author (You)
54 class Airplane : public TransportCompany
55 {
56 public:
57     Airplane()
58     {
59         speed = 800.00;
60         price = 2800;
61         type = "Airplane";
62     }
63     Airplane(string c1, string c2, double d)
64     {
65         speed = 800.00;
66         price = 2800;
67         type = "Airplane";
68         city1 = c1;
69         city2 = c2;
70         distance = d;
71     }

```

Рисунок 3.1 – Скриншот кода программы

```

You, 7 hours ago | 1 author (You)
94 class Car : public TransportCompany
95 {
96 public:
97     Car()
98     {
99         speed = 80.00;
100         price = 960;
101         type = "Car";
102     }
103     Car(string c1, string c2, double d)
104     {
105         speed = 80.00;
106         price = 960;
107         type = "Car";
108         city1 = c1;
109         city2 = c2;
110         distance = d;
111     }
112 };
113
114 int main()
115 {
116     setlocale(LC_ALL, "65001");
117     system("chcp 65001");
118     system("cls");
119
120     TransportCompany *arr[3]{
121         arr[0] = new Airplane("Москва", "Париж", 2487),
122         arr[1] = new Train("Москва", "Париж", 2509),
123         arr[2] = new Car("Москва", "Париж", 2827)};
124     try
125     {
126         for (int i = 0; i < 3; i++)
127         {
128             arr[i]->CheckCount();
129             cout << endl;
130         }
131     }
132     catch (string ex)
133     {
134         cout << ex << endl;
135     }
136
137     return 0;
138 }
You, 7 hours ago • laba 9-10

```

ПРОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ ТЕРМИНАЛ КОНСОЛЬ ОТЛАДКИ

```

=====
Airplane
Скорость 800 км/ч
Плата 2800 руб/ч
Из города Москва в город Париж
Путь 2487 км.
Займёт времени 3.11 ч.
Стоимость 8708 руб.

=====
Train
Скорость 220 км/ч
Плата 1700 руб/ч
Из города Москва в город Париж
Путь 2509 км.
Займёт времени 11.4 ч.
Стоимость 19380 руб.

На Car дорогой проезд.
PS C:\Users\seron\Desktop\study\3sem\OOP\laba10>

```

Рисунок 3.2 – Скриншот кода и результата работы программы


```

8  {
9
10     if (x * x - 2 * x - 3 + (x - 1) * sqrt(x * x - 9) == 0)
11     {
12         throw "Деление на ноль в функции z1";
13     }
14     if (x * x - 9 < 0)
15     {
16         throw "Корень из отрицательного числа в функции z1";
17     }
18     return (x * x + 2 * x - 3 + (x + 1) * sqrt(x * x - 9)) / (x * x - 2 * x - 3 + (x - 1) * sqrt(x * x - 9));
19 }
20
21 float z2(float x)
22 {
23
24     if (x - 3 == 0)
25     {
26         throw "Деление на ноль в функции z2";
27     }
28     if ((x + 3) / (x - 3) < 0)
29     {
30         throw "Корень из отрицательного числа в функции z2";
31     }
32     return sqrt((x + 3) / (x - 3));
33 }
34
35 int main()
36 {
37     setlocale(LC_ALL, "65001");
38     system("65001");
39     system("cls");
40
41     cout.precision(5);
42
43     float x, result;
44
45     cout << "Введите X: ";
46     cin >> x;
47
48     try
49     {
50         result = z1(x);
51         cout << "Значение функции z1 = " << result << endl;
52     }
53     catch (const char *e)
54     {
55         cerr << e << '\n';
56     }
57     try
58     {
59         result = z2(x);
60         cout << "Значение функции z2 = " << result << endl;
61     }
62     catch (const char *e)
63     {
64         cerr << e << '\n';
65     }
66
67     return 0;

```

ПРОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ ТЕРМИНАЛ КОНСОЛЬ ОТЛАДКИ

```

Введите X: 10
Значение функции z1 = 1.3628
Значение функции z2 = 1.3628
PS C:\Users\seron\Desktop\study\3sem\OOP\laba10>

```

Рисунок 4 – Скриншот кода и результата работы программы

Вывод: произведено ознакомление с темой обработки исключительных ситуаций. Выполнены задания по теме.