**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»**

**(БГТУ им. В.Г.ШУХОВА)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

дисциплина: **Объектно - ориентированное программирование**

Лабораторная работа № 7

тема: «**Исключительные ситуации в C++**»

Выполнил: ст. группы ВТ-22

Воскобойников Илья

Белгород 2020

**Цель работы:** Получение теоретических знаний об исключительных ситуациях в С++. Получение практических навыков при работе с исключениями в С++.

Задания к лабораторной работе

1. Изучить теоретические сведения об исключениях в С++.

2. Изучить самостоятельно стандартные классы для исключений предусмотренных в С++.

3. Разработать программу в соответствии с заданным вариантом задания.

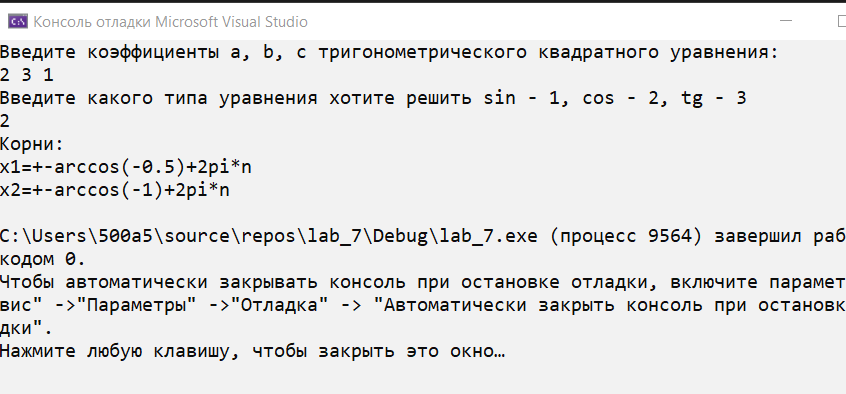
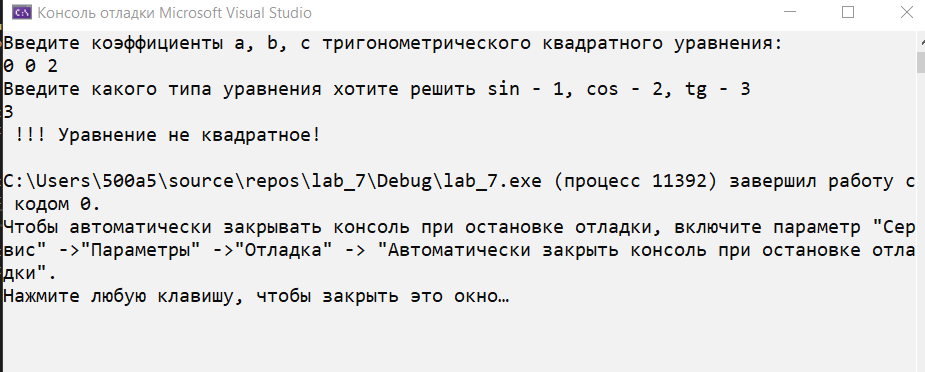
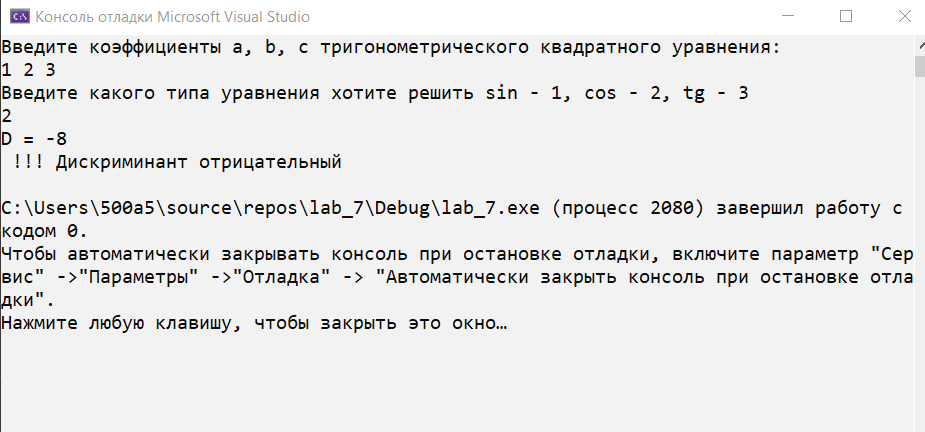
4. Оформить отчет.

**Вариант 3**

Разработать абстрактный класс решения уравнений. Реализовать класс для решения тригонометрических уравнений (потомок от абстрактного класса), предусмотреть обработку исключений при решении тригонометрических уравнений (ОДЗ тригонометрических функций).

**Код**

1. #include <iostream>
2. #include <locale.h>
3. #include <math.h>
4. #include <cstdlib>
5. using namespace std;
6. class EquatException {
7. protected:
8. string str;
9. public:
10. EquatException(string mess) :
11. str(mess) {};
12. string error() {
13. return (" !!!" + str);
14. };
15. };
16. class FirstNull : public EquatException {
17. public:
18. FirstNull(string) :
19. EquatException(" Уравнение не квадратное! ") {};
20. };
21. class NegDiscr : public EquatException {
22. public:
23. NegDiscr(string) :
24. EquatException(" Дискриминант отрицательный ") {};
25. };
26. class TrigODZ : public EquatException {
27. public:
28. TrigODZ(string) :
29. EquatException(" не удовлетворяет ОДЗ: [-1;1] ") {};
30. };
31. class Equations
32. {
33. public:
34. virtual void print() = 0;
35. virtual void write() = 0;
36. virtual void root() = 0;
37. virtual double findDiscrim() = 0;
38. virtual void print\_solution()=0;
39. virtual void solve\_equation() = 0;
40. };
41. class Square\_Trig\_equation : public Equations
42. {
43. private:
44. double a, b, c, x1, x2;
45. int m;
46. public:
47. void print() {
48. cout << "a = " << a << " b = " << b << " c = " << c;
49. };
50. void write() {
51. cout << "Введите коэффициенты a, b, c тригонометрического квадратного уравнения: "<<endl;
52. cin >> a;
53. cin >> b;
54. cin >> c;
55. cout<<"Введите какого типа уравнения хотите решить sin - 1, cos - 2, tg - 3 " << endl;
56. cin >> m;
57. };
58. void test\_null\_a()
59. {
60. if (a == 0.0)
61. throw FirstNull("");
62. }
63. void test\_discr(double discr)
64. {
65. if (discr < 0.0) {
66. cout << "D = " << discr << endl;
67. throw NegDiscr("");
68. }
69. }
70. void test\_trig(double x)
71. {
72. if (fabs(x) > 1.0)
73. {
74. cout << "Корень " << x;
75. throw TrigODZ("");
76. }
77. }
78. void root()
79. {
80. double d = findDiscrim();
81. test\_null\_a();
82. x1 = (-b + (sqrt(d))) / (2 \* a);
83. x2 = (-b - (sqrt(d))) / (2 \* a);
84. }
85. double findDiscrim()
86. {
87. double d;
88. d= (b \* b) - (4 \* a \* c);
89. test\_discr(d);
90. return d;
91. }
92. void print\_solution() {
93. cout << "Корни:" << endl;
94. switch (m)
95. {
96. case 1:
97. cout << "x1=((-1)^n)arcsin(" << x1 << ")+pi\*n"<<endl;
98. cout << "x2=((-1)^n)arcsin(" << x2 << ")+pi\*n"<<endl;
99. test\_trig(x1);
100. test\_trig(x2);
101. break;
102. case 2:
103. cout << "x1=+-arccos(" << x1 << ")+2pi\*n" << endl;
104. cout << "x2=+-arccos(" << x2 << ")+2pi\*n" << endl;
105. test\_trig(x1);
106. test\_trig(x2);
107. break;
108. default:
109. cout << "x1=arctg" << x1 << "+pi\*n" << endl;
110. cout << "x2=arctg" << x2 << "+pi\*n" << endl;
111. break;
112. }
113. }
114. void solve\_equation() {
115. write();
116. root();
117. print\_solution();
118. }
119. };
120. int main() {
121. setlocale(LC\_ALL, "rus");
122. Square\_Trig\_equation eq;
123. try {
124. eq.solve\_equation();
125. }
126. catch (EquatException &r) {
127. cout << r.error() << endl;
128. exit(0);
129. }
130. return 0;
131. }



**Вывод:** в ходе выполнения лабораторной работы были получены теоретические знания об исключительных ситуациях в С++. Получены практические навыки при работе с исключениями в С++.