Министерство высшего образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет» (ПНИПУ)**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

ОТЧЁТ

Тема: «Лабораторная работа №8»

Выполнил

Студент группы РИС-22-2б

Прядеин И.А.

Проверил доцент кафедры

ИТАС

Полякова О. А.

Пермь 2023

Постановка задачи

1. Реализовать класс, перегрузить для него операции, указанные в варианте.
2. Определить исключительные ситуации.
3. Предусмотреть генерацию исключительных ситуаций.

**Вариант 8:**

Класс-контейнер МНОЖЕСТВО с элементами типа int.

Реализовать операции:

[] -доступа по индексу;

== - проверка на равенство;

> число - принадлежность числа множеству;

- n - переход влево к элементу с номером n.

**UML диаграмма:**

**Скриншот результата выполнения программы**

**Ответы на вопросы:**

1. Что представляет собой исключение в C++?
2. На какие части исключения позволяют разделить вычислительный процесс? Достоинства такого подхода?
3. Какой оператор используется для генерации исключительной ситуации?
4. Что представляет собой контролируемый блок? Для чего он нужен?
5. Что представляет собой секция-ловушка? Для чего она нужна?
6. Какие формы может иметь спецификация исключения в секции ловушке? В каких ситуациях используются эти формы?
7. Какой стандартный класс можно использовать для создания собственной иерархии исключений?
8. Каким образом можно создать собственную иерархию исключений?
9. Если спецификация исключений имеет вид: void f1()throw(int, double); то какие исключения может порождать функция f1()?
10. Если спецификация исключений имеет вид: void f1()throw(); то какие исключения может порождать функция f1()?
11. В какой части программы может генерироваться исключение?
12. Написать функцию, которая вычисляет площадь треугольника по трём сторонам (формула Герона).

Функцию реализовать в 4 вариантах:

* без спецификаций исключений;
* со спецификацией throw();
* с конкретной спецификацией с подходящим стандартным исключением;
* спецификация с собственным реализованным исключением.