Практическая работа №6

Организация обработки исключений

# Введение

Организация обработки исключений играет важную роль в создании устойчивых к ошибкам программ. Обработка исключений помогает предотвратить сбои программы и обеспечить корректное завершение работы, даже если возникли ошибки во время выполнения. В данной работе будет рассмотрено, как правильно организовать обработку исключений в программных модулях.

# Цель работы

Цель данной работы – изучить методы и подходы к обработке исключений, научиться применять конструкции try-catch, обрабатывать и регистрировать исключения для повышения стабильности программного обеспечения.

# Практическая часть

## Задание 1: Базовая обработка исключений

1. Создайте простой программный модуль, в котором возникает ошибка (например, деление на ноль).  
2. Реализуйте обработку исключения с помощью конструкции try-catch, чтобы предотвратить аварийное завершение программы.  
3. Выведите сообщение об ошибке и продолжите выполнение программы.

## Задание 2: Иерархия исключений

1. Изучите, как работают различные типы исключений в вашем языке программирования (например, стандартные исключения и их наследники).  
2. Создайте несколько примеров обработки различных исключений (например, деление на ноль, выход за границы массива, ошибки ввода-вывода).  
3. Задокументируйте различия между ними и предложите стратегию обработки каждой категории.

## Задание 3: Генерация пользовательских исключений

1. Создайте пользовательский класс исключений для обработки специфических ситуаций в вашем программном проекте.  
2. Напишите пример кода, который генерирует и обрабатывает это исключение.  
3. Обоснуйте необходимость создания пользовательского исключения.

## Задание 4: Логирование исключений

1. Реализуйте механизм логирования исключений в файл или в базу данных для их дальнейшего анализа.  
2. Проверьте корректность логирования и убедитесь, что каждая ошибка записывается с указанием времени, типа ошибки и контекста.  
3. Задокументируйте подходы к анализу и устранению исключений на основе логов.

## Задание 5: Обработка исключений в многопоточности

1. Если ваш проект использует многопоточность, реализуйте обработку исключений в многопоточной среде.  
2. Убедитесь, что исключения в одном потоке не приводят к завершению программы или других потоков.  
3. Используйте синхронизацию для корректной работы с исключениями в многопоточных приложениях.

# Заключение

Работа должна быть оформлена в отчет со всеми соответствующими данными по выполнению практической работы. Работа должна быть объемной и информативной, так же должен соблюдаться ГОСТ Р 7.0.97-2016.