

## Лабораторная работа 5

1) Как работают исключения? В процессе работы программы могут возникать различные ошибки. Например, при передаче файла по сети оборвется сетевое подключение или будут введены некорректные и недопустимые данные, которые вызовут падение программы. Такие ошибки еще называются исключениями. Если исключение не обработано, то при его возникновении программа прекращает свою работу.

2) Можно ли между блоком `try` и `catch` записать произвольный код? Блок **`try`** не **может** существовать при отсутствии выражения **`catch`** либо выражения `finally`. Существование какого-либо **кода** в промежутке **между блоками `try`, `catch`, `finally`** является невозможным.

3) Что такое развёртывание стека? Развертывание стека – это процесс вызова деструкторов локальных объектов, когда исключительные ситуации выводят их из области видимости.

4) Как повторно сгенерировать то же исключение? Если возникает необходимость снова сгенерировать исключения из блока, который обрабатывает исключения, можно сделать это путем вызова `throw` без указания исключения. В результате текущее исключение будет передано во внешнюю последовательность `try/catch` обработки исключений. Причиной для этого может послужить желание обрабатывать исключения несколькими обработчиками. Например, один обработчик может заниматься одним аспектом исключения, а второй обработчик — другим. Исключение может быть снова сгенерировано или изнутри блока `catch`, или из функции, вызванной в этом блоке. Когда повторно генерируется исключение, оно не будет перехвачено той же самой инструкцией `catch`. Оно будет распространяться до следующей внешней инструкции `catch`.

5) Абсолютный обработчик исключений. -абсолютный обработчик может обработать любую исключительную ситуацию. Существует также абсолютный обработчик, который совместим с любым типом исключительной ситуации. Для написания такого обработчика надо вместо аргументов написать многоточие (эллипсис).

## Лабораторная 6

**1)** Для работы с файлами в стандартной библиотеке определен заголовочный файл `fstream`, который определяет базовые типы для чтения и записи файлов. В частности, это:

`ifstream`: для чтения с файла

`ofstream`: для записи в файл

`fstream`: совмещает запись и чтение

**2)** Разница между работой с бинарными и текстовыми файлами? Текстовый **файл** — это файл, в котором данные хранятся в виде символов ASCII, и обычно используется для хранения потока символов. Текстовые файлы организованы вокруг строк, каждая из которых заканчивается символом новой строки (`'\n'`). Файлы исходного кода сами по себе являются текстовыми файлами. Двоичный **файл** — это файл, в котором данные хранятся в файле так же, как они хранятся в основной памяти для обработки. Он хранится в двоичном формате вместо символов ASCII. Обычно он используется для хранения числовой информации (`int`, `float`, `double`). Обычно двоичный файл может быть создан только внутри программы, и его содержимое может быть прочитано только программой.

**3)** Флаги открытия файлов. Для открытия файла в функцию необходимо передать путь к файлу в виде строки. И также можно указать режим открытия. Список доступных режимов открытия файла:

`ios::in`: файл открывается для ввода (чтения). Может быть установлен только для объекта `ifstream` или `fstream`

`ios::out`: файл открывается для вывода (записи). При этом старые данные удаляются. Может быть установлен только для объекта `ofstream` или `fstream`

`ios::app`: файл открывается для дозаписи. Старые данные не удаляются.

`ios::ate`: после открытия файла перемещает указатель в конец файла

`ios::trunc`: файл усекается при открытии. Может быть установлен, если

также установлен режим out

ios::binary: файл открывается в бинарном режиме

**4)** Произвольный доступ в файле. Произвольный доступ к файлу осуществляется **путем манипулирования файловым указателем с помощью функции seekg() (для ввода) и функции seekp() (для вывода)**. Если вам интересно, g означает «get» (получить), а p – «put» (положить). Функции seekg() и seekp() принимают два параметра. Первый параметр – это смещение, определяющее, на сколько байтов переместить файловый указатель. Второй параметр – это флаг ios, который указывает, от чего должен быть смещен параметр смещения.

**5)** Что такое LF, CR, их коды. CR и LF это **управляющие символы или байт-код которые можно использовать для обозначения разрыва строки в текстовых файлах.**

CR = Возврат каретки (Carriage Return) ( \r , 0x0D в шестнадцатеричной, 13 в десятичной системе счисления) — перемещает курсор в начало строки, не переходя на следующую строку.

LF = **Перевод строки (Line Feed)** (\n, 0x0A в шестнадцатеричной, 10 в десятичной системе счисления) — перемещает курсор на следующую строку, не возвращаясь в начало строки.