

Каждая курсовая работа выполняется и оценивается индивидуально. Тему работы можно выбрать из предложенных (не более 10 студентов с одной темой) или согласовать с преподавателем свою тему, которая затрагивает разные аспекты программирования.

В программном решении необходимо:

- реализовать обработку некорректного пользовательского ввода;
- использовать в реализации принципы ООП (инкапсуляция, полиморфизм, наследование);
- задействовать структуры данных из стандартной библиотеки STL (н-р, vector<>, map<> и др.);
- реализовать работу с файловой системой;

В отчете по курсовой работе должны присутствовать следующие элементы:

- описание задачи; формализация функциональности (что поддерживается, что не поддерживается);
- схематизация структуры приложения и порядка выполнения (диаграммы классов UML, диаграммы IDEF0);
- программный код в приложении.

При оценивании курсовой работы учитывается:

- качество программного решения (функциональность, структура решения, следование ООП-стилю, возможность расширения и модификации функциональности);
- оформление отчёта.

1. Разработка математического интерпретатора

Способы работы с интерпретатором: через консоль и через файл.

Поддерживается работа (ввод, вывод, операции) с типами: «векторы», матрицы, рациональные дроби.

Пример работы с интерпретатором:

```
V1 = [ 1 2 3 ]
M1 = [ 1 0 0; 0 1 0; 0 0 1 ]
V2 = M1 * V1
R = 1 / 3
V3 = V2 * R
M2 = T(M1)
V3
M2
```

2. Разработка транслятора на C++ с языка Basic

Необходимо реализовать трансляцию основных алгоритмических структур языка (условный оператор, операторы цикла, объявление переменных, вывод на консоль).

Пример кода на VisualBasic, с которым должен справиться транслятор:

```
Dim A as Integer = 3
Dim B as Integer = 0
B = 5
If B = 0 Then
    Console.WriteLine("B equals 0")
ElseIf A > 2 AND B > 0 Then
    Console.WriteLine("A greater than 2 and B greater than 0")
End If
```

3. Разработка транслятора на C++ с учебного языка на русском языке

Придумать формат основных операторов на русском языке (условный оператор, операторы цикла, объявление переменных, работа с консолью) и реализовать трансляцию в код на языке C++.

4. Разработка «геометрического калькулятора»

Поддерживается работа с основными геометрическими фигурами (точка, прямая, треугольник, прямоугольник, квадрат, ромб, трапеция, круг, многоугольник): ввод, вывод, расчёт характеристик фигур (площадь, периметр, координаты центра масс), проверка свойств фигур (н-р, является ли треугольник равнобедренным), расчёт расстояний между фигурами (между центрами масс)

Визуальное отображение фигур можно реализовать с помощью библиотеки SFML (<https://www.sfml-dev.org/>)

5. Разработка «статистического калькулятора»

Поддерживается возможность формирования случайных выборок (равномерное распределение, нормальное распределение, указанный диапазон); вычисление статистических характеристик выборки (минимум, максимум, квартили, среднее арифм. значение, медиана, дисперсия, коэффициенты асимметрии, эксцесс); тест на нормальность; оценка взаимозависимости двух показателей.

6. Разработка игры «Морской бой»

Реализовать консольную игру с компьютером. Пользователь устанавливает свои корабли через консоль или загружает из файла. Компьютер размещает корабли случайным образом (не нарушая правил размещения). Компьютер не должен делать заведомо ошибочные выстрелы. В конце игры отображаются карты пользователя и компьютера с отмеченными выстрелами.

Визуальное отображение карты можно реализовать с помощью библиотеки SFML (<https://www.sfml-dev.org/>)

7. Разработка анализатора текстовых файлов

Реализовать программу для обработки множества текстовых файлов и подсчёта статистики (ранжированный список слов по частоте встречаемости, по длине слов, подсчёт количества предложений), отображение top-most-списков (н-р, «10 самых часто встречающихся слов», «5 самых длинных слов»).

Поддерживается возможность: делать настройку обработки (учитывать или не учитывать регистр, игнорировать слова со специальными символами, числами, на другом языке), сохранять результаты обработки для возможности сравнительного анализа (н-р, 10 самых встречающихся слов в сочинениях Л.Толстого и В.Набокова)

8. Разработка записной книжки с тэгами

Реализовать возможность работы с записями (создание, сохранение в файле, загрузка, модификация). У каждой записи есть атрибуты: автор, дата создания, дата изменения, а также набор тэгов (прикрепленные ключевые слова, например, «футбол», «учёба», «другое»). Реализовать возможность поиска, фильтрации записей по автору, тэгам, дате изменения или создания («за последние 3 дня»). Подсчёт статистики по тэгам, авторам.

9. Разработка новостного агрегатора

Реализовать возможность загрузки новостных сообщений из нескольких новостных агентств в сети Интернет. Поддержка возможности фильтрации сообщений по заданным ключевым словам («Russia», «covid-19»). При повторной загрузке не должно происходить дублирования сообщений.

Рекомендуется использовать RSS-каналы, предоставляющие сообщения в стандартизованном виде (н-р, лента сообщений от “The Guardian” в формате RSS: <https://www.theguardian.com/uk/rss>).

Для выполнения Http-запросов можно использовать библиотеки для C++: LibCurl, cppRestSDK, boost.asio.

10. Разработка приложения для анализа файловой системы

Приложение реализует обработку заданных директорий в файловой системе: подсчёт количества файлов, директорий, суммарного размера в байтах (с учётом или без учёта вложенных директорий), нахождение самых больших файлов (с учётом заданного порога), самых новых файлов (для заданной даты).

Также реализуется поиск возможных дубликатов файлов (совпадение по имени, по размеру).

В файле сохраняется история и результаты поиска.