**ЗАДАНИЯ НА Курсовое проектирование**

ПМ.01 «РАЗРАБОТКА ПРОГРАММНЫХ МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ»

*специальность* *09.02.07*

Курсовой проект выполняется студентами специальности 09.02.07 в 6 семестре   
3 курса.

Всего предлагается более 180 различных тем. Для каждой темы описаны 2 варианта заданий, различающихся по особенностям исходных данных, требованиям к форме и точности представления результатов, необходимости реализации тех или иных функций (подробно описаны 60 тем). Предполагается, что выбрана итерационная стратегия конструирования, то есть все требования к программе определены вначале, затем разработка строится в виде версий. Первая версия реализует основные требования к программному продукту. Во второй версии необходимо расширить функционал программы, доработать дизайн и функциональные возможности пользовательского интерфейса, добавить расширенную проверку исходных данных, различные сервисные функции. Всего должно быть создано 2 версии программного продукта, обе версии сдаются преподавателю. Для остальных тем (61-180) требования к версиям необходимо разработать самостоятельно.

Рассмотрим подробнее ход работы над курсовым проектом.

Сначала составляется техническое задание, которое необходимо утвердить и подписать исполнителем (студентом) и заказчиком, в роли которого выступает преподаватель. В техническом задании необходимо четко описать постановку задачи: требования к исходным данным, результатам программы, функциональные характеристики проекта, требования к составу и параметрам технических и программных средств, сроки разработки и т.д. Техническое задание сдается в печатном виде.

Затем выполняется проектирование программного продукта и составляются необходимые UML-диаграммы (диаграмма прецедентов, ER-диаграмма, диаграмма классов, последовательностей, состояний, активности и др.).

Затем разрабатывается первая версия программы для варианта а). Для создания второй версии в существующую программу добавляются дополнительные функции пункта б), таким образом реализуется вторая итерация и увеличивается функциональность программы.

Если в заданиях не указаны какие-либо требования к программе, то они определяются преподавателем или студентами. Так, например, если в задании говорится, что ввод данных осуществляется из текстового файла, а точный формат файла не указан, то этот формат может быть либо задан преподавателем, либо определен студентом самостоятельно, но в любом случае требования к формату должны быть четко определены в техническом задании.

ССЫЛКА НА ПОЛНЫЙ ПЕРЕЧЕНЬ ТЕМ КУРСОВЫХ ПРОЕКТОВ:

<https://docs.google.com/spreadsheets/d/1_Vm5GYuEaSk-jDtPTRWqW5eRGDvt3zdfQBVFadFP7KY/edit?usp=sharing>

**Темы курсовых проектов**

[1. Решение системы n-линейных уравнений с n-переменными 4](#_Toc127186522)

[2. Программа для исключения из матрицы N \* M строк и столбцов с одинаковой суммой элементов 4](#_Toc127186523)

[3. Разработка библиотеки конструирования и преобразования матриц N \* M 5](#_Toc127186524)

[4. Программа умножения матрицы N \* M на вектор 5](#_Toc127186525)

[5. Программа перемножения матриц N \* M и M \* N 5](#_Toc127186526)

[6. Программа для вычисления полинома ax2+bx+c , где x - квадратная матрица 6](#_Toc127186527)

[7. Вычисление определителя матрицы N \* N и обратной матрицы 6](#_Toc127186528)

[8. Программа сортировки матрицы N \* M по возрастанию (убыванию) сумм элементов строк или столбцов 6](#_Toc127186529)

[9. Программа сортировки матрицы N \* M по возрастанию (убыванию) произведений элементов строк или столбцов 7](#_Toc127186530)

[10. Программа нахождения коэффициентов линейной модели методом наименьших квадратов 7](#_Toc127186531)

[11. Разработка библиотеки обработки и преобразования трехмерных массивов 7](#_Toc127186532)

[12. Решение системы линейных уравнений с помощью определителя матрицы по методу Крамара 8](#_Toc127186533)

[13. Программа поиска словоформ и анаграмм в заданном тексте 8](#_Toc127186534)

[14. Программа форматирования строк текста по длине расстановкой переносов слов 8](#_Toc127186535)

[15. Программа для взлома шифра Цезаря 9](#_Toc127186536)

[16. Программа для шифрования текста шифром Вернама 10](#_Toc127186537)

[17. Программа поиска в заданном словаре слов, содержащих только буквы заданного слова 11](#_Toc127186538)

[18. Программа поиска в заданном словаре слов, содержащих все буквы заданного слова, плюс любые другие 11](#_Toc127186539)

[19. Программа для игры "Угадай букву" 12](#_Toc127186540)

[20. Программа для игры "Угадай слово" 12](#_Toc127186541)

[21. Программа для игры «Поле чудес» 13](#_Toc127186542)

[22. Частотный анализ текста (символьный) 13](#_Toc127186543)

[23. Частотный анализ текста (пословный, относительная частота встречаемости каждого слова) 14](#_Toc127186544)

[24. Частотный анализ текста (пословный, относительная частота встречаемости слов разной длины) 14](#_Toc127186545)

[25. Частотный анализ текста (по предложениям) 14](#_Toc127186546)

[26. Программа для игры «Собери слово» (на англ. языке) 15](#_Toc127186547)

[27. Игровая программа компьютерного тестирования с конструктором тестов 15](#_Toc127186548)

[28. Компьютерная игра «Английское лото» 16](#_Toc127186549)

[29. Компьютерная игра «Морской бой» 16](#_Toc127186550)

[30. Приложение по работе с БД КУЛИНАРНЫХ РЕЦЕПТОВ 16](#_Toc127186551)

[31. Приложение по работе с БД ТЕАТРА 17](#_Toc127186552)

[32. Приложение по работе с БД медицинского центра 17](#_Toc127186553)

[33. Приложение по работе с БД АПТЕК 18](#_Toc127186554)

[34. Приложение по работе с БД МАГАЗИНА БЫТОВОЙ ТЕХНИКИ 18](#_Toc127186555)

[35. Приложение по работе с БД МАГАЗИНА КОМПЬЮТЕРНОЙ ТЕХНИКИ 18](#_Toc127186556)

[36. Приложение по работе с БД СПОРТИВНОЙ ШКОЛЫ 19](#_Toc127186557)

[37. Приложение управления БД ГРУЗОПЕРЕВОЗОК 19](#_Toc127186558)

[38. Приложение по работе с БД ИЗДАТЕЛЬСТВА 20](#_Toc127186559)

[39. Приложение по работе с БД «Экзамены» 20](#_Toc127186560)

[40. Приложение по работе с БД АВТОСАЛОНА 20](#_Toc127186561)

[41. Приложение по работе с БД ГОСТИНИЧНОГО БИЗНЕСА 21](#_Toc127186562)

[42. Приложение по работе с БД АВТОСЕРВИСА 21](#_Toc127186563)

[43. Приложение по работе с БД РЕСТОРАНА 22](#_Toc127186564)

[44. Приложение по работе с БД ТЕХНИКУМА 22](#_Toc127186565)

[45. Численное интегрирование произвольной функции 22](#_Toc127186566)

[46. Определение статистических характеристик одномерной выборки 23](#_Toc127186567)

[47. Проверка нормальности распределения одномерной выборки 23](#_Toc127186568)

[48. Решение дифференциального уравнения (задача Коши) 23](#_Toc127186569)

[49. Моделирование игры «Life» 24](#_Toc127186570)

[50. Моделирование системы массового обслуживания методом Монте-Карло 24](#_Toc127186571)

[51. Программа-сервис прогнозирования финансовых активов 25](#_Toc127186572)

[52. Программа симуляции распространения инфекции на основе мультиагентной модели 25](#_Toc127186573)

[53. Сервис самоуничтожающихся одноразовых записок 25](#_Toc127186574)

[54. Программа клавиатурный тренажер 26](#_Toc127186575)

[55. Программа менеджер паролей 26](#_Toc127186576)

[56. Программа симуляции экосистемы на основе игры «Волчий остров» 26](#_Toc127186577)

[57. Программа автоматической классификации электронных писем 27](#_Toc127186578)

[58. Игра на основе нахождения минимального пути в графе 27](#_Toc127186579)

[59. Программа многопользовательский чат в локальной сети 27](#_Toc127186580)

[60. Программа для игры "серсо" (парные картинки) 27](#_Toc127186581)

*Далее следует подробное описание каждой темы.*

# 1. Решение системы n-линейных уравнений с n-переменными

Требуется разработать библиотеку для конструирования объекта система линейных уравнений и получения решения системы n-линейных уравнений с n-переменными. В классе хранится матрица коэффициентов и вектор свободных членов уравнений.

а) n≤20; время решения ≤5 с, исходные данные вводятся с клавиатуры; точность - 10-5; вывод результатов решения системы на экран.

б) n≤20; данные вводятся с клавиатуры или из текстового файла специального формата; должен быть обеспечен вывод исходных данных из файла на экран, их ручная корректировка и последующее сохранение в файл; факт проведения корректировки должен отображаться при выводе результатов; точность - 10-8; вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл (выводится исходная система уравнений в стандартной форме и её решение).

# 2. Программа для исключения из матрицы N \* M строк и столбцов с одинаковой суммой элементов

Требуется разработать библиотеку для конструирования объекта матрица N\*M с возможностью получить данные для матрицы случайным образом, из текстового файла, из таблицы; подсчитывать сумму элементов матрицы как по всей матрице, так и в каждом столбце или строке; исключать из матрицы строки или столбцы с одинаковой суммой элементов; выводить полученную матрицу в таблицу, в текстовый файл или на принтер; сортировать строки или столбцы матрицы по возрастанию или убыванию сумм элементов строк (столбцов).

а) m, n≤10; матрица содержит целые числа в диапазоне (-100;100); хранится в текстовом файле специального формата; требуется получить сумму всех элементов матрицы, найти и исключить из массива по желанию пользователя строки или столбцы с одинаковой суммой элементов; вывод результатов на экран.

б) m, n≤100; матрица содержит целые числа; может вводиться пользователем с клавиатуры, случайным образом в заданном интервале или из текстового файла специального формата; должна быть предусмотрена возможность вывода на экран хранящейся в файле матрицы, её корректировки и повторной записи в файл; требуется найти сумму всех элементов матрицы, сумму элементов в каждой строке или в каждом столбце, и исключить из массива по желанию пользователя строки или столбцы с одинаковой суммой элементов; сортировать строки или столбцы матрицы по возрастанию или убыванию сумм элементов строк (столбцов); осуществлять вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл.

# 3. Разработка библиотеки конструирования и преобразования матриц N \* M

Требуется разработать библиотеку для конструирования объекта матрица N\*M с возможностью получить данные для матрицы случайным образом, из текстового файла, из таблицы; подсчитывать сумму и произведение элементов как по всей матрице, так и отдельном столбце или строке с указанным индексом, на главной и побочной диагонали; добавлять в матрицу строку или столбец с заданными элементами; исключать из матрицы строку или столбец с заданным индексом; выводить полученную матрицу в таблицу, в текстовый файл или на принтер; сортировать строки или столбцы матрицы по возрастанию или убыванию сумм или же произведений элементов строк (столбцов).

а) m, n≤10; матрица содержит целые числа в диапазоне (-100;100); хранится в текстовом файле специального формата; строка или столбец задаётся в виде одномерного массива соответствующего размера, одномерный массив вводится с клавиатуры; требуется добавить в массив по желанию пользователя строку или столбец на указанное место; вывод результатов на экран.

б) m, n≤100; матрица содержит целые числа; может вводиться с клавиатуры, случайным образом в заданном интервале или из текстового файла специального формата; должна быть предусмотрена возможность вывода на экран хранящейся в файле матрицы, её корректировки и повторной записи в файл; требуется подсчитывать сумму и произведение элементов как по всей матрице, так и отдельном столбце или строке с указанным индексом, на главной и побочной диагонали; добавлять в матрицу строку или столбец с заданными элементами; исключать из матрицы строку или столбец с заданным индексом; выводить полученную матрицу в таблицу, в текстовый файл или на принтер; сортировать строки или столбцы матрицы по возрастанию или убыванию сумм или же произведений элементов строк (столбцов); вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл.

# 4. Программа умножения матрицы N \* M на вектор

а) m, n≤10; матрица и вектор содержат целые числа в диапазоне (-100;100); хранятся в текстовых файлах специального формата; вывод результатов на экран.

б) m, n≤100; матрица и вектор содержат целые числа; могут вводиться случайным образом в заданном диапазоне, с клавиатуры или из текстовых файлов; должна быть предусмотрена возможность вывода на экран хранящихся в файлах матрицы и вектора, их корректировки и повторной записи в файлы; вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл.

# 5. Программа перемножения матриц N \* M и M \* N

а) m, n≤10; обе матрицы содержат целые числа в диапазоне (-100;100); хранятся в текстовых файлах специального формата; вывод результатов на экран.

б) m, n≤100; матрицы содержит целые числа; могут вводиться случайным образом в заданном диапазоне, с клавиатуры или из текстового файла; должна быть предусмотрена возможность вывода на экран хранящихся в файле матриц, их корректировки и повторной записи в файлы; вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл.

# 6. Программа для вычисления полинома ax2+bx+c , где x - квадратная матрица

а) n≤20; матрица содержит целые числа в диапазоне (-100;100); хранится в текстовом файле специального формата; коэффициенты полинома вводятся с клавиатуры; вывод результатов на экран.

б) n≤20; матрица содержит целые числа; может вводиться с клавиатуры, случайным образом в заданном интервале или из текстового файла специального формата; коэффициенты полинома вводятся с клавиатуры; должна быть предусмотрена возможность вывода на экран хранящейся в файле матрицы, ее корректировки и повторной записи в файл; вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл.

# 7. Вычисление определителя матрицы N \* N и обратной матрицы

а) n≤20; матрица содержит целые числа в диапазоне (-100;100); хранится в текстовом файле специального формата; вывод результатов на экран.

б) n≤100; матрица содержит целые числа; может вводиться с клавиатуры, случайным образом в заданном интервале или из текстового файла специального формата; должна быть предусмотрена возможность вывода на экран хранящейся в файле матрицы, ее корректировки и повторной записи в файл; вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл.

# 8. Программа сортировки матрицы N \* M по возрастанию (убыванию) сумм элементов строк или столбцов

Требуется разработать библиотеку для конструирования объекта матрица N\*M с возможностью получить данные для матрицы случайным образом, из текстового файла, из таблицы; подсчитывать сумму каждой строки/каждого столбца матрицы; выводить полученную матрицу в таблицу, в текстовый файл или на принтер; сортировать строки или столбцы матрицы по возрастанию или убыванию сумм элементов строк (столбцов).

а) m, n≤100; матрица содержит целые числа в диапазоне (-100;100); хранится в текстовом файле специального формата; требуется отсортировать строки/столбцы матрицы по желанию пользователя по возрастанию или по убыванию сумм элементов строк/столбцов; вывод результатов на экран.

б) m, n≤100; матрица содержит целые числа, может вводиться с клавиатуры, случайным образом в заданном интервале или из текстового файла специального формата; требуется отсортировать строки/столбцы матрицы по желанию пользователя по возрастанию или по убыванию сумм элементов строк/столбцов; должна быть предусмотрена возможность вывода на экран хранящейся в файле матрицы, ее корректировки и повторной записи в файл; вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл.

# 9. Программа сортировки матрицы N \* M по возрастанию (убыванию) произведений элементов строк или столбцов

Требуется разработать библиотеку для конструирования объекта матрица N\*M с возможностью получить данные для матрицы случайным образом, из текстового файла, из таблицы; подсчитывать произведение каждой строки/каждого столбца матрицы; выводить полученную матрицу в таблицу, в текстовый файл или на принтер; сортировать строки или столбцы матрицы по возрастанию или убыванию произведений элементов строк (столбцов).

а) m, n≤100; матрица содержит целые числа в диапазоне (-100;100); хранится в текстовом файле специального формата; требуется отсортировать столбцы матрицы по желанию пользователя по возрастанию или по убыванию произведений элементов строк/столбцов; вывод результатов на экран.

б) m, n≤100; матрица содержит целые числа; может вводиться с клавиатуры, случайным образом в заданном интервале или из текстового файла специального формата; требуется отсортировать столбцы матрицы по желанию пользователя по возрастанию или по убыванию произведений строк/столбцов; должна быть предусмотрена возможность вывода на экран хранящейся в файле матрицы, ее корректировки и повторной записи в файл; вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл.

# 10. Программа нахождения коэффициентов линейной модели методом наименьших квадратов

а) требуется найти значение функции в соответствии с линейной моделью в заданной точке; таблица эмпирических значений вводится из файла специального формата; вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл.

б) требуется найти значения функции в соответствии с линейной моделью в заданных точках; таблица абсцисс этих точек хранится в текстовом файле; в другом файле хранится таблица эмпирических значений (до 100 точек); вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл. Предусмотреть построение графика функции по указанным точкам, настройку внешнего вида графика и сохранение его в виде графического файла.

# 11. Разработка библиотеки обработки и преобразования трехмерных массивов

Требуется разработать библиотеку для конструирования объекта трехмерная матрица с возможностью получить данные для матрицы случайным образом, из текстового файла, из таблиц; суммы элементов главной и побочной диагонали, а также перестановку столбцов матрицы по возрастанию или убыванию сумм элементов в горизонтальном или вертикальном направлении; выводить полученную матрицу в таблицы, в текстовый файл или на принтер.

а) n,m,k≤20; трехмерная матрица содержит целые числа; данные хранятся в текстовом файле специального формата; требуется организовать вычисление суммы элементов главной и побочной диагонали; вывод результатов на экран.

б) n,m,k≤20; трехмерная матрица содержит целые числа; может вводиться с клавиатуры, случайным образом в заданном интервале или из текстового файла специального формата; должна быть предусмотрена возможность вывода на экран хранящейся в файле матрицы, ее корректировки и повторной записи в файл; обеспечить перестановку столбцов матрицы по возрастанию или убыванию сумм элементов в горизонтальном или вертикальном направлении; вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл специального формата.

# 12. Решение системы линейных уравнений с помощью определителя матрицы по методу Крамара

Требуется разработать библиотеку для конструирования объекта система линейных уравнений и получения решения системы n-линейных уравнений с n-переменными. В классе хранится матрица коэффициентов и вектор свободных членов уравнений.

а) n≤10; время решения ≤5 с; исходные данные вводятся с клавиатуры; точность - 10-5; вывод результатов на экран.

б) n≤20; время решения ≤5 с; данные вводятся с клавиатуры или из текстового файла специального формата; должен быть обеспечен вывод исходных данных из файла на экран, их ручная корректировка и последующее сохранение в файл; факт проведения корректировки должен отображаться при выводе результатов; точность - 10-8; вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл (выводиться сама система уравнений в стандартной форме и её решение).

# 13. Программа поиска словоформ и анаграмм в заданном тексте

а) словарь хранится в текстовом файле, примерно 20 000 слов; текст вводится с клавиатуры. Требуется найти в тексте слова, имеющие в том же тексте свои словоформы, т.е. включающие все буквы исходного слова в том же порядке плюс любые другие справа или слева; или анаграммы (слова, полученные перестановкой букв, например, корвет-вектор, сорт-рост), с проверкой на наличие таких слов в словаре. Пользователь выбирает режим поиска: поиск словоформ или анаграмм. Вывод результатов поиска на экран.

б) словарь хранится в текстовом файле, примерно 90 000 слов; должна быть возможность выбора словаря; добавления слов в словарь, причем если вводимое с клавиатуры слово уже есть в словаре, то второй раз оно в словарь не записывается; удаление слов из словаря; поиска слова в словаре и отображение словаря, начиная с указанной буквы или слога (слова). Текст вводится с клавиатуры или загружается из файла по выбору пользователя; должна быть возможность корректировки текста и повторной записи его в файл. Требуется найти в тексте слова, имеющие в том же тексте свои словоформы или анаграммы, с проверкой на наличие таких слов в словаре. Вывод результатов поиска на экран, а по желанию пользователя – на принтер или в текстовый файл.

# 14. Программа форматирования строк текста по длине расстановкой переносов слов

Как показывают многочисленные эксперименты, разбиение русского слова на части для переноса с одной строки на другую с большой вероятностью выполняются правильно, если пользоваться следующими простыми приемами:

1) Две идущие подряд гласные можно разделить, если первой из них предшествует согласная, а за второй идет хотя бы одна буква (буква й при этом рассматривается вместе с предшествующей гласной как единое целое).

2) Две идущие подряд согласные можно разделить, если первой из них предшествует гласная, а в той части слова, которая идет за второй согласной, имеется хотя бы одна гласная (буквы ь, ъ вместе с предшествующей согласной рассматриваются как единое целое).

3) Если не удается применить пункты 1), 2), то следует попытаться разбить слово так, чтобы первая часть содержала более чем одну букву и оканчивалась на гласную, а вторая содержала хотя бы одну гласную.

Вероятность правильного разбиения увеличивается, если предварительно воспользоваться хотя бы неполным списком приставок, содержащих гласные, и попытаться прежде всего выделить из слова такую приставку.

а) словарь хранится в текстовом файле, примерно 20 000 слов; текст на русском языке пользователь может ввести с клавиатуры или загрузить из текстового файла. Выполнить форматирование его строк по длине с помощью переноса слов. Результат сохранить в тот же или в другой файл по выбору пользователя.

б) словарь хранится в текстовом файле, примерно 90 000 слов, должна быть возможность выбора словаря. Текст на русском языке пользователь может ввести с клавиатуры или загрузить из текстового файла. Должна быть возможность работы со словарём: добавления слов в словарь, причем если вводимое с клавиатуры слово уже есть в словаре, то второй раз оно в словарь не записывается; удаления слов из словаря; поиска слова в словаре и отображение словаря, начиная с указанной буквы или слога (слова). Должна быть возможность произвести поиск в тексте слов (с параметрами: с учетом регистра, только слово целиком, все словоформы) и выделения найденных слов в тексте. Должна быть возможность замены слов в тексте на другое слово (поиск и замена). Программа должна предоставлять функцию проверки орфографии с использованием подключенного словаря и подчеркивать слова, которых нет в словаре.

# 15. Программа для взлома шифра Цезаря

Юлий Цезарь был, якобы первым, кто придумал собственно шифр.

**Алгоритм шифрования:**алфавит размещается на круге по часовой стрелке (при этом в русском алфавите, после А идет Б, а после Я - А). Для зашифровки буквы текста заменяются буквами, отстоящими по кругу на заданное число букв дальше по часовой стрелке. Если, скажем, сдвиг на 3, то вместо i-й используется (i+3)-я буква, например, вместо А пишется Г а вместо Я пишется В. При расшифровке наоборот берут букву на заданное число букв ближе, т. е. двигаясь против часовой стрелки.

Шифр Цезаря расшифровать легко. Известны вероятности букв p[i],i =1,2,...,n, в языке сообщения ( n - число букв в алфавите). Подсчитаем частоты букв f[i] в зашифрованном сообщении. Если оно не очень короткое, то f[i] должны сравнительно хорошо согласовываться с p[i]: f[i] = p[i-s] для некоторого сдвига s. Затем начнем делать перебор по сдвигам. Когда сдвиг не угадан, общее различие между p[i] и f[i+s] , равное D(s) = S | p[i] - f[i+s] | (суммирование берется по всем i от 1 до n), будет велико, а когда сдвиг угадан – мало. Минимизация D(s) по всем s = 1,2,...,n дает ключ к расшифровке кода Цезаря.

Напишите и испытайте программу взлома шифра Цезаря.

а) зашифрованный текст вводится с клавиатуры или загружается из текстового файла по выбору пользователя; программа подсчитывает частоты букв в зашифрованном тексте и подбирает наиболее подходящий ключ к расшифровке кода Цезаря; расшифрованный текст выводиться в отдельное поле на экран, а по желанию пользователя в текстовый файл или на принтер.

б) зашифрованный текст вводится с клавиатуры или загружается из файла по выбору пользователя; должна быть возможность корректировки текста и повторной записи его в файл с выбором кодировки при открытии и сохранении; должна быть функция шифрования исходного текста со сдвигом на k-позиций по алфавиту (используя русский или английский алфавит по выбору пользователя), и функция дешифрования либо с указанием ключа, либо методом подбора, если ключ неизвестен.

# 16. Программа для шифрования текста шифром Вернама

**Шифр Вернама** — это метод шифрования алфавитного текста, один из методов подстановки для преобразования обычного текста в зашифрованный текст.

В этом механизме мы присваиваем номер каждому символу открытого текста, например   
(a = 0, b = 1, c = 2, ... z = 25).

Способ получения ключа: в алгоритме шифрования Вернама мы берем ключ для шифрования открытого текста, длина которого должна быть равна длине обычного текста.

**Алгоритм шифрования:**

1. Присвойте номер каждому символу открытого текста и ключу в соответствии с алфавитным порядком.
2. Побитовое XOR для обоих чисел (соответствующего номера символа обычного текста и номера ключевого символа).
3. Вычтите число из 26, если полученное число больше или равно 26, если это не так, оставьте его.

Алгоритм одноразового ввода - это усовершенствование [шифра](https://translated.turbopages.org/proxy_u/en-ru.ru.2ae6ca6f-63eb2a72-d486e4c1-74722d776562/https/www.geeksforgeeks.org/vernam-cipher-in-cryptography/) Вернама, предложенное офицером армии связи Джозефом Моборном. Это единственный доступный алгоритм, который не поддается взлому (полностью безопасен).

Два требования к одноразовому блокноту:

* Ключ должен генерироваться случайным образом до тех пор, пока его размер не будет равен размеру сообщения;
* Ключ должен использоваться для шифрования и дешифрования одного сообщения, а затем он отбрасывается.

Таким образом, для шифрования каждого нового сообщения требуется новый ключ той же длины, что и новое сообщение в одноразовом вводе. Зашифрованный текст, сгенерированный одноразовым блокнотом, является случайным, поэтому он не имеет никакой статистической связи с обычным текстом.

а) исходный текст вводится с клавиатуры или загружается из файла по выбору пользователя; программа представляет собой одноразовый блокнот с основными функциями редактирования текста: работа с буфером обмена, создание, открытие, сохранение файла и т.п.; должна быть возможность корректировки текста и повторной записи его в файл с выбором кодировки при открытии и сохранении; ключ должен генерироваться случайным образом до тех пор, пока его размер не будет равен размеру сообщения; ключ должен использоваться для шифрования и дешифрования только одного сообщения, а затем он отбрасывается; зашифрованный текст выводиться в отдельное поле на экран, а по желанию пользователя в текстовый файл или на принтер.

б) зашифрованный текст вводится с клавиатуры или загружается из файла по выбору пользователя; должна быть функция дешифрования текста с указанием ключа; должна быть возможность пересылки зашифрованного сообщения по электронной почте, ключ пересылается отдельно.

# 17. Программа поиска в заданном словаре слов, содержащих только буквы заданного слова

Программа должна найти в заданном словаре слова, содержащие только буквы заданного слова. Например, дано слово "программа", из него можно составить следующие слова: пар, паром, гамма, грамм, рог, гора, рама и др.

а) словарь хранится в текстовом файле - каждое слово на отдельной строке; содержит примерно 20000 слов. Слово для поиска вводится с клавиатуры. Вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл.

б) словарь хранится в текстовом файле, каждое слово на отдельной строке, примерно 20000 слов; должна быть предусмотрена возможность работы со словарем: добавление новых слов (причем, если вводимое с клавиатуры слово уже есть в словаре, оно не должно добавляться повторно); просмотр словаря в алфавитном порядке, начиная с заданной буквы или сочетания букв; удаление слов из словаря; подключение словаря по выбору пользователя.

Добавляется также режим игры, когда пользователю предлагается самому составить как можно больше слов из букв заданного слова (слово загадывается случайным образом из словаря, но должно быть достаточно длинным). Пользователю разрешается вводить только те слова, которые есть в словаре. Программа сверяет список слов с тем списком, который составлен автоматически с помощью поиска, и если пользователь отгадал все слова, засчитывается победа, если не все, подсчитывается процент правильных ответов. Также можно ограничить сеанс игры по времени, то есть нужно успеть отгадать все слова за определенное время.

# 18. Программа поиска в заданном словаре слов, содержащих все буквы заданного слова, плюс любые другие

Например, дано слово "парк", все его буквы содержатся в следующих словах: "парик", "кипарис", "припарка" и др.

а) словарь хранится в текстовом файле, примерно 20000 слов; должна быть возможность добавления слов в словарь, причем если вводимое с клавиатуры слово уже есть в словаре, то второй раз оно в словарь не записывается, но для него производится поиск слов, содержащих все буквы заданного слова плюс любые другие. Слово для поиска вводится с клавиатуры. Вывод результатов на экран, а по желанию пользователя – на принтер или в файл.

б) словарь хранится в текстовом файле, примерно 20000 слов; перед каждым словом указывается его "класс" - цифра от 0 до 9; должна быть предусмотрена возможность поиска как по всему словарю, так и только среди определенного класса; должна быть возможность просмотра словаря, начиная с заданной буквы или слога (слова); удаления слов из словаря; добавления слов в словарь, причем если вводимое с клавиатуры слово уже есть в словаре, то второй раз оно в словарь не записывается, но для него производится поиск слов, содержащих все буквы заданного слова плюс любые другие. Слово для поиска вводится с клавиатуры. Вывод результатов на экран, а по желанию пользователя – на принтер или в текстовый файл.

Добавляется также режим игры, когда пользователю предлагается самому составить как можно больше слов из букв заданного слова (слово из определенного количества букв загадывается случайным образом из словаря). Пользователю разрешается вводить только те слова, которые есть в словаре. Программа сверяет список слов с тем списком, который составлен автоматически с помощью поиска, пользователю нужно ввести не менее заранее определенного количества слов, например 7, в этом случае засчитывается победа. Также можно ограничить сеанс игры по времени, то есть нужно успеть ввести угаданные слова за определенное время.

# 19. Программа для игры "Угадай букву"

Человек играет против машины; машина "задумывает" букву; человек вводит с клавиатуры разные слова, и машина сообщает, есть ли задуманная буква в этом слове.

а) словарь хранится в текстовом файле - каждое слово на отдельной строке; содержит примерно 20 000 слов. Человек играет против машины; машина "задумывает" букву; человек вводит с клавиатуры разные слова, и машина сообщает, есть ли задуманная буква в этом слове. Человеку разрешено вводить только те слова, которые есть в словаре машины; иначе выводится сообщение об отсутствии слова в словаре. Работа идет в интерактивном режиме, результаты выводятся на экран.

б) словарь хранится в файле, каждое слово на отдельной строке, примерно 90 000 слов; должна быть предусмотрена возможность работы со словарем: добавление новых слов (причем, если вводимое с клавиатуры слово уже есть в словаре, оно не должно добавляться повторно); просмотр словаря в алфавитном порядке, начиная с заданной буквы или слога (слова); удаление слов из словаря; подключение словаря по выбору пользователя.

Человек играет против машины; машина "задумывает" букву; человек вводит с клавиатуры разные слова, и машина сообщает, есть ли задуманная буква в этом слове. Человеку разрешено вводить только те слова, которые есть в словаре машины; иначе выводится сообщение об отсутствии слова в словаре. Работа идет в интерактивном режиме, результаты выводятся на экран, по желанию пользователя протокол работы может записываться в файл.

# 20. Программа для игры "Угадай слово"

Человек играет против машины; машина "задумывает" слово; человек вводит с клавиатуры разные слова, а машина сообщает, сколько раз в них встречаются буквы из задуманного слова. Например, машина задумала слово "парк". Человек вводит слова и получает такие результаты: кот - 1 корт - 2 тол - 0 мех - 0 рок - 2 мах - 1 и т. д. Эта игра не очень легкая для человека, но программа для нее получается простая. Надо четко определить правила, как считать буквы, которых несколько штук во введенном или задуманном слове (например, если задумано слово "парк", а вводится слово "папа", то какой будет результат - 2 или 4).

а) словарь хранится в текстовом файле, примерно 20 000 слов; должна быть возможность добавления слов в словарь, причем если вводимое с клавиатуры слово уже есть в словаре, то второй раз оно в словарь не записывается. Человеку разрешено вводить только те слова, которые есть в словаре машины; иначе выводится сообщение об отсутствии слова в словаре. Работа идет в интерактивном режиме, результаты выводятся на экран.

б) словарь хранится в файле, каждое слово на отдельной строке, примерно 90 000 слов; должна быть предусмотрена возможность работы со словарем: добавление новых слов (причем, если вводимое с клавиатуры слово уже есть в словаре, оно не должно добавляться повторно); просмотр словаря в алфавитном порядке, начиная с заданной буквы или сочетания букв; удаление слов из словаря; подключение словаря по выбору пользователя.

Человеку разрешено вводить только те слова, которые есть в словаре машины; иначе выводится сообщение об отсутствии слова в словаре. Работа идет в интерактивном режиме, по желанию пользователя протокол работы может записываться в файл.

# 21. Программа для игры «Поле чудес»

Машина «задумывает» слово; человек называет буквы; если введенная буква есть в задуманном слове, то она «открывается». При этом идет подсчет очков. По ходу игры машина выводит подсказки, касающиеся значения заданного слова. Работа идет в интерактивном режиме, по желанию пользователя протокол работы может записываться на диск.

а) задумываются слова из словаря, хранящегося в текстовом файле; формат следующий: слово на отдельной строке начинается с 1-й позиции; затем начиная со следующей строки со 2-й позиции объяснение этого слова, которое выводится на экран перед началом игры; перед каждым ходом человека, задается случайное число от 1 до 5; если буква угадана правильно, то это число очков добавляется, иначе – отнимается.

б) задумываются слова из словаря, хранящегося в текстовом файле следующего формата: на отдельной строке – слово; затем объяснение этого слова, перед которым стоит цифра «0» – оно выводится перед началом игры; затем подсказки – дополнительные сведения о значении слова, которые выводятся по ходу игры (с помощью генератора случайных чисел, примерно через 4 хода); кроме этого, человек может «заказать» подсказку на любом ходе, но за это с него снимается количество очков, определенное на этом ходе. Перед каждой подсказкой указана цифра – от 1 до 5. Число очков на каждом ходе задается случайным образом – числа 10, 20, 25, 50, 75, 100, 200, 500; может быть также ход «+», когда человек может открыть любую букву в слове.

# 22. Частотный анализ текста (символьный)

Разработать программу для подсчёта относительной частоты встречаемости каждого символа в тексте.

а) текст вводится с клавиатуры, должна быть возможность записи его в файл; в программе имеются основные возможности работы с буфером обмена при вводе текста; вывод результатов на экран в виде сводной таблицы.

б) текст вводится по выбору пользователя либо с клавиатуры (при этом должна быть возможность записи его в файл), либо из текстового файла; производится настройка кодировки файла при открытии и сохранении; подсчитывается не только относительная частота встречаемости каждого символа в тексте, но и анализ количества и относительной частоты встречаемости буквенных символов латиницы, кириллицы, цифровых символов, знаков препинания, специальных знаков; производиться поиск удвоения символов; производиться поиск устойчивых сочетаний символов в тексте; разные виды поиска выполняются по запросу пользователя; вывод результатов на экран, а по желанию пользователя – на принтер или в файл.

# 23. Частотный анализ текста (пословный, относительная частота встречаемости каждого слова)

Подсчитывается относительная частота встречаемости каждого слова в тексте; границами слов являются символы: точка, запятая, двоеточие, точка с запятой, пробел, скобки, тире, вопросительный и восклицательный знаки.

а) текст вводится с клавиатуры, должна быть возможность записи его в текстовый файл; вывод результатов на экран, а по желанию пользователя – на принтер или в файл.

б) текст вводится по выбору пользователя с клавиатуры (при этом должна быть возможность записи его в файл), либо из текстового файла; производится настройка кодировки файла при открытии и сохранении; производится проверка слов на наличие их в словаре; должна быть возможность выбора подключаемого словаря и работы со словарём: удаление слов, добавление слов в словарь, причем если вводимое слово уже есть в словаре, оно не должно добавляться повторно; создание нового словаря; просмотр словаря в алфавитном порядке, начиная с заданной буквы или сочетания букв.

# 24. Частотный анализ текста (пословный, относительная частота встречаемости слов разной длины)

Подсчитывается относительная частота встречаемости слов разной длины (содержащих 1, 2, 3, 4 ….. буквы); границами слова являются символы: точка, запятая, двоеточие, точка с запятой, тире, пробел, скобки, вопросительный и восклицательный знаки.

а) текст вводится с клавиатуры, должна быть возможность записи его в файл; вывод результатов на экран, а по желанию пользователя – на принтер или в файл.

б) текст вводится по выбору пользователя с клавиатуры (при этом должна быть возможность записи его в файл), либо из текстового файла; производится настройка кодировки файла при открытии и сохранении; должна быть возможность анализа как всего текста, так и любой его части (часть, подлежащая анализу, выделяется с помощью мыши или клавиш со стрелками); производится проверка слов на наличие их в словаре; должна быть возможность выбора подключаемого словаря и работы со словарём: удаление слов, добавление слов в словарь, причем если вводимое слово уже есть в словаре, оно не должно добавляться повторно; создание нового словаря; просмотр словаря в алфавитном порядке, начиная с заданной буквы или сочетания букв.

# 25. Частотный анализ текста (по предложениям)

Подсчитывается относительная частота встречаемости в тексте предложений различной длины (1, 2, 3, …… слова). Границами слов являются символы: точка, запятая, двоеточие, точка с запятой, пробел, скобки, тире, вопросительный и восклицательный знаки; границами предложения – точка, вопросительный и восклицательный знаки.

а) текст вводится с клавиатуры, должна быть возможность записи его в файл; вывод результатов на экран, а по желанию пользователя – на принтер или в файл.

б) текст вводится по выбору пользователя с клавиатуры (при этом должна быть возможность записи его в файл), либо из текстового файла; производится настройка кодировки файла при открытии и сохранении; должна быть возможность анализа как всего текста, так и любой его части (часть, подлежащая анализу, выделяется с помощью мыши или клавиш со стрелками); производится проверка слов на наличие их в словаре; должна быть возможность выбора подключаемого словаря и работы со словарём: удаление слов, добавление слов в словарь, причем если вводимое слово уже есть в словаре, оно не должно добавляться повторно; создание нового словаря; просмотр словаря в алфавитном порядке, начиная с заданной буквы или сочетания букв.

# 26. Программа для игры «Собери слово» (на англ. языке)

Пользователю предлагается анаграмма некоторого слова на английском языке, необходимо расставить буквы слова в нужном порядке.

а) словарь хранится в текстовом файле - каждое слово на отдельной строке; содержит примерно 20 000 слов; должна быть предусмотрена возможность добавления слов в словарь, причем, если вводимое с клавиатуры слово уже есть в словаре, оно не должно добавляться повторно. Работа идет в интерактивном режиме, вывод результатов на экран.

б) словарь хранится в файле, каждое слово на отдельной строке, примерно 90 000 слов; слова распределены по темам; пользователю даётся подсказка, из какой темы взято загаданное слово; должна быть предусмотрена возможность работы со словарем: добавление новых слов в какую-либо тему (причем, если вводимое с клавиатуры слово уже есть в словаре, оно не должно добавляться повторно); просмотр слов по указанной теме в алфавитном порядке, начиная с заданной буквы; удаление слов из указанной темы; подключение словаря по выбору пользователя.

Работа идет в интерактивном режиме, по желанию пользователя протокол работы может записываться в файл.

# 27. Игровая программа компьютерного тестирования с конструктором тестов

Требуется разработать программу компьютерного тестирования с режимом администрирования для создания новых тестов.

а) список вопросов и вариантов ответов хранится в XML-файле. Вопросы поочерёдно выдаются на экран, пользователь выбирает один или несколько правильных вариантов ответов и отмечает их. Идёт подсчёт количества правильных ответов. По окончании теста подсчитывается процент правильных ответов и выставляется оценка по стобалльной системе. Работа идет в интерактивном режиме, вывод результатов на экран. Результаты тестирования сохраняются с возможностью просмотра результатов тестирования разных пользователей.

б) список вопросов с вариантами ответов и названия графических файлов (иллюстраций) для вопросов хранится в XML-файле. Вопросы и иллюстрации выдаются на экран в случайном порядке, без повторений, пользователь выбирает один или несколько правильных вариантов ответов и отмечает их. Идёт подсчёт количества правильных ответов. По окончании теста подсчитывается процент правильных ответов и выставляется оценка по стобалльной системе и комментарий в текстовой форме («Отлично», «Хорошо», «Удовлетворительно» или «Плохо»). Работа идет в интерактивном режиме, по желанию пользователя в тренировочном режиме могут быть показаны правильные варианты ответов после нажатия кнопки «Проверить» при обработке каждого вопроса. Предусмотрен режим администрирования для создания нового теста, добавления вопросов с вариантами ответов и иллюстраций в конкретный тест. При добавлении иллюстрации должна быть возможность выбрать на устройстве графический файл разрешенного формата, причем файл автоматически копируется в специальную папку проекта. Дополнительно должны быть предусмотрены возможности предварительной настройки теста: ограничения по времени прохождения теста, процентов правильных ответов на «Отлично», «Хорошо», «Удовлетворительно» и «Плохо». Результаты тестирования сохраняются с возможностью просмотра результатов тестирования разных пользователей и выгрузки результатов тестирования в текстовый файл или в электронную таблицу.

# 28. Компьютерная игра «Английское лото»

а) словарь русских слов хранится в текстовом файле, словарь соответствующих английских слов и список графических файлов хранится в текстовом файле. Пользователю в случайном порядке предлагаются русские слова (с картинками), он должен ввести соответствующее слово на английском языке. Идёт подсчёт количества попыток и количества правильных ответов. По окончании игры выводится процент правильных ответов. Работа идет в интерактивном режиме, результаты выводятся на экран.

б) словарь русских слов хранится в текстовом файле, слова распределены по темам; словарь соответствующих английских слов и список графических файлов хранится в текстовом файле; должна быть возможность добавления слов по определённой теме в словари, причем если вводимое с клавиатуры слово уже есть в словаре, то второй раз оно в словарь не записывается; просмотр слов по указанной теме в алфавитном порядке, начиная с заданной буквы; удаление слов из указанной темы; подключение словаря по выбору пользователя.

Пользователю в случайном порядке предлагаются русские слова по заданной теме или по всему словарю (с картинками), он должен ввести соответствующее слово на английском языке. Идёт подсчёт количества попыток и количества правильных ответов. По окончании игры выводится процент правильных ответов. Работа идет в интерактивном режиме, результаты выводятся на экран, по желанию пользователя протокол работы может записываться в файл.

# 29. Компьютерная игра «Морской бой»

а) человек играет против машины; необходимо предусмотреть два квадратных поля, на одном из которых пользователь может поместить определённое количество кораблей, на другом компьютер размещает свои корабли в случайном порядке. Пользователь указывает клетку поля, в которое производится «выстрел». Если в этом поле имеется корабль противника, клетка подсвечивается или выделяется каким-либо образом. Компьютер производит выстрелы в случайном порядке, но если произошло попадание, то последующие выстрелы должны быть вблизи первого попадания, до тех пор, пока не будет уничтожен весь корабль. Пользователь выигрывает, если ему удалось уничтожить корабли противника прежде, чем уничтожены его корабли.

б) в программе дополнительно должны быть предусмотрены возможности настройки с помощью дополнительной формы: величины поля, количества кораблей каждого вида, цветов оформления игры. Должна быть предусмотрена авторизация пользователя, сохранение результатов игры и возможность просмотра предыдущих результатов.

# 30. Приложение по работе с БД КУЛИНАРНЫХ РЕЦЕПТОВ

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

Должна быть информация о продуктах, рецептах и видах приготовляемых блюд.

* Для каждого объекта создать класс с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

В программе должно быть реализовано два режима работы: пользовательский режим для подбора рецептов по заданному списку продуктов, с расчётом веса каждого ингредиента на заданное количество порций; режим администрирования, который предполагает работу с удобной формой создания новых рецептов и редактирования данных о продуктах и видах блюд.

# 31. Приложение по работе с БД ТЕАТРА

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

Нужна информация о репертуаре театров, об актерах и режиссерах: кто в каком спектакле занят, один актер может играть в нескольких спектаклях.

* Для каждого объекта создать класс (издание, произведение и т.п.) с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации:

а). Перечень спектаклей, в которых занят определённый актёр;

Перечень спектаклей определённого театра;

б). Поиск спектаклей по фамилии режиссера;

Перечень спектаклей за указанное число, с запросом названия театра или без;

На какой спектакль самые дешёвые билеты за указанное число?

# 32. Приложение по работе с БД медицинского центра

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

БД должна содержать информацию об отделениях медицинского центра, врачах, пациентах и их посещениях.

* Для каждого объекта создать класс (врач, пациент и т.п.) с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации о пациентах:

1. по фамилии пациента;  
   по номеру отделения;  
   по диагнозу;  
   по дате посещения или по диапазону дат;
2. по совокупности признаков.

# 33. Приложение по работе с БД АПТЕК

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

База данных должна содержать информацию о лекарствах: название, вид упаковки, способ употребления, цена, изготовитель, противопоказания, срок годности. Учтите, что разные фирмы могут выпускать одно и то же лекарство по разным ценам и в разной упаковке.

* Для каждого объекта создать класс (лекарство, аптека и т.п.) с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации о лекарствах:

1. по названию лекарства определить, где его можно купить;  
   по названию лекарства информацию о том, где можно купить данное лекарство дешевле;
2. по изготовителю перечень лекарств, находящихся в данной аптеке;  
   какие лекарства определённой аптеки подлежат списанию в данном месяце.

# 34. Приложение по работе с БД МАГАЗИНА БЫТОВОЙ ТЕХНИКИ

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

БД должна содержать информацию о магазинах и товарах.

* Для каждого объекта создать класс (магазин, товар и т.п.) с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации:

1. справки о том, в каком магазине есть нужный товар и по какой цене;   
   какие товары есть в данном магазине;   
   какие магазины есть в данном районе;
2. по какой цене и сколько товара продал каждый магазин за указанный интервал времени;   
   где находится данный товар по минимальной цене.

# 35. Приложение по работе с БД МАГАЗИНА КОМПЬЮТЕРНОЙ ТЕХНИКИ

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

БД должна содержать информацию о товарах магазина компьютерной техники.

* Для каждого объекта создать класс (товары, категории товаров, поставщики, продавцы и т.п.) с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации:

1. нужного товара по его названию;   
   товаров по диапазону цен;   
   товаров по категории;

товаров по категории и по диапазону цен;

компьютеров по фильтрам (объем оперативной памяти, разрядность процессора и т.д.)

1. по какой цене и сколько товара продал каждый продавец за указанный интервал времени;

какие товары поставляет определенный поставщик.

# 36. Приложение по работе с БД СПОРТИВНОЙ ШКОЛЫ

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

БД должна содержать информацию о спортивных секциях, тренерах, днях занятий, стоимости занятий, группах и детях, занимающихся в определенных группах.

* Для каждого объекта создать класс с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации:

1. секции по фамилии тренера;   
   секции по дням занятий;   
   групп по названию секции;

групп по фамилии тренера;

секции по фамилии занимающегося.

1. группы в определенной секции по уровню подготовки.

# 37. Приложение управления БД ГРУЗОПЕРЕВОЗОК

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

БД должна содержать информацию о автомашинах, техосмотрах, водителях, заказах, заказчиках.

* Для каждого объекта создать класс с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации:

1. о заказах по фамилии водителя;   
   о заказах по фирме заказчика или фамилии контактного лица;   
   о заказах за определенный период (по диапазону дат);

о машинах по грузоподъемности;

о машинах, которые должны проходить техосмотр;

1. по нескольким признакам.

# 38. Приложение по работе с БД ИЗДАТЕЛЬСТВА

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

В издательстве издается художественная литература. Нужна информация о каждом издании: дата подписки в печать, дата выхода из печати, тираж, число страниц, художник, корректор, редактор, в какой типографии печатается. А также информация о самом произведении, авторе и жанр издания: сборник, роман, стихи, мемуары, литературная критика и т. п.

* Для каждого объекта создать класс (издание, произведение и т.п.) с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации:

1. перечень изданий по названию типографии, имени художника, редактора или корректора, а также по совокупности этих признаков;  
   перечень изданий за определённый период времени;
2. информация о произведении по его названию (части названия);  
   информация о всех произведениях определённого автора;  
   перечень произведений по жанру;  
   название издания (изданий) с максимальным тиражом.

# 39. Приложение по работе с БД «Экзамены»

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

БД должна содержать информацию о специальностях, дисциплинах, наличии экзамена по дисциплине и в каком семестре, группах и оценках за каждую сессию.

* Для каждого объекта создать класс с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации о экзаменах:

1. по названию дисциплины;  
   по номеру группы;  
   по семестру;  
   по специальности;
2. по совокупности признаков.

# 40. Приложение по работе с БД АВТОСАЛОНА

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

БД должна содержать информацию о автомобилях, их владельцах, менеджерах, продажах и покупателях.

* Для каждого объекта создать класс с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации:

1. по фамилии владельца;  
   по номеру автомобиля;  
   по марке автомобиля;

по цене в определенном диапазоне;

по статусу «продан» или «продается»  
по дате продажи или по диапазону дат;

1. по совокупности признаков.

# 41. Приложение по работе с БД ГОСТИНИЧНОГО БИЗНЕСА

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

БД должна содержать информацию о корпусах, номерах гостиницы, о постояльцах гостиницы и датах заезда и выезда.

* Для каждого объекта создать класс с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации:

1. по дате заезда;  
   по дате выезда;  
   по количеству мест в номере;

по типу номера «Люкс», «Полулюкс», «Эконом»;

по статусу «Свободен» или «Занят», для всех номеров в указанном корпусе;  
по статусу «Свободен» или «Занят», для всех номеров в указанном диапазоне дат;

1. по совокупности признаков.

# 42. Приложение по работе с БД АВТОСЕРВИСА

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

БД должна содержать информацию о работниках автосервиса, услугах по ремонту, заказах на ремонт автомобилей, сроках выполнения заказов.

* Для каждого объекта создать класс с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации:

1. по фамилии владельца автомобиля;  
   по номеру автомобиля;  
   по номеру заказа;  
   по дате оформления заказа или по диапазону дат;
2. по совокупности признаков.

# 43. Приложение по работе с БД РЕСТОРАНА

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

БД должна содержать информацию о работниках ресторана, блюдах ресторана, категориях блюд, заказах.

* Для каждого объекта создать класс с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации о заказах:

1. по фамилии клиента (гостя ресторана);  
   по номеру заказа;  
   по фамилии официанта, принявшего заказ;  
   по дате посещения ресторана или по диапазону дат;

по статусу «Обслужен» или «Принят»;

1. по совокупности признаков.

# 44. Приложение по работе с БД ТЕХНИКУМА

Разработать программу для создания и работы с собственной БД (с возможностью добавления, удаления и корректировки информации в БД).

БД должна содержать информацию о том, какие имеются специальности, какие группы обучаются по каждой специальности, какие предметы в каждой группе, количество часов на предмет (лекции и практика), списки групп, результаты последней сессии по группам, средний балл группы.

* Для каждого объекта создать класс (специальность, группа и т.п.) с описанием необходимых полей и методов доступа;
* Для реализации поиска информации подготовить соответствующие методы;
* Подготовить форму пользовательского интерфейса для ввода информации об объектах и корректировки данных в случае необходимости.

Необходимо обеспечить возможность поиска информации:

1. перечень всех предметов по номеру группы с указанием количества часов;  
   перечень групп по номеру специальности;
2. информация о предмете по его названию (части названия);  
   результаты сессии и средний балл по номеру группы;  
   название специальности по её номеру и наоборот.

# 45. Численное интегрирование произвольной функции

а) выполняется интегрирование методом трапеций, границы интегрирования и число разбиений вводятся с клавиатуры; выбор функции (под интегрального выражения) по желанию пользователя; вывод результатов на экран;

б) выбор режима расчетов: либо с заданным количеством разбиений, либо с заданной точностью, по желанию пользователя; границы интегрирования, число разбиений или точность вычислений вводятся с клавиатуры; результат должен быть получен разными методами (прямоугольников, трапеций, Симпсона, Рунге-Кутта); вывод результатов на экран (значение интеграла, число разбиений, достигнутая точность), а по желанию пользователя – на принтер или в текстовый файл.

# 46. Определение статистических характеристик одномерной выборки

Требуется рассчитать х, σ2, σ, моду, медиану, max, min, диапазон, квартили, 10% и 90% квантили, 95% и 99% доверительный интервал для среднего.

а) выборка (до 10 тыс. значений) вводится из файла специального формата, содержит вещественные числа от 1 до 10 с точностью 7-8 десятичных знаков; результаты должны быть рассчитаны с исключением и без исключения выскакивающих значений (отстоящих от среднего дальше чем на 3σ); вывод результатов на экран;

б) выборка (до 10 тыс. значений) вводится из файла специального формата, содержит вещественные числа от 1 до 10 с точностью 7-8 десятичных знаков; результаты рассчитываются для всей выборки и для произвольных ее частей (номер начального отсчета и число отсчетов вводятся с клавиатуры); вывод результатов по желанию пользователя на экран, принтер или в файл.

# 47. Проверка нормальности распределения одномерной выборки

Требуется оценить вероятность того, что данная выборка подчиняется нормальному распределению.

а) выборка (до 100 значений) вводится с клавиатуры; содержит целые числа в диапазоне от 1000 до 1 000 000; должна быть обеспечена возможность корректировки введенных значений и записи их в файл; вывод результатов на экран;

б) выборка (до 10 тыс. значений) вводится из файла специального формата; вывод результатов по желанию пользователя на экран, принтер или в файл.

# 48. Решение дифференциального уравнения (задача Коши)

Решается дифференциальное уравнение первого порядка; начальные условия задаются с клавиатуры.

а) требуется найти значения функции на заданном интервале с заданным шагом (все вводится с клавиатуры); точность - 10-5; вывод результатов по желанию пользователя на экран;

б) требуется найти значения функции в заданных точках (абсциссы которых хранятся в текстовом файле специального формата); точность - 10-5; вывод результатов по желанию пользователя на экран, принтер или в файл.

# 49. Моделирование игры «Life»

а) Есть «вселенная», представленная в виде квадрата, разбитого на квадратные же поля. Поле может быть пустым, либо на нём может жить клетка. Пользователь расставляет клетки на игровом поле перед началом игры. Каждый «день» игры рассчитывается новое поколение клеток по следующим правилам:

* на пустом поле, рядом с которым ровно 3 живые клетки, зарождается новая клетка;
* если у живой клетки есть 2 или 3 живые соседки, эта клетка продолжает жить;
* если соседей меньше 2 или больше 3, клетка умирает (от «одиночества» или от «перенаселённости» соответственно).

Результат игры должен сохраняться в файл специального формата, который позволит загрузить сохраненную игру и продолжить наблюдение.

б) Перед тем как будет рассчитано новое поколение, каждая клетка пытается найти среди окружающих её свободных полей более привлекательное на её взгляд положение и перемещается в него. Привлекательность поля зависит от количества соседей и определяется геномом клетки, в котором записано, какое количество соседей она считает комфортным. Геном представлен массивом из 9 чисел-генов, каждое из которых может принимать значение 0 (ген молчит) или 1 (ген активен). Первое (нулевой элемент) определяет привлекательность точки с 0 соседей, второе — с 1 соседей и так далее до 8.

Если ген активен, поле с соответствующим числом соседей рассматривается клеткой как привлекательное для перемещения. Если молчит, в такую точку клетка перемещаться не будет. Например, если у клетки геном [0,1,1,0,0,0,0,0,0], она будет стараться переместиться в точку, у которой есть 1 или 2 соседа. А если такой нет, останется на месте. Из точек с одинаковой привлекательностью выбирается случайная.

Геном отражается в цвете клеток. Чем более красная клетка, тем больше она любит одиночество. Чем более синяя, тем больше любит компанию. Чем более зелёная, тем ближе она к «золотой середине» — предпочтению 2 или 3 соседей. При зарождении новой клетки она получает такой же геном, как у той из 3 её соседок, которая сходила последней («кто последний завершил комбинацию, тот и папа»). Порядок хода клеток — случайный.

При этом классическая Conway's Game of Life — это предельный случай, когда у всех клеток геном [0,0,0,0,0,0,0,0,0].

# 50. Моделирование системы массового обслуживания методом Монте-Карло

а) В одноканальную систему массового обслуживания с отказами поступает пуассоновский поток заявок. Время между моментами поступления двух последовательных заявок распределено по экспоненциальному закону f(τ)=x\*е-xτ; время обслуживания заявок случайное и распределено по закону f1(t)=y\*е-yt. Найти методом Монте-Карло за время Т: среднее число обслуженных заявок; среднее время обслуживания одной заявки; вероятность обслуживания; вероятность отказа; процент обслуженных заявок и заявок, получивших отказ. Произвести K испытаний. Результаты выводятся на экран, а по желанию пользователя – на принтер или в файл.

б) В многоканальной системе для обслуживания открыты два канала или более. Предполагается, что клиенты ожидают в общей очереди и обращаются в первый освободившийся канал обслуживания.

В многоканальной системе поток заявок подчиняется пуассоновскому закону с параметром λ, а время обслуживания – экспоненциальному с параметром μ. Приходящий первым обслуживается первым, и все каналы обслуживания работают в одинаковом темпе. 6

Возможные состояния СМО S0 (все каналы свободны), S1 (один канал занят, остальные свободны), S2 (два канала заняты, остальные свободны), Sn (все каналы заняты), Sn+1 (все каналы заняты, в очереди одна заявка), Sn+2 (все каналы заняты, в очереди две заявки) и т.д.

Определить: вероятность того, что система свободна, вероятность того, что в системе находятся N заявок, вероятность того, что заявка окажется в очереди, среднее число заявок в очереди, среднее число заявок в системе, среднее время нахождения заявки в очереди, среднее число нахождения заявки в системе.

Результаты выводятся на экран, а по желанию пользователя – на принтер или в файл.

# 51. Программа-сервис прогнозирования финансовых активов

Сервис, позволяющий оценивать и прогнозировать стоимость финансовых активов, например, акций, валют, и т.п.

а) Пользователь может указывать дату и время, на которую нужен прогноз;

пользователь может выбирать 1 или более, доступных в сервисе, финансовых активов для прогнозирования;

б) Пользователь может указать временной интервал, для которого будет обновляться прогноз. Например, для 1 сек. сервис будет прогнозировать стоимость на 1 сек. вперёд, а по прошествии этой секунды актуализировать данные по активам и снова прогнозировать на 1 сек.; пользователь может экспортировать прогноз в файл специального формата.

# 52. Программа симуляции распространения инфекции на основе мультиагентной модели

а) Система состоит из отдельных агентов с заданным набором параметров, которые определяют поведение агента (подвижность, заразность, иммунитет, и т.п.). Агенты располагаются на некоторой территории, поделённой на сектора с заданным числом связей между секторами; результаты наблюдений за системой выводятся на экран в графическом режиме.

б) Предусмотреть два режима работы программы: пользовательский режим, в котором можно наблюдать за системой во времени; режим администрирования, в котором выполняются предварительные настройки параметров системы; возможно дать системе имя и сохранить её параметры в виде файла специального формата с последующей возможностью загрузить сохраненную систему и наблюдать за ней; результаты наблюдений выводятся на экран в графическом режиме и в виде статистических таблиц; возможна выгрузка результатов наблюдений в файл или на принтер.

# 53. Сервис самоуничтожающихся одноразовых записок

Сервис, позволяющий обмениваться с другими пользователями сообщениями в режиме Клиент-Сервер и Клиент-Клиент.

а) Сообщение от пользователя 1 передаётся пользователю 2. Сообщение предварительно зашифровывается. В режиме Клиент-Сервер переданное сообщение хранится на сервере указанный промежуток времени затем удаляется, если клиент 2 подключится к Серверу до времени удаления, то сообщение передаётся, а затем удаляется. Сообщения нигде не сохраняются.

б) Можно указывать адрес сервера или 2го клиента; можно устанавливать время хранения сообщений, но не более того, что указано на сервере; программа позволяет шифровать сообщения заданным ключом.

# 54. Программа клавиатурный тренажер

Реализовать клавиатурный тренажёр, наподобие «Stamina», «Соло на клавиатуре».

а) Программа должна иметь режим набора букв, слогов, слов, предложений с возможностью тренировать определенные сочетания букв кириллицы или латиницы; упражнения хранятся в виде файлов; выбор упражнений производиться пользователем; ведется статистика пользователя: какие упражнения он прошёл и с какими результатами;

б) Режим урок предполагает последовательное выполнение упражнений с увеличением уровня сложности; упражнения в одном уровне сложности берутся в случайном порядке, но для перехода на следующий уровень необходимо пройти не менее определенного количества упражнений с удовлетворительным результатом; режим свободной набивки текста с подсчетом скорости набора; режим дуэль с другим пользователем; рейтинг пользователей по скорости и ошибкам; режим администрирования для настройки параметров режима урок и добавления новых уровней и упражнений.

# 55. Программа менеджер паролей

Сервис обеспечивающий хранение и генерацию логинов и паролей.

а) программа должна обеспечивать хранение базы данных в зашифрованном виде локально или на сервисе при указании адреса; создания паролей вручную или автоматическую генерацию пароля с выбранным уровнем сложности; автоматическую вставку паролей в соответствующие сервисы; иметь возможность создания мастер паролей, защиту от брутфорса;

б) программа должна иметь возможность восстановления мастер пароля с двухуровневой аутентификацией; программа должна иметь режим автоматической перегенерации всех паролей или выборочно, с выбранной пользователем периодичностью.

# 56. Программа симуляции экосистемы на основе игры «Волчий остров»

Реализовать симуляцию экосистемы на основе известной игры «Волчий остров».

Волчий остров размером N \* N заселен дикими зайцами и волками. Имеется по несколько представителей каждого вида. Зайцы в каждый момент времени с одинаковой вероятностью 1/9 передвигаются в один из восьми соседних квадратов (за исключением участков, ограниченных береговой линией) или сидят неподвижно. Каждый заяц с вероятностью p(br) превращается в двух зайцев. Каждый волк с вероятностью p(bw) превращается в двух волков. Каждый волк передвигается случайным образом (как и заяц), пока в одном из соседних восьми квадратов не окажется заяц, за которым он охотится. Если волк и заяц оказываются в одном квадрате, волк съедает зайца и получает h «очков». В противном случае он теряет d «очков». Волки с нулевым количеством очков умирают.

а) Увеличить количество параметров «животного», например, добавить: скорость, силу, область зрения, и т.д.; характеристики потомков определяются на основании характеристик предков (применить генетические алгоритмы);

б) Добавляется режим администрирования игры: редактор карт, управление временем симуляции, добавление особей других видов (хищников, травоядных); во время игры добавляется возможность интерактивного изменения симуляции, например, перенести «животное» в другое место или добавить новое.

# 57. Программа автоматической классификации электронных писем

Разработать программу-бот, которая позволяет сортировать электронные письма по указанному признаку (или ряду признаков), а также находить письма, у которых отсутствуют некоторые маркеры и генерировать для этих отправителей автоматическую отправку ответного письма со стандартным текстом. Предполагает реализацию двух режимов работы: пользовательский режим для работы с письмами и режим администрирования для выполнения настроек.

# 58. Игра на основе нахождения минимального пути в графе

Поиск минимального пути на игровом поле из одной точки в другую. Игровое поле представляет собой прямоугольную область из клеток. В прямоугольной таблице N×M, в каждой клетке которой записано некоторое число. В начале игрок находится в левой верхней клетке. За один ход ему разрешается перемещаться в соседнюю клетку либо вправо, либо вниз (влево и вверх перемещаться запрещено). При проходе через клетку с игрока берут столько очков, какое число записано в этой клетке (очки берут также за первую и последнюю клетки его пути). Задача игрока – пройти из начальной точки игрового поля в другую, управляя движением своего игрового персонажа, таким образом, чтобы набрать минимальное количество очков.

# 59. Программа многопользовательский чат в локальной сети

Программа предназначена для обмена информацией по локальной или беспроводной сети. Приложение-сервер запускается на компьютере и работает в фоновом режиме. Клиентское приложение не требует от пользователя никаких настроек, требует только ввода ника пользователя. Приложение позволяет обмениваться текстовыми сообщениями по локальной сети, передавать файлы и демонстрировать экран своего рабочего стола. Сообщение можно передать как отдельному пользователю, так и организовать рассылку нескольким пользователям. Также отображается, какие пользователи в данный момент находятся в локальной сети.

# 60. Программа для игры "серсо" (парные картинки)

Парные карточки перемешиваются между собой и раскладываются на игровом поле в случайном порядке «рубашкой» вверх. Каждый игрок может открывать любые две карточки за один ход. Если при открытии образовалась «парочка», то есть картинки одинаковые, то карточки остаются открытыми, а игрок получает очки и делает следующий ход. Если картинки на перевернутых карточках разные, то карточки переворачиваются обратно «рубашкой» вверх и ход переходит к следующему игроку. Выигрывает тот, кто набирает больше всех парных карточек за игру.

Возможен также вариант игры с одним игроком. Игрок тренирует внимание, его задача – найти все парные карточки как можно быстрее.