

## 1. Общие требования к программам курсового проекта

- 1) **Исходные данные находятся в файле на внешнем носителе** (текущий диск, диск, внешний диск (флэшка));
- 2) Программа должна быть **многофункциональной** (способы вызова подпрограмм и обмен данными между функциями определяет разработчик)
- 3) **Интерфейс** программы должен быть таким, чтобы пользователь, запуская исполняемый (.exe) файл, мог ориентироваться лишь на инструкцию пользователя, не зная при этом всего алгоритма условий работы программы.

Задания на курсовой проект градируются по категории сложности:

1. Задания на оценку «удовлетворительно» (минимальный базовый уровень усвоения материалов дисциплины);
2. Задания на оценку «хорошо»;
3. Задания на оценку «отлично» (уровень максимальной и повышенной сложности).

Темы курсовых заданий выбираются самостоятельно студентами. Выбор темы фиксируется в списке преподавателя в срок до **15.04.2024**

Закрепление тем и утверждение Задания на курсовой проект утверждаются заведующим кафедрой не позднее **15.04.2024**

## 2. Общие требования к программам минимального базового уровня («удовлетворительно»)

- 1) Используются методы структурного программирования для **обработки числовых и символьных массивов**, операторы и функции C++. Программа разрабатывается в одном файле.
- 2) Задание включает нескольких пунктов. Пункты задания в программе разрабатываются в рамках отдельных функций. Интерфейс пользователя оформляется в виде меню.
- 3) Требования к пояснительной записке указаны в методических рекомендациях к курсовой работе по ТП;
- 4) Срок представления курсовой работы: **1 июня 2024.**
- 5) Список задач указан в **приложении 1.**

## 3. Общие требования к программам базового уровня («хорошо»)

- 1) Используются методы структурного программирования для обработки данных **структурированного типа (структуры и объединения)**, операторы и функции C++. Программа разрабатывается в одном файле.
- 2) Задание включает нескольких пунктов. Пункты задания в программе разрабатываются в рамках отдельных функций. Интерфейс пользователя оформляется в виде меню.
- 3) Требования к пояснительной записке указаны в методических рекомендациях к курсовой работе по ТП;

- 4) Срок представления курсовой работы: **1 июня 2024.**
- 5) Список задач указан в **приложении 2.**

#### 4. Общие требования к программам уровня повышенной сложности («отлично»)

- 6) Используются методы объектно-ориентированного программирования для обработки данных **пользовательского типа (классы)**, методы, дружественные функции, конструкторы и деструкторы, операторы и функции, другие составляющие ООП C++. Программа разрабатывается как **программный многофункциональный проект**, в котором, кроме файла (.cpp) с главной функцией имеется, как минимум, собственный заголовочный файл (расширение .h) и еще один файл с расширением .cpp (например, содержащий описание базовых методов класса).
- 7) Задание включает нескольких пунктов. Пункты задания в программе разрабатываются в рамках отдельных методов класса. Интерфейс пользователя оформляется в виде меню.
- 8) Требования к пояснительной записке указаны в методических рекомендациях к курсовой работе по ТП;
- 9) Срок представления курсовой работы: **1 июня 2024.**
- 10) Список задач, аналогичен задачам, указанных в **приложении 2.**

## ПРИЛОЖЕНИЕ 1 («удовлетворительно»)

## Варианты заданий

## Вариант № 1

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив вещественных значений размерностью  $[4 \times 5]$  элементов и выводит его на экран. Вещественные элементы должны содержать не более 3-х знаков после запятой;
- б) производит поиск *max* или *min* элемента в указанной строке матрицы и результат выдает на экран. Предусмотреть возможность многократного поиска, по запросу пользователя;
- в) по окончании работы сохраняет матрицу в файле до следующего сеанса работы, а результаты поиска (*max* или *min* элемента для запрашиваемых строк) – в другом файле.

## Вариант № 2

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла одномерный массив целых чисел размерностью  $[15]$  элементов и выводит его на экран. Элементы массива в файле расположены в виде одного столбца;
- б) производит поиск *max* или *min* элемента массива по запросу и результат выдает на экран. Предусмотреть возможность повторного поиска по запросу пользователя;
- в) по окончании работы сохраняет массив в файле с именем. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 5]$  элементов с двумя пробелами между ними.

## Вариант № 3

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив целых чисел размерностью  $[4 \times 5]$  элементов и выводит его на экран. Элементы массива в файле могут быть расположены в виде одного столбца или одной строки;
- б) многократно производит изменение матрицы путем замены указанного элемента на другой, вводимый с клавиатуры, и выводит измененную матрицу на экран;
- в) по окончании работы сохраняет матрицу в файле с именем. Данные этого массива сохраняются в виде матрицы, размерностью  $[4 \times 5]$  элементов с двумя пробелами между ними.

## Вариант № 4

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла одномерный массив целых чисел размерностью  $[25]$  элементов и выводит его на экран. Элементы массива в файле расположены произвольно: в виде одного столбца или одной строки;
- б) производит расчет среднего арифметического элементов массива, изменяет указанный пользователем элемент и вновь производит расчет среднего арифметического элементов нового массива. Предусмотреть многократную возможность изменения элементов массива;
- в) по окончании работы сохраняет новый массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[5 \times 5]$  элементов с двумя пробелами между ними,

а результаты вычисления среднего арифметического элементов записываются в другой файл.

### **Вариант № 5**

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив целых чисел размерностью  $[4 \times 5]$  элементов и выводит его на экран в виде матрицы. Элементы массива в файле могут быть расположены в виде одного столбца или одной строки;
- б) многократно, в зависимости от запроса, производит расчет среднего арифметического элементов указанного пользователем столбца;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[4 \times 5]$  элементов с двумя пробелами между ними, а результаты вычисления среднего арифметического элементов и номер соответствующего столбца записываются в другой файл.

### **Вариант № 6**

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла одномерный массив целых чисел размерностью  $[15]$  элементов и выводит его на экран. Элементы массива в файле расположены в виде одной строки;
- б) многократно, в зависимости от запроса, производит подсчет количества отрицательных, или положительных, или нулевых элементов массива и выдает результат на экран;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 5]$  элементов с двумя пробелами между ними, а результаты количества отрицательных, положительных и нулевых элементов массива записываются в другой файл.

### **Вариант № 7**

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив целых чисел размерностью  $[4 \times 5]$  элементов и выводит его на экран в виде матрицы. Элементы массива в файле могут быть расположены в виде одного столбца или одной строки;
- б) многократно, в зависимости от запроса, производит подсчет суммы элементов указанной строки и выдает результат на экран;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[4 \times 5]$  элементов с двумя пробелами между ними, а результаты подсчета суммы элементов указанной строки и ее номер записываются в другой файл.

### **Вариант № 8**

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла одномерный массив целых чисел размерностью  $[15]$  элементов и выводит его на экран. Элементы массива в файле расположены в виде одного столбца;
- б) производит расчет суммы элементов массива, изменяет указанный пользователем элемент и вновь производит расчет суммы элементов нового массива. Предусмотреть многократную возможность изменения элементов массива;

- в) по окончании работы сохраняет новый (последний) массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 5]$  элементов с двумя пробелами между ними, а результаты суммы элементов всех получаемых в результате выполнения п. б) массивов записываются в другой файл.

### Вариант № 9

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив целых чисел размерностью  $[3 \times 6]$  элементов и выводит его на экран в виде матрицы. Элементы массива в файле могут быть расположены произвольно;
- б) многократно производит в зависимости от запроса поиск *max* или *min* элемента указанной строки матрицы и результат выдает на экран;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 6]$  элементов с двумя пробелами между ними, а результаты поиска записываются в другой файл.

### Вариант № 10

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив целых чисел размерностью  $[3 \times 4]$  элементов и выводит его на экран в виде матрицы. Элементы массива в файле могут быть расположены произвольно;
- б) многократно производит в зависимости от запроса поиск *max* или *min* элемента указанного столбца матрицы и результат выдает на экран;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 4]$  элементов с двумя пробелами между ними, а результаты поиска *max* или *min* элемента указанных столбцов матрицы и их номера записываются в другой файл.

### Вариант № 11

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив целых чисел размерностью  $[3 \times 6]$  элементов и выводит его на экран в виде матрицы. Элементы массива в файле могут быть расположены произвольно;
- б) многократно производит изменение матрицы путем замены указанного элемента на другой, вводимый с клавиатуры и выводит измененную матрицу на экран;
- в) по окончании работы сохраняет последний измененный массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 6]$  элементов с двумя пробелами между ними.

### Вариант № 12

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив целых чисел размерностью  $[4 \times 6]$  элементов и выводит его на экран в виде матрицы. Элементы массива в файле могут быть расположены произвольно;
- б) многократно, в зависимости от запроса, производит расчет среднего арифметического элементов указанной строки;

- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[4 \times 6]$  элементов с двумя пробелами между ними, а результаты расчета среднего арифметического элементов указанных строк матрицы и их номера записываются в другой файл.

### Вариант № 13

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив вещественных значений размерностью  $[3 \times 6]$  элементов и выводит его на экран. Вещественные элементы должны содержать не более 5-и знаков после запятой;
- б) многократно, в зависимости от запроса, производит расчет среднего арифметического элементов указанного столбца;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 6]$  элементов с двумя пробелами между ними, а результаты расчета среднего арифметического элементов указанных столбцов матрицы и их номера записываются в другой файл.

### Вариант № 14

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив вещественных значений размерностью  $[3 \times 6]$  элементов и выводит его на экран. Вещественные элементы должны содержать не более 5-и знаков после запятой;
- б) многократно, в зависимости от запроса, производит подсчет количества отрицательных, или положительных, или нулевых элементов матрицы и выдает результат на экран;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 6]$  элементов с двумя пробелами между ними, а число отрицательных, положительных и нулевых элементов массива записываются в другой файл.

### Вариант № 15

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла, имя которого задает пользователь, двумерный массив целых чисел размерностью  $[3 \times 6]$  элементов и выводит его на экран;
- б) многократно, в зависимости от запроса, производит подсчет суммы элементов указанной строки и выдает результат на экран;
- в) при окончании работы сохраняет матрицу в файле, имя которого задает пользователь, до следующего сеанса работы.

### Вариант № 16

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив целых чисел размерностью  $[3 \times 6]$  элементов и выводит его на экран в виде матрицы. Элементы массива в файле могут быть расположены произвольно;
- б) многократно, в зависимости от запроса, производит подсчет суммы элементов указанного столбца и выдает результат на экран;

- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 6]$  элементов с двумя пробелами между ними, а результаты расчета суммы элементов указанных столбцов матрицы и их номера записываются в другой файл.

### Вариант № 17

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив вещественных чисел размерностью  $[4 \times 5]$  элементов и выводит его на экран в виде матрицы. При выводе все вещественные числа ограничены 4-мя знаками после запятой. Элементы массива в файле могут быть расположены произвольно;
- б) многократно производит в зависимости от запроса поиск *max* или *min* элемента указанной строки матрицы и результат выдает на экран;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[4 \times 5]$  элементов с двумя пробелами между ними, а результаты поиска *max* или *min* элемента указанных строк матрицы и их номера записываются в другой файл. В этом же файле сохраняется результат определения самого максимального и самого минимального из найденных элементов в указанных строках.

### Вариант № 18

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив вещественных чисел размерностью  $[4 \times 5]$  элементов и выводит его на экран в виде матрицы. При выводе все вещественные числа ограничены 4-мя знаками после запятой. Элементы массива в файле могут быть расположены произвольно;
- б) многократно производит в зависимости от запроса поиск *max* или *min* элемента указанного столбца матрицы и результат выдает на экран;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[4 \times 5]$  элементов с двумя пробелами между ними, а результаты поиска *max* или *min* элемента указанных столбцов матрицы и их номера записываются в другой файл. В этом же файле сохраняется результат определения самого максимального и самого минимального из найденных элементов в указанных столбцах.

### Вариант № 19

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив целых чисел размерностью  $[3 \times 4]$  элементов и выводит его на экран. Элементы массива в файле могут быть расположены произвольно;
- б) многократно производит изменение матрицы путем замены указанного элемента на другой, вводимый с клавиатуры, и выводит измененную матрицу на экран;
- в) по окончании работы сохраняет измененный массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 4]$  элементов с двумя пробелами между ними, а величины измененных (старых) элементов и значение их индексов  $[i,j]$  записываются в другой файл.

### Вариант № 20

Разработать блок-схему алгоритма программы и программу, которая:



- а) Читает из файла двумерный массив целых чисел размерностью  $[7 \times 2]$  элементов и выводит его на экран. Элементы массива в файле могут быть расположены произвольно;
- б) многократно, в зависимости от запроса, производит расчет среднего арифметического элементов указанной пользователем строки;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[7 \times 2]$  элементов с двумя пробелами между ними, а результаты расчета среднего арифметического элементов указанных строк матрицы и их номера записываются в другой файл.

### **Вариант № 21**

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный целочисленный массив размерностью  $[3 \times 7]$  элементов и выводит его на экран. Элементы массива в файле могут быть расположены произвольно;
- б) многократно, в зависимости от запроса, производит расчет среднего арифметического элементов указанного пользователем столбца;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 7]$  элементов с двумя пробелами между ними, а результаты расчета среднего арифметического элементов указанных столбцов матрицы и их номера – в другом файле.

### **Вариант № 22**

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный целочисленный массив размерностью  $[3 \times 7]$  элементов и выводит его на экран. Элементы массива в файле могут быть расположены произвольно;
- б) многократно, в зависимости от запроса, производит подсчет количества отрицательных, или положительных, или нулевых элементов матрицы и выдает результат на экран;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 7]$  элементов с двумя пробелами между ними, а результаты количества отрицательных, положительных и нулевых элементов массива – в другом файле.

### **Вариант № 23**

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла, имя которого задает программист, двумерный массив целочисленных чисел размерностью  $[4 \times 5]$  элементов и выводит его на экран;
- б) многократно, в зависимости от запроса, производит подсчет суммы элементов указанной строки и выдает результат на экран;
- в) по окончании работы сохраняет (переписывает) массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[4 \times 7]$  элементов с тремя пробелами между ними. Номера указанных строк, значение всех их элементов и сумма элементов каждой строки последовательно записывается в другой файл.

### **Вариант № 24**

Разработать блок-схему алгоритма программы и программу, которая:



- а) Читает из файла двумерный целочисленный массив размерностью  $[3 \times 6]$  элементов и выводит его на экран. Элементы массива в файле могут быть расположены произвольно;
- б) многократно, в зависимости от запроса, производит подсчет суммы элементов указанного столбца и выдает результат на экран;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[4 \times 6]$  элементов с тремя пробелами между ними, а номера указанных столбцов, значение всех их элементов и сумма элементов каждого столбца последовательно записывается в другой файл.

### Вариант № 25

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла двумерный массив вещественных чисел размерностью  $[2 \times 6]$  элементов и выводит его на экран в виде матрицы. При выводе все вещественные числа ограничены 4-мя знаками после запятой. Элементы массива в файле могут быть расположены произвольно;
- б) многократно производит в зависимости от запроса поиск *max* или *min* элемента указанной строки матрицы и результат выдает на экран;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[2 \times 6]$  элементов с двумя пробелами между ними, а результаты поиска *max* или *min* элемента указанных строк матрицы и их номера – в другом файле. В этом же файле сохраняется результат определения самого максимального и самого минимального из найденных элементов в указанных строках.

### Вариант № 26

- а) Читает из файла двумерный массив вещественных чисел размерностью  $[3 \times 6]$  элементов и выводит его на экран в виде матрицы. При выводе все вещественные числа ограничены 4-мя знаками после запятой. Элементы массива в файле могут быть расположены произвольно;
- б) многократно производит в зависимости от запроса поиск *max* или *min* элемента указанного столбца матрицы и результат выдает на экран;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 6]$  элементов с двумя пробелами между ними, а результаты поиска *max* или *min* элемента указанных столбцов матрицы и их номера – в другом файле. В этом же файле сохраняется результат определения самого максимального и самого минимального из найденных элементов в указанных столбцах.

### Вариант № 27

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла, имя которого задает программист, двумерный целочисленный массив размерностью  $[5 \times 6]$  элементов и выводит его на экран;
- б) многократно производит изменение матрицы путем замены указанного элемента на другой, вводимый с клавиатуры, и выводит измененную матрицу на экран;
- в) по окончании работы сохраняет последний измененный массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[2 \times 6]$  элементов с двумя пробелами между ними. В другом файле сохраняются значения и индексы  $[i,j]$  измененных элементов.

**Вариант № 28**

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла одномерный массив вещественных чисел размерностью [12] элементов и выводит его на экран. Вещественные числа при выводе должны содержать не более 4-х знаков после запятой;
- б) многократно, в зависимости от запроса, производит расчет среднего арифметического только четных, или только нечетных элементов массива;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[2 \times 6]$  элементов с двумя пробелами между ними. В другом файле сохраняются значения рассчитываемых элементов и результат их среднего арифметического.

**Вариант № 29**

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла, имя которого задает программист, двумерный целочисленный массив размерностью  $[2 \times 4]$  элементов и выводит его на экран. Данные в файле расположены произвольно;
- б) многократно, в зависимости от запроса, производит расчет среднего арифметического элементов указанного столбца;
- в) по окончании работы сохраняет массив в файле с именем, указанным программистом. Данные этого массива сохраняются в виде матрицы, размерностью  $[3 \times 7]$  элементов с двумя пробелами между ними, а результаты расчета среднего арифметического элементов указанных столбцов матрицы и их номера – в другом файле.

**Вариант № 30**

Разработать блок-схему алгоритма программы и программу, которая:

- а) Читает из файла, имя которого задает программист, одномерный массив целых чисел размерностью [14] элементов и выводит его на экран;
- б) многократно, в зависимости от запроса, производит подсчет количества отрицательных, или положительных, или нулевых элементов массива и выдает результат на экран;
- в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью  $[2 \times 7]$  элементов с двумя пробелами между ними. Количество отрицательных, положительных и нулевых элементов массива сохраняет в другом файле.

## ПРИЛОЖЕНИЕ 2 (структуры («хорошо») и классы («отлично»))

### Перечень примерных направлений для тем проектно-исследовательского характера

Предполагаются проектирование и разработка программ, имитирующих базы данных для учебной предметной области. База данных может быть представлена одним или несколькими текстовыми файлами с информацией в виде соответствующих записей. **Количество записей для тестового варианта не менее 10.** Программная реализация предполагает использование в программе одной Структуры или нескольких связанных Структур («хорошо»); или Базового (и дочерних) классов и их методов («отлично»).

Функциями обработки (подпрограммами) или методами классов могут быть функции обработки исходной информации по запросам пользователя. Реализовать не менее пяти запросов справочного или аналитического характера (статистика по количественному (сколько), или качественному (процент, условия выборки) анализу данных)

### Варианты условных баз данных (варианты курсовых проектов)

1. **«Автосалон».** Проектирование базы данных для нужд автосалона. БД содержит информацию об автомобилях (модель, год выпуска, цвет, технические характеристики, цену с вариантами комплектации и др.), о менеджерах, оформивших сделку (ФИО менеджера, эффективность (суммарное количество сделок и суммарный доход от продаж) и др.), о продажах (дата продажи и др.), о покупателях (ФИО, Паспортные данные, телефон, статус (постоянный/нет), % скидки и др.).
2. **«Видеопрокат».** Проектирование базы данных для нужд салона видеопроката. БД содержит информацию о видеокассетах (наименование, жанр, описание, залоговая стоимость, стоимость проката за сутки, дата выдачи, дата возврата и др.) и о клиентах (ФИО, Паспортные данные, телефон, статус (постоянный/нет), % скидки и др.).
3. **«Прокат»** (см. «Видеопрокат»). Проектирование базы данных для нужд различных фирм, деятельность которых связана с услугами по предоставлению клиентам некоего товара напрокат (автомобили, велосипеды и др.). Информационная насыщенность баз данных «Прокат» отличается от базы данных «Видеопрокат» только особенностью характеристик товара, предоставляемого напрокат клиентам.
4. **«Библиотека».** Проектирование базы данных для нужд библиотеки. БД содержит информацию о книгах (автор, наименование, жанр, цена, другие реквизиты, дата выдачи, дата возврата и др.) и о читателях (ФИО, паспорт, адрес, дата рождения, номер читательского билета, полученной книге и др.). Читатель может взять несколько книг.
5. **«Турагентство».** Проектирование базы данных для нужд туристической фирмы. БД содержит информацию о маршрутах (наименование, продолжительность, даты начала и окончания, цена путевки, описание маршрута, дополнительные услуги и др.) и о клиентах (ФИО, Паспортные данные, телефон, статус (постоянный/нет), количество путевок, % скидки и др.). Клиент может заказать несколько путевок по разным маршрутам.

6. **«Железнодорожные перевозки».** Проектирование базы данных для нужд железнодорожной кассы. БД содержит информацию о поездах (номер, тип (экспресс, скорый, пассажирский), номер вагона, тип вагона (общий, плацкарт, купе, спальный и т.п.) и др.), пунктах назначения (наименование, дата отправления/прибытия, время отправления, расстояние, стоимость проезда (доплата за срочность, доплата за тип вагона), пассажирах (ФИО, Паспортные данные, телефон, статус (постоянный/нет), % скидки, и др.).
7. **«Авиаперевозки».** Проектирование базы данных по авиаперевозкам, которая должна обеспечивать хранение данных о расписании авиарейсов (номер рейса, маршрут, пункты промежуточной посадки, дни полета, время отправления), хранение текущих сведений о наличии свободных мест на каждом рейсе и выдачу справок об этом, обеспечивать поиск ближайшего (по времени) рейса до заданного пункта (при наличии на рейсе свободных мест), оформление нужного пассажиру числа билетов на рейс (с соответствующим уменьшением числа свободных мест на этот рейс), формирование посадочной ведомости пассажиров на рейс.
8. **«Автобусные перевозки».** Проектирование базы данных для нужд пассажирских автовокзалов. БД содержит сведения о междугородних автобусных маршрутах: пунктах назначения и прибытия, расстоянии, времени в пути, стоимости проезда, промежуточных пунктах, расписании движения всех рейсов за день по маршруту и т. д., сведения о наличии свободных мест на рейсах и т. п. Система должна обеспечивать поиск и выдачу сведений по разным вопросам.
9. **«Преподаватель».** Проектирование базы данных для нужд кадрового отдела ВУЗа. БД содержит информацию о преподавателях и их учебной нагрузке (ФИО, телефон, стаж, научная степень, звание, кафедра, занимаемая должность, ведомая дисциплина, нагрузка, количество публикаций по категориям и др.), должностях (наименование категорий должностей по кафедрам и их количество), предметах (наименование, количество часов и др.). Преподаватель может вести занятия по нескольким предметам.
10. **«Аптечный склад».** Проектирование базы данных для нужд аптечного склада, информация которой будет использоваться для учета продаж аптекам лекарственных препаратов. База данных содержит информацию о лекарствах (код лекарства, название лекарства, производитель, цена(руб.) и др.), аптеках (номер аптеки, название аптеки, адрес аптеки, номер телефона и др.), заявок (номер заявки, дата составления заявки, номер аптеки, дата выполнения заявки и др.), закупок лекарственных препаратов (номер заявки, код лекарства, количество (шт.))
11. **«Поликлиника».** Проектирование базы данных для нужд районной поликлиники, информация которой позволит хранить сведения о заболеваниях пациентов, частоте их обращения, загрузке врачей, выручке от оказания медицинских услуги др. БД содержит информацию о врачах (Ф.И.О. врача, специальность, категория и др.), пациентах (номер медкарты, Ф.И.О. пациента, дата рождения, адрес, пол, скидка на обслуживание (%) и др.), приеме пациентов (номер талона на прием к врачу, дата визита, цель посещения, стоимость визита (руб.) и др.), диагнозе (код диагноза, наименование диагноза и др.).

12. **«Футбольный чемпионат».** Проектирование базы данных по справочной информации чемпионата премьер лиги по футболу. В базе данных футбольной лиги хранятся сведения о футбольных клубах, а также данные о графике проведения игр, названиях и вместимости стадионов, расстояниях поездки (для каждой команды) до мест их расположения, датах проведенных встреч, местах их проведения и названиях стадиона, названиях команд и списках, заявленных и участвовавших в матче игроков команд, счете матча, авторах голов и т. д. Система должна обеспечивать поиск и выдачу сведений по разным вопросам.
13. **«Доставка пиццы».** Проектирование базы данных для нужд интернет-магазина по продаже пиццы. БД содержит информацию о клиентах и их заказах (Фамилия, имя, отчество, адрес, телефон, название пиццы, количество заказа, цене заказа, дата и время и др.), об ассортименте пиццы (наименование, описание, цена, скидка), о заказах (наименование, количество, дата и время заказа, дата и время доставки)
14. **«Торговые операции».** Проектирование базы данных для нужд компаний, совершающих друг с другом торговые сделки. БД содержит информацию о фирмах-поставщиках и фирмах-клиентов (название фирмы, юридический адрес, фамилия, имя, отчество руководителя, контактный телефон) и о товаре-сделке (название товара, единицы измерения, количество, цена, скидка, дата купли/продажи, описание и др.). Предполагается взаимодействие нескольких (более двух) фирм.
15. **«Ремонт бытовой техники».** Проектирование базы данных для нужд ремонтной мастерской, информация которой будет использоваться для учета услуг по ремонту и гарантийному обслуживанию товаров, анализа сроков исполнения заказов, видов неисправностей и др. БД содержит информацию о товарах (код товара, наименование товара, фирма, модель, технические характеристики, гарантийный срок, изображение и др.), сотрудниках (Ф.И.О. сотрудника, должность), заказах (код заказа, Ф.И.О. клиента, код товара, гарантия (Да/Нет), дата поступления заказа, телефон клиента), исполнения заказов (код заказа, исполнитель, вид ремонта, стоимость ремонта, дата исполнения заказа, сообщение клиенту (Да/Нет), дата получения товара, сумма оплаты услуг (руб.) и др.).
16. **«Интернет-продажи».** Проектирование базы данных для аналитических центров интернет-продаж, информация которой будет использоваться для анализа спроса потребителей на конкретные модели товаров разных производителей, динамики реализации товаров в интернет-магазинах за определенные интервалы времени, сравнения условий доставки товаров в разных магазинах и др. БД содержит информацию о интернет-магазинах (код магазина, электронный адрес, доставка (Да/Нет)), товарах (код товара, название товара, фирма, модель, технические характеристики, цена (руб.), гарантийный срок, изображение), интернет-заказах (код заказа, код магазина, код товара, дата заказа, время заказа, количество, Ф.И.О. клиента, контактный телефон, подтверждение заказа (Да/Нет) и др.), доставках (код заказа, дата доставки, время доставки, адрес доставки, Ф.И.О. клиента, Ф.И.О. курьера и др.). Товар может продаваться в нескольких интернет-магазинах.
17. **«Интернет-клиент».** Проектирование базы данных для коммерческого отдела компании, продающей различные товары через Интернет. Задачей является отслеживание финансовой составляющей работы компании (покупок клиентов) с целью

привлечения и удержания клиентов путем предоставления ему дополнительных скидок и льгот. Работа компании организована следующим образом: на Интернет-сайте компании представлены (выставлены на продажу) некоторые товары. Каждый из них имеет некоторое название, цену и единицу измерения (штуки, килограммы, литры). Коммерческим отделом собираются данные о клиентах. Определяющее значение имеют стандартные анкетные данные, а также телефон и адрес электронной почты для связи. В случае приобретения товаров на сумму, например, свыше 5000 р. клиент переходит в категорию «постоянных клиентов» и получает скидку на каждую покупку в размере 2%. По каждому факту продажи фиксируются клиенты, товары, количество, дата продажи, дата доставки и др. Исходя из суммарной стоимости заказа, компания предоставляет дополнительные скидки клиенту и оповещает его об этом.

18. **«Обмен валюты».** Проектирование базы данных для пункта обмена валют, которая должна обеспечивать хранение, поиск, анализ данных о сделках, совершенных в пункте обмена, и др. БД содержит информацию о клиентах (Ф.И.О. клиента, номер паспорта), сделках (код проданной валюты, код купленной валюты, номер кассира, номер клиента, дата сделки, время сделки, сумма проданной валюты, сумма купленной валюты) валютах (код проданной валюты, код купленной валюты, название валюты, курс продажи, курс покупки), кассирах, совершивших сделку. Клиент может совершать несколько сделок.
19. **«Банковские вклады».** Необходимо спроектировать базу данных для банка, информация которой будет использоваться для анализа работы с клиентами по вкладам. БД содержит информацию о клиентах, которые помещают денежные средства на вклады (код клиента, Ф.И.О. клиента, номер паспорта, адрес, телефон), вкладах (код вклада; наименование вклада; срок хранения (месяцев); ставка, % годовых), счетах клиентов (№ счета, код клиента, код вклада, дата открытия счета, дата закрытия счета, сумма вложенная (руб.)). Клиентам банка предлагается размещать денежные средства на срочные банковские вклады различных видов, при этом клиент банка может помещать свои средства на несколько счетов.
20. **«Телепрограмма».** Проектирование справочной базы данных по телевизионным программам различных каналов на неделю. БД должна обеспечивать хранение программы телепередач нескольких телекомпаний на неделю (по дням, времени показа) с указанием категорий телепередач: новости, спорт (по видам), художественные фильмы (по жанрам), сериалы и т.д., обеспечивать формирование совокупной программы просмотра по определенному запросу (вкусу) с указанием временных «накладок» телепередач, иметь возможность формировать список «любимых» передач (сериалов, фильмов, телепередач) для обязательного включения в ежедневный (еженедельный и т. д.) просмотр. Для художественных фильмов и сериалов желательно иметь возможность хранения дополнительных сведений: название, страна, год, режиссер, ведущие актеры, краткое содержание.
21. **«Справочник абитуриента».** Проектирование справочной базы данных для абитуриентов по техническим специальностям некоторых вузов. БД должна обеспечивать хранение данных о вузах: наименование, регион, город, адрес, перечень специальностей и форм обучения по ним, конкурсы прошлого года по каждой из спе-



циальностей (по дневной, вечерней, заочной и всем ускоренным формам обучения), размер оплаты при договорном обучении по специальности (дифференцированно по разным формам обучения), а также обеспечить формирование выборок сведений по разным критериям, связанных с выбираемой специальностью.

22. **«Расписание занятий».** Проектирование базы данных для составления фрагмента расписания аудиторных занятий в вузе, составляемое на определенный период (семестр) для одного курса. БД включает сведения о названиях предметов (учебных дисциплин), видах аудиторных занятий (лекция, лабораторная работа, практическое занятие), фамилиях преподавателя (преподавателей), обозначениях аудиторий, днях недели и номерах «пар» (сдвоенных академических часов занятий), показателе четности недели проведения занятия («числитель/знаменатель»), степени «полноты» группы на занятии (группа или подгруппы), обозначениях учебных групп, номерах курса, обозначениях институтов (факультетов) и др.
23. **«Учет академической успеваемости».** Проектирование базы данных для учета успеваемости студентов и ее анализу: подсчетом средних баллов, качества знаний (процент успевающих на 4 и 5), процента успеваемости и неуспеваемости для групп, специальностей, курсов, институтов (факультетов) в среднем и по отдельным учебным дисциплинам, в частности. Сведения об академической успеваемости студентов содержатся в их зачетных книжках и в экзаменационных (зачетных) ведомостях: Ф.И.О. студента, номер зачетной книжки, год поступления, институт (факультет), специальности, учебная группа, семестр, название предмета (учебной дисциплины), вид отчетности (зачет, экзамен, КП, КР), сама оценка («зачтено», «не зачтено», «отлично», «хорошо», «удовлетворительно», «неудовлетворительно»), дата, фамилия преподавателя, количество часов и т. д. База данных также должна формировать заполненные экзаменационные (зачетных) ведомости для любой группы по любому предмету, семестру и т. д., формировать и обеспечивать печать итоговых ведомостей по результатам семестра с аналитическими результатами.
24. **«Театральная касса».** Проектирование базы данных по бронированию театральных билетов. БД содержит информацию о репертуаре (название спектакля, жанр, дата проведения, продолжительность, театр (сцена) и др.), театрах (наименование, адрес), наличие мест (на указанный спектакль или указанную дату), дифференцированных ценах на спектакль (в зависимости от театра и места (партер, ряд, балкон, бельэтаж и т.д.)), информацию о брони (номер, клиент, количество билетов, места). Клиент имеет право выбора мест. В случае бронирования указанных мест клиентом, данные места исключаются из списка доступных.
25. **«Гостиничный сервис».** Проектирование базы данных по бронированию гостиниц. БД содержит информацию о гостиницах (название, категория гостиницы, тип номера (люкс, 1-местный, 2-х местный и т.п.), суточная цена), номерах (тип номера, количество, наличие мест на текущую дату или дату запроса, и др.), клиентах (ФИО, дата заезда, дата выезда, количество мест, тип номера, дополнительных услугах, цене, и др.), дополнительных гостиничных услугах (вид услуги (все включено, завтрак в номер, вызов такси и др.), цена). Клиент имеет право выбрать гостиницу, выбрать или заказать номер в ней на указанную дату и срок. В случае бро-



нирования указанного номера на указанный срок, данный номер исключаются из списка доступных в выбранной клиентом гостинице.

26. **«Оператор сотовой связи».** Проектирование базы данных для анализа клиентской базы (или - клиентских баз) оператора (или - операторов) сотовой связи. БД содержит информацию об абоненте (№ п/п, фамилия, имя, отчество, абонентский федеральный номер, тарифный план, дата подключения, серия, номер паспорта, дата рождения, место жительства, дополнительные услуги разговоры, наименование услуги на разговоры, дополнительные услуги SMS, наименование услуги на SMS, дополнительные услуги MMS, наименование услуги на MMS, роуминг и др.), тарифном плане (тарифный план, абонентская плата, стоимость минуты с абонентами БВК, стоимость минуты с другими операторами, стоимость SMS, стоимость MMS, дополнительная информация), дополнительные услуги разговоры (название, стоимость подключения, абонентская плата, стоимость 1 минуты), дополнительные услуги SMS (название, стоимость подключения, абонентская плата, стоимость 1 SMS), дополнительные услуги MMS (название, стоимость подключения, абонентская плата, стоимость 1 MMS).