

## ЛАБОРАТОРНАЯ РАБОТА №1.

НАЗВАНИЕ РАБОТЫ: Создание классов.

ЦЕЛЬ РАБОТЫ: Изучение основных возможностей языка C++ создания и использования классов.

### ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:

1. Согласовать с преподавателем задание.
2. Осуществить проектирование соответствующего класса. Включить в интерфейс класса, помимо функций, указанных в задании, следующие функции:
  - все типы конструкторов (если в дополнительных указаниях к группе заданий не указано иное). Конструкторы обязательно должны проверять корректность инициализации данных класса.
  - функции доступа и инициализации;
  - функцию сравнения объектов на равенство (если в задании не указано иное);
  - функции ввода с консоли и вывода на консоль.
3. Задания разбиты на разделы по уровню сложности. Некоторые разделы могут сопровождаться специфическими дополнительными указаниями к выполнению заданий именно этой группы.
4. Реализовать полученный проект средствами языка C++. При реализации конструкторов и функций класса, меняющих состояние объекта (значения его данных), следить за тем, чтобы объект всегда находился в непротиворечивом состоянии.
5. В функции main() создать объекты соответствующего типа и осуществить демонстрацию функциональности разработанного класса.
6. Подготовить отчет по лабораторной работе.

### УКАЗАНИЯ К НАПИСАНИЮ ПРОГРАММНОГО ПРИЛОЖЕНИЯ:

1. Не использовать избыточные параметры в функциях класса:

Не правильно:

```
Drob Drob::Add(Drob d1, Drob d2)
{
    // Реализация функции
}
```

Правильно:

```
Drob Drob::Add( Drob d2)
{
    // Реализация функции
}
```

2. Избегать ввода/вывода в функциях, для этого не предназначенных.

Не правильно:

```
void Drob::Equal(Drob d) const
{
    if ((double(ch) / d.ch) == (double(zn) / d.zn))
        cout << "Дробь равна";
}
```

```

        else
            cout << "Дроби не равны";
    };

```

**Правильно:**

```

bool Drob::Equal(Drob d) const
{
    if ((double(ch) / d.ch) == (double(zn) / d.zn))
        return true;
    else
        return false;
};

```

**3. Не передавать в функцию класса вместо объекта совокупность значений его свойств:**

**Не правильно:**

```

Drob Drob::Add(int ch2, int zn2)
{
    // Реализация функции
}

```

**Правильно:**

```

Drob Drob::Add( Drob d2)
{
    // Реализация функции
}

```

**4. Не включать в класс ненужные дополнительные данные:**

**Не правильно:**

```

class Drob
{
private:
    int ch;
    int zn;
    int ch2;
    int zn2;
public:
    ...
    void Add()
    {
        ch+=ch2;
        zn+=zn2;
    }
    ...
};

```

**Правильно:**

```

class Drob
{
private:
    int ch;
    int zn;
public:
    ...
    Drob Add(Drob d2)
    {
        Drob d;
        d.zn = zn * d2.zn;
        d.ch = zn*d2.ch + d2.zn*ch;
        return d;
    }
}

```

}; ...

5. Располагать объявление и реализацию классов в том же файле, что и функцию main (не использовать заголовочных файлов и отдельных файлов с реализацией).

6. Активно использовать константные данные и функции класса. Избегать использования статических данных и функций класса; глобальных по отношению к классу функций, в т.ч. дружественных.

7. При реализации функции main() избегать создания консольного интерфейса: функция main должна представлять собой последовательность вызовов функций класса с возможным выводом получаемых значений на экран.

8. Не допускать использования функций, перегружающих операции.

9. Допускается, при необходимости, использование вложенных классов.

10. Имена классов и функций классов начинать с прописной буквы (за исключением функций, названия которых начинаются с get или set).

#### КРИТЕРИИ ОЦЕНИВАНИЯ:

##### **Факторы, понижающие оценку:**

- Несоответствие работы заданию.
- Игнорирование указаний к написанию программного приложения.
- Некорректная работа функций класса.
- Незнание теории.
- Нарушение сроков согласования задания и защиты работы.

#### СОДЕРЖАНИЕ ОТЧЕТА:

1. Название работы.
2. Цель работы.
3. Содержание задания, включающего в себя и общую и оригинальную часть.
4. Интерфейс разработанного класса - прототипы открытых функций и данные.
5. Исходник программы, содержащий реализацию класса.
6. Исходник функции main().
7. Скриншоты результатов работы программы.
8. Выводы по работе.

## СПИСОК ЗАДАНИЙ:

### **Простые (8 баллов):**

#### **1. «Числовая дробь».**

Данные класса: числитель и знаменатель дроби.

Функции класса: сложение, вычитание, умножение, деление, сравнение двух дробей на равенство, умножение и сложение с дробью целого числа.

#### **2. «Числовая дробь».**

Данные класса: числитель и знаменатель дроби.

Функции класса: сложение, вычитание, умножение, деление, сравнение двух дробей на неравенство, умножение и вычитание из дроби целого числа.

#### **3. «Числовая дробь».**

Данные класса: числитель и знаменатель дроби.

Функции класса: сложение, вычитание, умножение, деление, сравнение двух дробей на больше или равно, деление и сложение с дробью целого числа.

#### **4. «Числовая дробь».**

Данные класса: числитель и знаменатель дроби.

Функции класса: сложение, вычитание, умножение, деление, сравнение двух дробей на меньше или равно, деление и вычитание из дроби целого числа.

#### **5. «Правильная числовая дробь».**

Данные класса: числитель и знаменатель правильной числовой дроби, целая часть.

Функции класса: сложение, вычитание, умножение, деление двух дробей, умножение и сложение с правильной дробью целого числа, вычисление НОД числителя и знаменателя, упрощение дроби.

#### **6. «Комплексное число».**

Данные класса: действительная и мнимая части комплексного числа.

Функции класса: сложение, вычитание, умножение, сравнение двух комплексных чисел, умножение и сложение с комплексным числом вещественного числа, вычисление модуля и аргумента комплексного числа.

#### **7. «Время».**

Данные класса: часы, минуты, секунды.

Функции класса: нахождение разности двух моментов времени в секундах, прибавление ко времени одной секунды, сравнение двух времен.

### **8. «Дата».**

Данные класса: число, месяц, год.

Функции класса: прибавление к дате одного дня, сравнение двух дат, проверка, относится ли текущая дата к високосному году.

### **9. «Денежная сумма».**

Данные класса: размер суммы, код валюты, курс по отношению к доллару.

Функции класса: сравнение двух сумм, сложение двух сумм, вычисление значения суммы в долларах.

### **10. «Сочетание».**

Данные класса:  $n$  и  $m$ .

Функции класса: сравнение двух сочетаний, вычисление значения числа сочетаний.

### **11. «Двучлен».**

Данные класса: коэффициенты при переменных в нулевой и первой степенях.

Функции класса: сложение, вычитание, сравнение двух двучленов, умножение и сложение с двучленом вещественного числа, вычисление значения двучлена для заданного значения переменной, решение соответствующего уравнения.

### **12. «Трехчлен».**

Данные класса: коэффициенты при переменных в нулевой, первой и второй степенях.

Функции класса: сложение, вычитание, сравнение двух трехчленов, умножение и сложение с трехчленом вещественного числа, вычисление значения трехчлена для заданного значения переменной.

### **13. «Точка в двумерном пространстве».**

Данные класса: координаты точки.

Функции класса: сравнение двух точек, проверка, в каком квадранте находится точка, вычисление расстояния между точками.

### **14. «Вектор в двумерном пространстве».**

Данные класса: координаты вектора по оси абсцисс и ординат.

Функции класса: сложение, вычитание, сравнение двух векторов, умножение вектора на число, вычисление значения модуля вектора, вычисление скалярного произведения двух векторов.

### **15. «Вектор в трехмерном пространстве».**

Данные класса: координаты вектора по оси абсцисс, ординат и аппликат.

Функции класса: сложение, вычитание, сравнение двух векторов, умножение вектора на число, вычисление значения модуля вектора, вычисление скалярного произведения двух векторов.

#### **16. «Отрезок в двумерном пространстве».**

Данные класса: координаты начала и конца отрезка по оси абсцисс и ординат.

Функции класса: сравнение двух отрезков, проверка, пересекаются ли отрезки в пространстве, параллельны ли прямые, на которых лежат два отрезка.

#### **17. «Окружность в двумерном пространстве».**

Данные класса: координаты по оси абсцисс и ординат центра, радиус.

Функции класса: сравнение двух окружностей, вычисление площади и периметра окружности, проверка на пересечение кругов, соответствующим двум окружностям, проверка двух окружностей на концентричность.

#### **18. «Сектор круга».**

Данные класса: угол и радиус.

Функции класса: сравнение двух секторов, вычисление площади сектора, вычисление длины дуги сектора.

#### **19. «Кольцо в двумерном пространстве».**

Данные класса: координаты по оси абсцисс и ординат центра, меньший радиус, больший радиус.

Функции класса: сравнение двух колец, вычисление площади кольца, вычисление периметра кольца.

#### **20. «Прямоугольник в двумерном пространстве».**

Данные класса: координаты по оси абсцисс и ординат левой нижней вершины, длина и ширина.

Функции класса: сравнение двух прямоугольников, вычисление площади и периметра прямоугольника, проверка, попадает ли точка внутрь прямоугольника.

#### **21. «Треугольник в двумерном пространстве».**

Данные класса: длины сторон треугольника.

Функции класса: сравнение двух треугольников, вычисление площади и периметра треугольника, проверка, является ли треугольник прямоугольным, равнобедренным, равносторонним.

#### **22. «Треугольник в двумерном пространстве».**

Данные класса: координаты вершин треугольника.

Функции класса: сравнение двух треугольников, вычисление площади и периметра треугольника, проверка, попадает ли точка внутрь треугольника.

#### **23. «Матрица 2x2 вещественных чисел».**

Данные класса: статический массив элементов матрицы.

Функции класса: сложение и умножение матриц, умножение матрицы на число, сравнение двух матриц, вычисление определителя матрицы, транспонирование матрицы.

#### **24. «Матрица 3x3 вещественных чисел».**

Данные класса: статический массив элементов матрицы.

Функции класса: сложение и умножение матриц, умножение матрицы на число, сравнение двух матриц, вычисление определителя матрицы, транспонирование матрицы.

### **25. «Десятичный счетчик».**

Данные класса: минимальное значение, максимальное значение, текущее значение, шаг приращения.

Функции класса: сравнение двух счетчиков, увеличение текущего значения на шаг приращения, уменьшение текущего значения на шаг приращения, получение текущего значения.

### **26. «Диапазон на числовой прямой».**

Данные класса: левая граница диапазона, правая граница диапазона.

Функции класса: сравнение двух диапазонов, нахождение длины диапазона, проверка, попадает ли некоторое число в диапазон, проверка, пересекаются ли диапазоны.

### **27. «Геометрическая прогрессия».**

Данные класса: начальный элемент, коэффициент прогрессии.

Функции класса: сравнение двух прогрессий, нахождение суммы первых  $n$  элементов прогрессии, получение элемента прогрессии с указанным номером.

### **28. «Арифметическая прогрессия».**

Данные класса: начальный элемент, коэффициент прогрессии.

Функции класса: сравнение двух прогрессий, нахождение суммы первых  $n$  элементов прогрессии, получение элемента прогрессии с указанным номером.

## **Задания, предполагающие сущности с динамически размещаемым массивом элементов (9 баллов).**

**Указание к выполнению заданий этой группы:** обязательно включить в класс конструктор, использующий в качестве одного из параметров одномерный или двумерный массив чисел; конструктор преобразования реализовывать не нужно. Обязательно включить в класс деструктор. Для всех вариантов включить в класс функцию сравнения объектов на равенство.

### **29. «Произвольный многоугольник в двумерном пространстве».**

Данные класса: количество вершин, указатель на динамически размещаемый массив координат вершин прямоугольника.

Функции класса: определение, является ли многоугольник той или иной фигурой на плоскости, вычисление площади многоугольника.

### **30. «Вектор в многомерном пространстве».**

Данные класса: размерность вектора, указатель на динамически размещаемый массив координат вектора.

Функции класса: сложение, вычитание двух векторов, умножение вектора на число, вычисление значения модуля вектора.

### **31. «Полином от одной переменной».**

Данные класса: размерность полинома, указатель на динамически размещаемый массив коэффициентов при переменных.

Функции класса: сложение, вычитание двух полиномов, умножение и сложение с полиномом числа, вычисление значения полинома для заданного значения переменной.

### **32. «Полином от одной переменной» (перемножение полиномов).**

Данные класса: размерность полинома, указатель на динамически размещаемый массив коэффициентов при переменных.

Функции класса: сложение, вычитание двух полиномов, умножение и сложение с полиномом числа, вычисление значения полинома для заданного значения переменной, **перемножение полиномов.**

### **33. «Матрица произвольной размерности» (вычисление определителя).**

Данные класса: количество строк, количество столбцов, указатель на динамически размещаемый массив элементов матрицы.

Функции класса: сложение и умножение двух матриц, умножение матрицы на число, вычисление определителя матрицы.

### **34. «Матрица произвольной размерности» (транспонирование матрицы).**

Данные класса: количество строк, количество столбцов, указатель на динамически размещаемый массив элементов матрицы.

Функции класса: сложение и умножение двух матриц, умножение матрицы на число, транспонирование матрицы.

## **Задания, предполагающие сущности с динамически размещаемым списком своих элементов (10 баллов).**

**Указания к выполнению заданий этой группы:** обязательно включить в класс конструктор, использующий в качестве одного из параметров одномерный или двумерный массив чисел, конструктор преобразования реализовывать не нужно. Обязательно включить в класс деструктор. Для всех вариантов включить в класс функцию сравнения объектов на равенство. Рекомендуется реализовать вспомогательную функцию доступа к значениям, содержащимся в списке, по их координатам и использовать эту функцию для реализации прочих функций класса.

### **35. «Разреженный полином от одной переменной».**

Данные класса: размерность полинома, указатель на динамический список ненулевых коэффициентов при переменных.

Функции класса: сложение, вычитание двух полиномов, умножение и сложение с полиномом числа, вычисление значения полинома для заданного значения переменной.



**36. «Разреженный полином от одной переменной» (перемножение полиномов).**

Данные класса: размерность полинома, указатель на динамический список ненулевых коэффициентов при переменных.

Функции класса: сложение, вычитание двух полиномов, умножение и сложение с полиномом числа, вычисление значения полинома для заданного значения переменной, перемножение полиномов.

**37. «Разреженная матрица произвольной размерности» (вычисление определителя).**

Данные класса: количество строк, количество столбцов, указатель на динамический список ненулевых элементов матрицы.

Функции класса: сложение и умножение двух матриц, умножение матрицы на число, вычисление определителя матрицы.

**38. «Разреженная матрица произвольной размерности» (транспонирование матрицы).**

Данные класса: количество строк, количество столбцов, указатель на динамический список ненулевых элементов матрицы.

Функции класса: сложение и умножение двух матриц, умножение матрицы на число, транспонирование матрицы.

**39. «Стек чисел».**

Данные класса: указатель на головку стека в динамическом списке элементов стека.

Функции класса: считывание без извлечения элемента стека, считывание с извлечением элемента стека, запись элемента в стек.

**40. «Очередь чисел».**

Данные класса: указатель на начало очереди в динамическом списке элементов очереди, указатель на конец очереди в динамическом списке элементов очереди.

Функции класса: считывание без извлечения элемента очереди, считывание с извлечением элемента очереди, запись элемента в очередь.

**41. «Кольцо чисел».**

Данные класса: указатель на текущий элемент кольца в динамическом списке элементов кольца.

Функции класса: считывание без извлечения текущего элемента кольца, считывание с извлечением текущего элемента кольца, перемещение указателя по и против часовой стрелки, запись элемента в кольцо.

**42. «Разреженный вектор в многомерном пространстве».**

Данные класса: размерность вектора, указатель на динамический список ненулевых координат вектора.

Функции класса: сложение, вычитание двух векторов, умножение вектора на число, вычисление значения модуля вектора.