

Лабораторная работа 14. Введение в ООП**ВАРИАНТЫ ЗАДАНИЙ**

1. Разработать класс CLine (Прямая, заданная общим уравнением вида $Ax+By+C=0$)

Класс	Элементы данных	Интерфейс
CLine	A, B, C	Конструкторы, операции =, && (возвращает 1, если прямые пересекаются, иначе – 0), * (определяет точку пересечения двух прямых), ^ (возвращает угол наклона прямой к оси X), ==, !=, <<, >>.

Написать тексты h-файла и сpp-файла для класса CLine. Разработать и отладить программу создания и использования объектов класса.

2. Разработать класс CDate:

Класс	Элементы данных	Интерфейс
CDate	d (день), m (месяц), y (год)	Конструкторы, методы SetDate, операции =, + (прибавляет к дате некоторое количество дней), - (возвращает разность дат в днях), ++ (перевести дату на 1 день вперед), <, >, <<, >>.

Написать тексты h-файла и сpp-файла для класса CDate. Разработать и отладить программу создания и использования объектов класса.

3. Разработать класс Fraction (Дробь):

Класс	Элементы данных	Интерфейс
Fraction	a (числитель), b (знаменатель)	Конструкторы, операции =, +, -, ==, !=, ! (проверить конечность дроби, возвращает true/false), <<, >>.

Написать тексты h-файла и сpp-файла для класса Fraction. Разработать и отладить программу создания и использования объектов класса.

4. Написать тексты h-файлов и сpp-файлов для класса Circle. Описание класса:

Класс	Элементы данных	Интерфейс
Circle	x, y, R	Конструкторы, функции move, square, операции =, <, > (сравнение по площади), + (увеличение радиуса), <<, >>

Разработать и отладить программу с примерами создания и использования объектов класса Circle.

5. Написать тексты h-файлов и сpp-файлов для класса Rectangle (прямоугольник).

Описание класса:

Класс	Элементы данных	Интерфейс
Rectangle	x1, y1, dx, dy	Конструкторы, функции move, resize, операции =, && (пересечение прямоугольников), (получение наименьшего прямоугольника, содержащего два заданных прямоугольника), !(возвращает площадь прямоугольника), <<, >>

Разработать и отладить программу с примерами создания и использования объектов класса Rectangle.

6. Разработать класс Complex:

Класс	Элементы данных	Интерфейс
Complex	x, y	Конструкторы, операции =, +, -, ==, !=, *, - (унарный минус, что соответствует умножению на -1), <<, >>.

Написать тексты h-файла и сpp-файла для класса Complex. Разработать и отладить программу создания и использования объектов класса.

7. Написать тексты h-файлов и сpp-файлов для класса Point (прямоугольник).

Описание класса:

Класс	Элементы данных	Интерфейс
Point	x1, y1, dx, dy	Конструкторы, позволяющие создать экземпляр класса: с нулевыми координатами; с заданными координатами, функции move (перемещение точки), distance (вычисление расстояния до начала координат), операции = (присваивание), >, < (сравнение расстояния до начала координат), <<, >>

Разработать и отладить программу с примерами создания и использования объектов класса Point.

8. Написать тексты h-файлов и сpp-файлов для класса List (однонаправленный список). Описание класса:

Класс	Элементы данных*	Интерфейс
Структура Node	Значение, указатель на следующий элемент	Конструктор
List	Корень списка,	Конструкторы, методы для добавления узла в конец

	Количество узлов списка	списка, вставки узла после указанного узла, удаления узла, получения количества узлов, проверки, пуст ли список, операции =, + (получения одного списка из двух), ==, !=, <<, >>.
--	-------------------------	---

Разработать и отладить программу с примерами создания и использования объектов класса `List`.

* Один из вариантов, по согласованию с преподавателем можно реализовать по-другому.

9. Написать тексты h-файлов и сpp-файлов для класса `List` (двунаправленный список). Описание класса:

Класс	Элементы данных*	Интерфейс
Структура <code>Node</code>	Значение, указатель на следующий элемент, указатель на предыдущий	Конструктор
<code>List</code>	Корень списка, Количество узлов списка	Конструкторы, методы для добавления узла в конец списка, вставки узла после указанного узла, удаления узла, удаления последнего узла, получения количества узлов, проверки, пуст ли список, операции =, + (получения одного списка из двух), ==, !=, <<, >>.

Разработать и отладить программу с примерами создания и использования объектов класса `List`.

* Один из вариантов, по согласованию с преподавателем можно реализовать по-другому.

10. Разработать класс `Line` (Прямая, заданная уравнением вида $y=kx+b$)

Класс	Элементы данных	Интерфейс
<code>Line</code>	k, b	Конструкторы, метод <code>IsOnline(x,y)</code> – определяет, лежит ли точка с координатами (x,y) на прямой; операции =, ! (отражает прямую относительно оси X), метод <code>Intersection</code> (определяет точку пересечения двух прямых), операции ==, !=, <<, >>.

Написать тексты h-файла и сpp-файла для класса `Line`. Разработать и отладить программу создания и использования объектов класса.

11. Разработать класс `CLine` (Прямая, заданная общим уравнением вида $Ax+By+C=0$)

Класс	Элементы данных	Интерфейс
-------	-----------------	-----------

CLine	A, B, C	Конструкторы, операции =, ! (отражает прямую относительно оси Y), * (определяет точку пересечения двух прямых), ^ (возвращает угол наклона прямой к оси X), ==, !=, <<, >>.
-------	---------	---

Написать тексты h-файла и сpp-файла для класса CLine. Разработать и отладить программу создания и использования объектов класса.

12. Разработать класс Fraction (Дробь):

Класс	Элементы данных	Интерфейс
Fraction	p (числитель), q (знаменатель)	Конструкторы, операции =, +=, -=, <, >, ! (Период дроби (результат – строка)), <<, >>.

Написать тексты h-файла и сpp-файла для класса Fraction. Разработать и отладить программу создания и использования объектов класса. Добавить в программу возможность создания и сортировки (упорядочения) массива объектов.

13. Написать тексты h-файлов и сpp-файлов для класса Rectangle (прямоугольник). Описание класса:

Класс	Элементы данных	Интерфейс
Rectangle	x1, y1, x2, y2 (координаты левого нижнего и правого верхнего углов)	Конструкторы, функции move, операции: =, == (равно), != (не равно), ! (возвращает периметр прямоугольника), <, > (сравнение по площади), <<, >>

Разработать и отладить программу с примерами создания и использования объектов класса Rectangle.

14. Написать тексты h-файлов и сpp-файлов для класса Rectangle (прямоугольник). Описание класса:

Класс	Элементы данных	Интерфейс
Rectangle	x, y, dx, dy	Конструкторы, функции square, move, IsInRect(x,y) – определяет, лежит ли точка с координатами (x,y) внутри прямоугольника; операции: =, += (увеличить высоту и ширину), ! (отразить прямоугольник относительно оси x), <<, >>

Разработать и отладить программу с примерами создания и использования объектов класса Rectangle.

15. Написать тексты h-файлов и сpp-файлов для класса Treugolnik (треугольник). Описание класса:

Класс	Элементы данных	Интерфейс
-------	-----------------	-----------

Treugolnik	x1, y1, x2, y2, x3, y3	Конструкторы, функции move, square, операции =, <, > (сравнение площади), *= (изменить пропорции в некоторое число раз) , <<, >>
------------	---------------------------	--

Разработать и отладить программу с примерами создания и использования объектов класса Treugolnik.

16. Разработать класс CDate:

Класс	Элементы данных	Интерфейс
CDate	d (день), m (месяц), y (год)	Конструкторы, операции =, += (прибавляет к дате некоторое количество дней), -= (возвращает разность дат в днях), ++, -- (перевести дату на 1 день вперед/назад), <, >, <<, >>.

Написать тексты h-файла и сpp-файла для класса CDate. Разработать и отладить программу создания и использования объектов класса.

17. Написать тексты h-файлов и сpp-файлов для класса Circle. Описание класса:

Класс	Элементы данных	Интерфейс
Circle	x, y, R	Конструкторы, функции IsInCircle(x,y) – определяет, лежит ли точка с координатами (x,y) внутри круга, операции =, <, > (сравнение по площади), + (увеличение радиуса) , <<, >>

Разработать и отладить программу с примерами создания и использования объектов класса Circle.

18. Написать тексты h-файлов и сpp-файлов для класса Rect (прямоугольник). Описание класса:

Класс	Элементы данных	Интерфейс
Rect	x1, y1, x2, y2 (координаты левого верхнего и правого нижнего углов)	Конструкторы, функции move, операции: =, == (равно), != (не равно), !(возвращает площадь прямоугольника), <, > (сравнение по площади), <<, >>

Разработать и отладить программу с примерами создания и использования объектов класса Rect.

19. Написать тексты h-файлов и сpp-файлов для класса Treugolnik (треугольник). Описание класса:

Класс	Элементы данных	Интерфейс
Treugolnik	x1, y1, x2, y2, x3, y3	Конструкторы, функции move, square, операции =, <, > (сравнение площади), !(отразить треугольник относительно оси y) , <<, >>

Разработать и отладить программу с примерами создания и использования объектов класса `Treugolnik`.

20. Разработать класс `Line` (Прямая, заданная уравнением вида $y=kx+b$)

Класс	Элементы данных	Интерфейс
<code>Line</code>	<code>k, b</code>	Конструкторы, метод <code>IsOnLine(x,y)</code> – определяет, лежит ли точка с координатами (x,y) на прямой; операции <code>=, !</code> (отражает прямую относительно оси X), <code>*</code> (определяет точку пересечения двух прямых), <code>==, !=, <<, >></code> .

Написать тексты `h`-файла и `cpp`-файла для класса `Line`. Разработать и отладить программу создания и использования объектов класса.

**21. Написать тексты `h`-файлов и `cpp`-файлов для класса `CString` (строка).
Описание класса:**

Класс	Элементы данных	Интерфейс
<code>CString</code>	Длина строки, Символы строки	Конструкторы, функция <code>Len</code> (длина строки), операции <code>=, ==, !=, <, ></code> (сравнение строк), <code>+, <<, >></code>

Разработать и отладить программу с примерами создания и использования объектов класса `CString`. Добавить в программу возможность создания и сортировки (упорядочения) массива строк.

22. Создать класс `Arr` для работы с одномерным массивом целых чисел. Разработать следующие элементы класса:

Класс	Элементы данных	Интерфейс
<code>Arr</code>	массив целых значений: размер массива	Конструкторы, функция <code>Sort</code> (упорядочение массива), <code>Add</code> (добавление элемента в массив), <code>Del</code> (удаление элемента из массива), операции <code>=, +</code> (слияние массивов), <code>*</code> (умножение на скаляр), <code><<, >></code>

Разработать и отладить программу с примерами создания и использования объектов класса `Arr`.

**23. Написать тексты `h`-файлов и `cpp`-файлов для класса `CString` (строка).
Описание класса:**

Класс	Элементы данных	Интерфейс
<code>CString</code>	Длина строки, Символы строки	Конструкторы, функция <code>Len</code> (длина строки), функция <code>Find</code> (поиск подстроки), операции <code>=, ==, !=, <, ></code> (сравнение строк), <code>+, <<, >></code>

Разработать и отладить программу с примерами создания и использования объектов класса `CString`.

24. Разработать класс `CTime`:

Класс	Элементы данных	Интерфейс
<code>CTime</code>	<code>h</code> (часы), <code>m</code> (минуты), <code>s</code> (секунды)	Конструкторы, методы <code>SetTime</code> , операции <code>=</code> , <code>+</code> (прибавляет некоторое количество минут), <code>-</code> (возвращает разность времени в секундах), <code>++</code> (перевести время на 1 секунду вперед), <code><</code> , <code>></code> , <code><<</code> , <code>>></code> .

Написать тексты `h`-файла и `cpp`-файла для класса `CTime`. Разработать и отладить программу создания и использования объектов класса.

25. Написать тексты `h`-файлов и `cpp`-файлов для класса `Rectangle` (прямоугольник). Описание класса:

Класс	Элементы данных	Интерфейс
<code>Rectangle</code>	<code>x1</code> , <code>y1</code> , <code>x2</code> , <code>y2</code>	Конструкторы, функции <code>move</code> , операции: <code>=</code> , <code>==</code> (равно), <code>!=</code> (не равно), <code>!</code> (возвращает периметр прямоугольника), <code><</code> , <code>></code> (сравнение по площади), <code><<</code> , <code>>></code>

Разработать и отладить программу с примерами создания и использования объектов класса `Rectangle`.