

1. Определить класс CComplexVector для работы с вектором комплексных чисел.

Внутри класса вектор должен быть реализован с помощью указателя на тип complex (или двух указателей на int или double) и целой переменной, в которой хранится количество отведенной под вектор памяти (или длина вектора). Длина вектора задается в конструкторе класса и изменяется, если происходит присваивание вектору другой длины.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения, вычитания, скалярного умножения. При сложении и вычитании длина результата – это минимум из длин исходных векторов. При скалярном умножении недостающие координаты считать нулевыми.

В отдельном файле должен быть написан тест на данный класс.

2. Определить класс CVector для работы с вектором вещественных чисел.

Внутри класса вектор должен быть реализован с помощью указателя на тип double и целой переменной, в которой хранится количество отведенной под вектор памяти (или длина вектора). Длина вектора задается в конструкторе класса и изменяется, если происходит присваивание вектору другой длины.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения, вычитания, скалярного умножения. При сложении и вычитании длина результата – это минимум из длин исходных векторов. При скалярном умножении недостающие координаты считать нулевыми.

В отдельном файле должен быть написан тест на данный класс.

3. Определить класс CIntN для работы с целыми беззнаковыми числами, состоящими из N десятичных цифр.

Внутри класса число должно быть реализовано с помощью указателя на тип char и целой переменной, в которой хранится количество отведенной под число памяти. Число N задается в конструкторе класса.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения, вычитания. При сложении и вычитании количество значащих десятичных цифр результата (N) может отличаться от N аргументов.

В отдельном файле должен быть написан тест на данный класс.

4. Определить класс CRat для работы с вектором несократимых дробей вида p_i/q_i , где p_i — целое, q_i — натуральное.

Внутри класса вектор должен быть реализован с помощью указателей (одного или двух) и целой переменной, в которой хранится количество отведенной под вектор памяти (или длина вектора). Длина вектора задается в конструкторе класса и изменяется, если происходит присваивание вектору другой длины.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения, вычитания. При сложении и вычитании длина результата — это минимум из длин исходных векторов.

В отдельном файле должен быть написан тест на данный класс.

5. Определить класс `CIntm` для работы с вектором чисел в кольце вычетов по модулю m . Число m должно задаваться с помощью оператора `#define`. Внутри класса вектор должен быть реализован с помощью указателя и целой переменной, в которой хранится количество отведенной под вектор памяти (или длина вектора). Длина вектора задается в конструкторе класса и изменяется, если происходит присваивание вектору другой длины.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения, вычитания. При сложении и вычитании длина результата – это минимум из длин исходных векторов.

В отдельном файле должен быть написан тест на данный класс.

6. Определить классы `CPoint` для работы с массивом целочисленных точек на плоскости и `CDist` для работы с массивом расстояний между целочисленными точками.

Внутри класса массивы должны быть реализованы с помощью указателей (одного или двух) и целой переменной, в которой хранится количество отведенной под массив памяти (или длина массива). Длина массива задается в конструкторе класса.

В классах должны быть определены необходимые конструкторы (в том числе конструкторы копирования и перемещения), деструкторы, операторы присваивания (копированием и перемещением), сложения (`CPoint` и `CDist`, возвращающий `CPoint`, а также `CDist` и `CPoint`, возвращающий `CPoint`), вычитания (`CPoint` из `CPoint`, возвращающий `CDist`). При сложении и вычитании длина результата – это минимум из длин исходных векторов.

В отдельном файле должен быть написан тест на данный класс.

7. Определить класс CString для работы со строкой. Длина строки задается в конструкторе класса. Внутри класса строка должна быть реализована с помощью указателя (char *).

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения (конкатенация), умножения (слева и справа) строки на беззнаковое целое число (оно равносильно сложению строки с собой нужное число раз). При сложении и умножении строк слагаемые и сомножители не должны изменяться.

В отдельном файле должен быть написан тест на данный класс.

8. Определить класс CStr для работы со строкой. Длина строки задается в конструкторе класса. Внутри класса строка должна быть реализована с помощью указателя (char *).

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения строки с символом (добавление символа в конец строки), вычитания из строки последнего символа (удаление последнего, если это возможно). При сложении и вычитании строк слагаемые не должны изменяться. При невозможности вычитания должно генерироваться исключение.

В отдельном файле должен быть написан тест на данный класс.

9. Определить класс `CArray` для работы с массивом целых чисел. Внутри класса массив должен быть реализован с помощью указателя и целой переменной, в которой хранится количество отведенной под массив памяти (или длина массива). Длина массива задается в конструкторе класса.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения (дописывания одного массива в конец другого), умножения (слева и справа) массива на беззнаковое целое число (оно равносильно сложению массива с собой нужное число раз). При сложении и умножении массивов слагаемые и сомножители не должны изменяться.

В отдельном файле должен быть написан тест на данный класс.

10. Определить класс `CArr` для работы с массивом целых чисел. Внутри класса массив должен быть реализован с помощью указателя и целой переменной, в которой хранится количество отведенной под массив памяти (или длина массива). Длина массива задается в конструкторе класса.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения массива с целым числом (добавление числа в конец массива), уменьшения (левого и правого декремента –) массива (удаление последнего элемента, если это возможно). В случае ошибки операция декремента должна генерировать исключение. При сложении массива с числом исходный массив не должен изменяться.

В отдельном файле должен быть написан тест на данный класс.

11.

Определить класс CMatrix для работы с квадратной матрицей целых чисел. Внутри класса матрица должна быть реализована с помощью указателя (int **) и целой переменной, в которой хранится количество отведенной под матрицу памяти (или размер матрицы). Размер матрицы задается в конструкторе класса.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения, вычитания матриц, умножения матрицы (слева и справа) на число. При сложении и вычитании размер результата – это минимум из размеров исходных матриц, исходные матрицы при этом не должны меняться.

В отдельном файле должен быть написан тест на данный класс.

12.

Определить класс CComplexMatrix для работы с матрицей комплексных чисел. Внутри класса матрица должна быть реализована с помощью указателя (complex **, класс complex тоже должен быть реализован) или двух указателей int ** и целой переменной, в которой хранится количество отведенной под матрицу памяти (или размер матрицы).

Размер матрицы задается в конструкторе класса.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения и вычитания. При сложении и вычитании размер результата – это минимум из размеров исходных матриц, исходные матрицы при этом не должны меняться.

В отдельном файле должен быть написан тест на данный класс.

13.

Определить класс `CIntp` для работы с вектором чисел в поле вычетов по модулю p , (Z_p , p – простое число, которое задается с помощью оператора `#define`.) Внутри класса вектор должна быть реализован с помощью указателя (`int *`) и целой переменной, в которой хранится количество отведенной под вектор памяти (или длина вектора). Длина вектора задается в конструкторе класса.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения и деления (деление не должно быть основано на переборе). При делении на вектор, у которого некоторая координата равна 0, генерируется исключение. При сложении и делении размер результата – это минимум из размеров исходных векторов, исходные вектора при этом не меняются. В отдельном файле должен быть написан тест на данный класс.

14.

Определить класс `CSet` для работы с побитовым множеством целых чисел в диапазоне от 0 до N . Внутри класса множество должно быть реализовано с помощью указателя (`unsigned int *`) и целой переменной, в которой хранится верхняя граница диапазона. Число N задается в конструкторе класса. Принадлежность числа множеству означает, что бит, соответствующий этому числу, равен 1, в случае нулевого бита число не принадлежит множеству.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения (объединение множеств), вычитания (пересечения). При сложении верхняя граница диапазона результата равна максимуму из соответствующих границ слагаемых, при вычитании – минимуму. Исходные множества не должны при этих операциях меняться.

В отдельном файле должен быть написан тест на данный класс.

15.

Определить класс CRat для работы с полиномом от 2 переменных степени не выше N, N задается в конструкторе класса. Внутри класса полином должен быть реализован с помощью указателя (int *) и целой переменной, в которой хранится количество отведенной под полином памяти (или суммарная степень полинома).

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), операторы сложения, вычитания, распечатка полинома по возрастанию степеней переменных. При сложении и вычитании исходные полиномы не меняются.

В отдельном файле должен быть написан тест на данный класс с распечаткой полинома по степеням переменных.

16.

Определить класс CInt2 для работы с целыми беззнаковыми числами, состоящими из N двоичных цифр, N задается в конструкторе класса. Внутри класса число должно быть реализован с помощью указателя (unsigned int *) и целой переменной, в которой хранится количество отведенной под число памяти (или число N).

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), операторы сложения и вычитания (чисел в двоичной системе). Одна двоичная цифра должна занимать 1 бит. При сложении и вычитании исходные числа не меняются.

В отдельном файле должен быть написан тест на данный класс.

17.

Определить классы `CVect` для работы с массивом векторов на плоскости и `CDist` для работы с массивом углов между векторами. Длина массива задается в конструкторе класса.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), операторы сложения (`CVect` и `CDist`, возвращающий `CVect`, а также `CDist` и `CVect`, возвращающий `CVect`), вычитания (`CVect` из `CVect`, возвращающий `CDist`).

При сложении и вычитании длина результата – это минимум из длин исходных массивов, исходные массивы при этом не меняются.

В отдельном файле должен быть написан тест на данный класс.

18.

Определить класс `CPoly` для работы с многочленом от одной переменной с вещественными коэффициентами. Внутри класса полином должен быть реализован с помощью указателя (`double *`) и целой переменной, в которой хранится количество отведенной под полином памяти (или степень полинома). Степень полинома задается в конструкторе класса.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), оператор сложения (наибольший общий делитель слагаемых), при сложении исходные полиномы не меняются.

В отдельном файле должен быть написан тест на данный класс.

19.

Определить класс CStr для работы со строкой. Внутри класса строка должна быть реализована с помощью указателя (char *).

Длина строки задается в конструкторе класса.

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), операторы сложения строки с символом (добавление символа в конец строки), вычитания из строки символа (удаление последнего, если он совпадает с вычитаемым). При сложении и вычитании исходная строка не меняется.

В отдельном файле должен быть написан тест на данный класс.

20.

Определить класс CPoly для работы с многочленом от одной переменной с коэффициентами из поля вычетов Z_p , p – простое число, которое задается с помощью оператора #define. Степень многочлена задается в конструкторе класса. Внутри класса полином должен быть реализован с помощью указателя (int *) и целой переменной, в которой хранится количество отведенной под полином памяти (или степень полинома).

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения полинома с целым числом (если это число отрицательное, то результатом сложения должен быть продифференцированный полином, если число положительное – проинтегрированный.) При сложении и вычитании исходный полином не должен меняться.

В отдельном файле должен быть написан тест на данный класс.

21.

Определить класс `CArr` для работы с упорядоченным по возрастанию массивом целых чисел. Длина массива задается в конструкторе класса. Внутри класса массив должен быть реализован с помощью указателя (`int *`) и целой переменной, в которой хранится количество отведенной под массив памяти (или длина массива).

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения (массив, полученный слиянием слагаемых). При сложении исходные массивы не должны меняться.

В отдельном файле должен быть написан тест на данный класс.

22.

Определить класс `CMatrix` для работы с квадратной матрицей над полем Z_2 . Матрица будет определять множество решений соответствующей ее строкам системы линейных однородных уравнений. Размер матрицы задается в конструкторе класса. Внутри класса матрица должна быть реализована с помощью указателя (`int **`) и целой переменной, в которой хранится количество отведенной под матрицу памяти (или размер квадратной матрицы).

В классе должны быть определены необходимые конструкторы (в том числе конструктор копирования и перемещения), деструктор, операторы присваивания (копированием и перемещением), сложения, определяющий матрицу, для которой система решений является пересечением систем линейных уравнений слагаемых. При сложении матриц разного размера меньшую матрицу дополнить нулями (вложение линейных пространств). При сложении исходные матрицы не должны меняться.

В отдельном файле должен быть написан тест на данный класс.