

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

Длина вектора задается с помощью оператора #define. В классе должны быть определены необходимые конструкторы, операторы сложения, вычитания, скалярного умножения.

В отдельном файле должен быть написан тест на данный класс.

2.

Определить класс CVector для работы с вектором вещественных чисел.

Длина вектора задается с помощью оператора #define. В классе должны быть определены необходимые конструкторы, операторы сложения, вычитания, скалярного умножения.

В отдельном файле должен быть написан тест на данный класс.

3.

Определить класс CIntN для работы с целыми без знаковыми числами, состоящими из N десятичных цифр, где N задается с помощью оператора #define. В классе должны быть определены необходимые конструкторы, операторы сложения, вычитания,

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. Длина вектора задается с помощью оператора #define. В классе должны быть определены необходимые конструкторы, операторы сложения, вычитания.

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 в конструкторе класса. Длина вектора задается с помощью оператора #define. В классе должны быть определены необходимые конструкторы, операторы сложения, вычитания.

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. Длина вектора задается с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы, операторы сложения (`CPoint` и `CDist`, возвращающий `CPoint`, а также `CDist` и `CPoint`, возвращающий `CPoint`), вычитания (`CPoint` из `CPoint`, возвращающий `CDist`).

В отдельном файле должен быть написан тест на данный класс.

7. Определить класс `CString` для работы со строкой. Максимальная длина строки задается с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы, операторы сложения (конкатенация с обрезанием лишних символов), умножения (слева и справа) строки на без знаковое целое число (оно равносильно сложению строки с собой нужное число раз).

В отдельном файле должен быть написан тест на данный класс.

8. Определить класс `CStr` для работы со строкой. Максимальная длина строки задается с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы, операторы сложения строки с символом (добавление символа в конец строки, если это возможно), вычитания из строки последнего символа (удаление последнего, если это возможно).

В отдельном файле должен быть написан тест на данный класс.

9. Определить класс `CArray` для работы с массивом целых чисел. Максимальная длина массива задается с помощью оператора `#define`, фактическая длина задается в конструкторе. В классе должны быть определены необходимые конструкторы, операторы сложения (дописывания одного массива в конец другого с обрезанием лишних элементов), умножения (слева и справа) массива на беззнаковое целое число (оно равносильно сложению массива с собой нужное число раз).

В отдельном файле должен быть написан тест на данный класс.

10. Определить класс `CArr` для работы с массивом целых чисел. Максимальная длина массива задается с помощью оператора `#define`, фактическая длина задается в конструкторе. В классе должны быть определены необходимые конструкторы, операторы сложения массива с целым числом (добавление числа в конец массива, если это возможно), уменьшения – массива (удаление последнего элемента, если это возможно).

В отдельном файле должен быть написан тест на данный класс.

11. Определить класс `CMatrix` для работы с квадратной матрицей целых чисел. Размер матрицы задается с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы, операторы сложения, вычитания матриц, умножения матрицы (слева и справа) на число.

В отдельном файле должен быть написан тест на данный класс.

12.

Определить класс `CComplexMatrix` для работы с матрицей комплексных чисел. Размер матрицы задается с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы, операторы сложения, и вычитания.

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). Длина вектора задается с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы, операторы сложения и деления (деление не должно быть основано на переборе).

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 , N задается с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы, операторы сложения (объединение множеств), вычитания (пересечения).

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 с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы, операторы сложения, вычитания.

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

16.

Определить класс `CInt2` для работы с целыми без знаковыми числами, состоящими из N двоичных цифр, N задается с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы и оператор сложения (чисел в двоичной системе). Одна двоичная цифра должна занимать 1 бит.

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 с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы, операторы сложения (`CVect` и `CDist`, возвращающий `CVect`, а также `CDist` и `CVectt`, возвращающий `CVect`), вычитания (`CVect` из `CVect`, возвращающий `CDist`).

В отдельном файле должен быть написан тест на данный класс.

18.

Определить класс `CPoly` для работы с многочленом от одной переменной с вещественными коэффициентами. Максимальная степень полинома задается с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы и оператор сложения (наибольший общий делитель слагаемых).

В отдельном файле должен быть написан тест на данный класс.

19.

Определить класс `CStr` для работы со строкой. Максимальная длина строки задается с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы, операторы сложения строки с символом (добавление символа в конец строки, если это возможно), вычитания из строки символа (удаление последнего, если он совпадает с вычитаемым).

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. Максимальная степень полинома задается с помощью оператора `#define`. В классе должны быть определены необходимые конструкторы и оператор сложения полинома с целым числом (если это число отрицательное, то результатом сложения должен быть продифференцированный нужное число раз полином, если число положительное – проинтегрированный, в случае превышения степени результата числа N старший коэффициент игнорируется).

В отдельном файле должен быть написан тест на данный класс.

21.

Определить класс `CArr` для работы с упорядоченным по возрастанию массивом целых чисел. Максимальная длина массива задается с помощью оператора `#define`, фактическая длина задается в конструкторе. В классе должны быть определены необходимые конструкторы и оператор сложения (массив, полученный слиянием слагаемых, лишние числа игнорируются).

В отдельном файле должен быть написан тест на данный класс.

22.

Определить класс `CMatrix` для работы с квадратной матрицей над полем Z_2 . Размер матрицы задается с помощью оператора `#define`. Матрица будет определять множество решений соответствующей ее строкам системы линейных однородных уравнений. В классе должны быть определены необходимые конструкторы и оператор сложения, определяющий матрицу, для которой система решений является пересечением систем линейных уравнений слагаемых.

В отдельном файле должен быть написан тест на данный класс.