1. Реализовать класс матрицы целых чисел. Размерность матриц не ограничена. Класс должен:

* считывать элементы матрицы из файла формата csv;
* реализовывать конструкторы копирования и перемещения (если необходимо);
* реализовывать операторы копирующего и перемещающего присваивания (если необходимо);
* реализовывать операторы сложения, вычитания, умножения матриц;
* обрабатывать ошибки с использованием исключений

1. Реализовать класс матрицы целых чисел. Размерность матриц не ограничена. Класс должен:

* считывать элементы матрицы из файла формата csv;
* реализовывать конструкторы копирования и перемещения (если необходимо);
* реализовывать операторы копирующего и перемещающего присваивания (если необходимо);
* реализовывать операторы сложения, вычитания, сравнения матриц, умножения матрицы и числа;
* обрабатывать ошибки с использованием исключений

1. Реализовать шаблонный класс матрицы. Размерность матриц не ограничена. Класс должен:

* реализовывать методы для заполнения матрицы поэлементно и построчно;
* реализовывать операторы сложения и сравнения матриц (==, !=);
* реализовывать конструкторы копирования и перемещения (если необходимо);
* реализовывать операторы копирующего и перемещающего присваивания (если необходимо);
* реализовывать оператор выставки в поток (<<) для вывода матрицы в консоль;
* обрабатывать ошибки с использованием исключений

1. Реализовать шаблонный класс матрицы. Размерность матриц не ограничена. Класс должен:

* реализовывать методы для заполнения матрицы поэлементно и построчно;
* реализовывать оператор индексирования;
* реализовывать конструкторы копирования и перемещения (если необходимо);
* реализовывать операторы копирующего и перемещающего присваивания (если необходимо);
* реализовывать метод, транспонирующий матрицу;
* реализовывать оператор выставки в поток (<<) для вывода матрицы в консоль;
* обрабатывать ошибки с использованием исключений.

1. Разработать класс гистограммы целочисленных значений. Класс должен:

* определять конструктор, принимающий массив значений и количество интервалов. Диапазон значений разбивается на данное число интервалов и подсчитывается количество значений, попавших в каждый интервал. Совокупность шкалы интервалов и подсчитанных таким образом значений и есть гистограмма;
* определять конструктор по умолчанию;
* определять конструкторы копирования и перемещения (если необходимо);
* реализовывать операторы копирующего и перемещающего присваивания (если необходимо);
* определять оператор вставки в поток (<<), выводящий гистограмму в консоль при помощи ранее заданного символа. Пример вывода:

+++

+++++++

++

+++

* обрабатывать ошибки при помощи механизма исключений

1. Cмоделировать стрельбу по мишени группой человек. Каждый человек имеет свое имя, возраст (в годах) и стаж обучения стрельбе (в годах). Люди делятся на новичков, опытных и ветеранов. Каждый класс-человек должен реализовывать метод «shoot» без параметров, возвращающий логическое значение (попал – true, не попал – false). Факт попадания определяется случайным образом, причем для новичка вероятность попасть равна 0,01\*стаж обучения; для опытного = 0,05\*стаж обучения стрельбе; для ветерана = 0,9 – 0,01\*возраст. Люди стреляют по очереди, начиная с первого, пока кто-то не попадет в мишень. Стрельба прекращается после того, как кто-то попал или все выстрелили по одному разу. После каждого выстрела необходимо выводить на экран всю информацию о стрелявшем (перегрузить оператор вставки в поток) и результат стрельбы.
2. Смоделировать сдачу зачета студентами. Каждый студент характеризуется ФИО и количеством посещенных занятий. Студенты делятся на обычных, сообразительных и гениев. Каждый студент должен иметь метод «Сдать зачет» без параметров, возвращающий логическое значение (сдал – true, не сдал – false). Обычные студенты точно сдают зачет, если посетили все занятия, если были более чем на половине занятий, то сдают с вероятностью 0,5; иначе не сдают. Сообразительные студенты тоже точно сдают зачет, если посетили все занятия, если были более чем на половине занятий, то сдают с вероятностью 0,7; иначе не сдают зачет. Гении точно сдают зачет, если были хотя бы на одном занятии, иначе не сдают. Создайте массив из произвольного числа различных студентов, и смоделируйте сдачу зачета с выводом подробных результатов всех сведений о студентах (перегрузить оператор вставки в поток), а также результате сдачи.
3. Смоделировать нарезку листов стали. Листы стали характеризуются толщиной. Листы делятся на квадратные, прямоугольные и треугольные (в виде прямоугольного треугольника, задаются двумя катетами, в мм). Для каждый тип стального листа должен иметь метод «Площадь», возвращающий площадь листа. Также определите метод «Вес», который вычисляет вес листа, умножая площадь листа на его толщину и плотность стали. В главной программе создайте массив из произвольного числа различных листов стали. Выведите информацию о листах (перегрузить оператор вставки в поток) и рассчитайте суммарную площадь и суммарный вес всех листов.
4. Разработать шаблонный класс односвязного списка. Класс должен:

* определять метод для добавления элемента в начало списка;
* определять метод для удаления элемента из начала списка;
* определять метод для проверки, пустой ли список;
* определять оператор сложения списков, добавляющий второй список в конец первого;
* определять операторы постфиксного и префиксного декрементов, удаляющие элемент из начала списка;
* определять конструкторы копирования и перемещения (если необходимо);
* определять операторы копирующего и перемещающего присваивания (если необходимо);
* обрабатывать ошибки при помощи механизма исключений

1. Разработать шаблонный класс односвязного списка. Класс должен:

* определять метод для добавления элемента в начало списка;
* определять метод для добавления элемента в конец списка;
* определять метод для удаления элемента из начала списка;
* определять метод для проверки, пустой ли список;
* определять оператор умножения списка на число, повторяющий данный список N раз (пример: [1, 2, 3] \* 2 = [1, 2, 3, 1, 2, 3]);
* определять оператор вставки в поток, выводящий список в консоль;
* определять конструкторы копирования и перемещения (если необходимо);
* определять операторы копирующего и перемещающего присваивания (если необходимо);
* обрабатывать ошибки при помощи механизма исключений

1. Разработать шаблонный класс односвязного списка. Класс должен:

* определять метод для добавления элемента в начало списка;
* определять метод для удаления элемента из начала списка;
* определять метод для проверки, пустой ли список;
* определять метод join, возвращающий строку с перечислением элементов списка через разделитель, указанный в параметре метода (пример: [1, 2, 3].join(‘,’) = “1, 2, 3”);
* определять конструктор, принимающий строку с перечислением элементов списка через разделитель, и данный разделитель, как показано в примере выше. Принять, что ошибка преобразования типов невозможна, однако, формат строки может быть некорректным;
* определять конструкторы копирования и перемещения;
* определять операторы копирующего и перемещающего присваивания;
* обрабатывать ошибки при помощи механизма исключений

1. Разработать шаблонный класс комплексного числа в алгебраической форме. Принять, что пользователь не может задать невалидный формальный параметр шаблона (например, строку). Класс должен:

* содержать реальную и мнимую части и методы для их задания и доступа к ним;
* определять операторы сложения, вычитания, умножения, деления, сравнения комплексных чисел (==,!=);
* определять оператор умножения данного числа на действительное число;
* определять метод, возвращающий сопряжённое число;
* определять метод, возвращающий модуль комплексного числа;
* определять оператор вставки в поток (<<), выводящий в консоль комплексное число в виде Re при наличии только реальной части, iIm при наличии только мнимой части, Re + iIm в противном случае;
* обрабатывать ошибки при помощи механизма исключений

1. Разработать шаблонный класс комплексного числа в тригонометрической форме. Принять, что пользователь не может задать невалидный формальный параметр шаблона (например, строку). Класс должен:

* содержать модуль и аргумент числа и методы для их задания и доступа к ним;
* определять операторы умножения, деления, сравнения комплексных чисел (==, !=);
* определять оператор умножения данного числа на действительное число;
* определять метод для возведения данного числа в степень N;
* определять метод, возвращающий сопряжённое число;
* определять методы возвращающие действительную и мнимую части данного числа в алгебраической форме;
* определять оператор вставки в поток (<<), выводящий в консоль комплексное число в виде z(cos(f) + i\*sin(f));
* обрабатывать ошибки при помощи механизма исключений

1. Разработать шаблонный класс Pair, хранящий пару значений разных типов. Класс должен:

* реализовывать методы для доступа к значениям и задания значений;
* реализовывать операторы сравнения (==, !=);
* реализовывать оператор сложения, складывающий соответствующие элементы пары;
* реализовывать оператор выставки в поток (<<), выводящий в консоль пару в виде “(first, second)”;
* реализовать оператор извлечения из потока (>>), принимающий два значения из консоли и заполняющий ими пару строк;
* метод, принимающий два значения и возвращающий составленную из них пару;
* метод, принимающий пару и возвращающий пару, в которой первое значение поставлено на место второго, второе – на место первого