

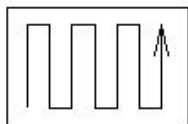
Задача 1

1. Будем называть соседями элемента с индексами i, j некоторой матрицы такие элементы, индексы которых отличаются от i, j не более чем на единицу. Для данной целочисленной матрицы $A (a_{ij})$ размерности $m \times m$ найти матрицу B , состоящую из нулей и единиц, элемент которой b_{ij} равен единице, когда среди соседей a_{ij} есть не менее двух совпадающих с a_{ij} .
2. Элемент матрицы назовем седловой точкой, если он является наименьшим в своей строке и одновременно наибольшим в своем столбце или, наоборот, является наибольшим в своей строке и наименьшим в своем столбце. Для заданной целой матрицы размером $N \times M$ напечатать индексы всех ее седловых точек.

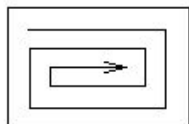
Выполнить следующие задания, используя процедуры и функции.

3. Для заданной вещественной матрицы определить, образуют ли ее элементы упорядоченную последовательность следующего вида:

a)



b)



4. В квадратной матрице определить количество строк
 - a) упорядоченных по возрастанию;
 - b) упорядоченных по убыванию;
 - c) состоящих из равных элементов;
 - d) неупорядоченных.

Использовать функцию, определяющую свойство каждой строки.

5. Задано конечное множество имен жителей некоторого города, причем для каждого из жителей перечислены имена его детей. Жители A и B называются родственниками, если:
 - либо A — ребенок B
 - либо B — ребенок A
 - либо существует некий V такой, что A является родственником V , а V является родственником B .Перечислить все пары жителей города, которые являются родственниками.
6. В матрице $A(n \times n)$ определить количество строк, элементы которой образуют арифметическую прогрессию. Использовать подпрограмму проверки строки.
7. В заданной матрице $A(n \times n)$ найти максимум из всех минимальных элементов матрицы по столбцам.
8. В заданной матрице $A(n \times n)$ найти минимум всех сумм абсолютных величин элементов матрицы по столбцам. Для нахождения суммы абсолютных величин столбца использовать функцию.
9. Подсчитать количество строк матрицы $A(n \times n)$, элементы которых образуют монотонную последовательность. Для определения монотонности использовать процедуру.

10. Уплотнить матрицу $A(n \times n)$ влево и вверх. Для выявления нулевых строк и столбцов использовать подпрограмму.
11. Матрица $A(n \times n)$ просматривается сверху вниз по строкам. Найти скалярное произведение строки и столбца, соответствующих строке с первым найденным отрицательным элементом и последним нулевым.
12. Упорядочить строки матрицы $A(m, n)$ по неубыванию суммы элементов строк.
13. Упорядочить строки матрицы $A(m, n)$ по неубыванию их евклидовых норм.
14. Упорядочить строки матрицы $A(m, n)$ по неубыванию количества их нечетных элементов.
15. Упорядочить строки матрицы $A(m, n)$ по неубыванию сумм цифр элементов строк.
16. Проверить, есть ли в матрице $A(n \times n)$ строки, не содержащие более двух отрицательных элементов.
17. Дана матрица $A(n \times n)$. Построить вектор, каждый элемент которого содержит наименьший по абсолютной величине элемент строки.
18. Составить программу поиска минимального элемента, расположенного под главной диагональю, и максимального элемента, расположенного над главной диагональю заданной вещественной матрицы $A(n \times n)$.
19. Заполнить квадратную матрицу $N \times N$ следующим образом: элементы, расположенные на главной диагонали, принять равными 1; выше главной диагонали — сумме индексов; ниже — их разности.
20. Заполнить квадратную матрицу $N \times N$ единицами и нулями в шахматном порядке, начиная с верхнего левого угла.
21. Дана матрица $N \times M$. Определить сумму элементов, кратных 3, и количество отрицательных элементов.
22. Задана квадратная матрица $N \times N$. Определить, где больше четных элементов: выше или ниже главной диагонали.
23. Дана квадратная целочисленная матрица $N \times N$. Найти суммы элементов строк, имеющих четные элементы на главной диагонали.
24. Даны две квадратные матрицы $A(N \times N)$ и $B(M \times M)$. Определить сумму элементов, расположенных на главных диагоналях.
25. Дана матрица $N \times M$. Определить четные элементы, имеющие нечетную сумму индексов. Показать индексы этих элементов.
26. Задана квадратная матрица $N \times N$. Найти суммы элементов тех строк, у которых элементы, расположенные на главной диагонали, равны нулю.
27. Дана действительная квадратная матрица $N \times N$. Найти максимальный элемент на главной диагонали и вывести строку, в которой он находится.
28. Задана квадратная матрица $N \times N$. Найти номер столбца K и строки L с максимальным произведением. Сформировать вектор $B(2n)$, нечетные элементы которого равны сумме, а четные — разности элементов K -го столбца и L -й строки матрицы.
29. В матрице $A(N \times M)$ найти строку с максимальной суммой элементов и строку с минимальной суммой элементов. Далее сформировать вектор $B(2 \times M)$, элементы которого чередовались бы с элементами максимальной и минимальной строк.

30. Дана действительная квадратная матрица $N \times N$. Требуется переставить строки матрицы по возрастанию первых элементов строк.
31. Дана действительная квадратная матрица $N \times N$. Требуется преобразовать матрицу: поэлементно вычесть последнюю строку из всех строк, кроме последней.
32. Задана прямоугольная матрица $A(N \times M)$. Найти k — номер строки с максимальной суммой элементов. Далее сформировать матрицу $B(N \times M)$, каждый элемент строки которой равнялся бы элементу соответствующей строки матрицы A , деленному на соответствующий элемент k -й строки.
33. Дана квадратная матрица $A(N \times N)$. За один просмотр элементов матрицы $A(N \times N)$ сформировать вектор $C(N)$, каждый j -й элемент которого равен произведению элементов j -го столбца исходной матрицы, и вектор $D(N)$, каждый j -й элемент которого равен сумме соответствующей строки матрицы A .
34. Задана квадратная матрица $A(N \times N)$. Найти k — номер столбца с максимальной суммой элементов и номер строки l с минимальной суммой элементов, а также элемент с минимальным значением в матрице A . Далее сформировать вектор $P(N)$, каждый элемент которого равен разности соответствующих элементов k -столбца и l -строки, деленной на минимальный элемент матрицы A .
35. Задана квадратная матрица $A(5 \times 5)$. Сформировать вектор $x(5)$, каждый элемент которого равен наибольшему значению элементов соответствующей строки исходной матрицы. Вычислить $x_1x_5 + x_2x_4 + x_3x_3 + x_4x_2 + x_5x_1$.
36. В данной целочисленной квадратной матрице $A(N \times N)$ указать индексы всех элементов, имеющих наибольшее значение.
37. Даны две прямоугольные матрицы $A(N \times M)$ и $B(N \times M)$. Найти матрицу $C(N \times M)$, элементы которой равны сумме соответствующих элементов матриц A и B , после чего произвести транспонирование полученной матрицы C .
38. Дана квадратная матрица $A(N \times N)$. За один просмотр найти строку с минимальной суммой элементов и строку с максимальной суммой элементов и образовать произведение этих строк.

Задача 2

1. Дано целое число N и текстовый файл. Создать новый файл, содержащий все слова длины N из исходного файла (знаки препинания, расположенные в начале и в конце слов, не учитывать). Если исходный файл не содержит слов длины N , оставить результирующий файл пустым.
2. Дан символ C (прописная русская буква) и текстовый файл. Создать строковый файл, содержащий все слова из исходного файла, начинающиеся₁оканчивающиеся₂ этой буквой (как прописной, так и строчной). Знаки препинания, расположенные в начале и в конце слов, не учитывать. Если исходный файл не содержит подходящих слов, оставить результирующий файл пустым.
3. Дано число N и текстовый файл. Удалить из файла абзац с номером N (абзацы отделяются друг от друга одной или несколькими пустыми строками и нумеруются от 1). Пустые строки, предшествующие и следующие за удаляемым абзацем, не удалять. Если абзац с данным номером отсутствует, то оставить файл без изменений.
4. Дано число N и текстовый файл. Удалить из файла абзац с номером N (абзацы выделяются с помощью красной строки (5 пробелов) и нумеруются от 1). Пустые строки между абзацами не учитывать и не удалять. Если абзац с данным номером отсутствует, то оставить файл без изменений.
5. Дан текстовый файл, каждая строка которого изображает целое число, дополненное слева и справа несколькими пробелами. Вывести сумму этих чисел и их количество.
6. Дан текстовый файл, каждая строка которого изображает целое или вещественное число, дополненное слева и справа несколькими пробелами (вещественные числа имеют ненулевую дробную часть). Вывести сумму целых₁вещественных₂ чисел и их количество.
7. Дан текстовый файл, каждая строка которого содержит изображения нескольких вещественных чисел, разделенных пробелами. Создать файл вещественных чисел, содержащий эти числа в том же порядке.
8. Даны два текстовых файла с именами $Name1$ и $Name2$. Добавить в начало₁конец₂ каждой строки файла $Name1$ соответствующую строку файла $Name2$. Если файл $Name2$ короче файла $Name1$, то оставшиеся строки файла $Name1$ не изменять.
9. Дан текстовый файл $NameT$ и файл целых чисел $NameN$. Добавить в начало₁конец₂ каждой строки файла $NameT$ изображение соответствующего числа из файла $NameN$. Если файл $NameN$ короче файла $NameT$, то оставшиеся строки файла $NameT$ не изменять.
10. Дан текстовый файл с именем $NameT$. В каждой его строке первые 60 позиций отводятся под текст, а оставшаяся часть — под вещественное число. Создать два файла: строковый файл с именем $NameS$, содержащий текстовую часть исходного файла, и файл вещественных чисел с именем $NameR$, содержащий числа из исходного файла.
11. Даны два файла целых чисел одного размера с именами $Name1$ и $Name2$. Создать текстовый файл с именем $NameT$, содержащий изображения этих чисел, расположенные в два столбца шириной по 30 символов: первый содержит числа из файла $Name1$, второй — из файла $Name2$. В начале и конце

каждой строки текстового файла ввести разделитель " | " (код 124). Числа выравниваются по левому₁|правому₂ краю столбца.

12. Даны вещественные числа A, B и целое число N . Создать текстовый файл, содержащий таблицу значений функции $f(x) = [\sin(x)]_1 | [\cos(x)]_2 | [\exp(x)]_3$ на промежутке $[A, B]$ с шагом $(B-A)/N$. Таблица состоит из двух столбцов: с аргументами x (10 позиций, из них 3 под дробную часть) и со значениями $f(x)$ (15 позиций, из них 8 под дробную часть). Столбцы выравниваются по правому краю и разделяются 10 пробелами.
13. Дан текстовый файл с именем *NameT*, содержащий таблицу из трех столбцов вещественных чисел. Ширина столбцов таблицы и способ их выравнивания являются произвольными. Специальных символов-разделителей таблица не содержит. Создать файлы вещественных чисел с именами *Name1*, *Name2* и *Name3*, каждый из которых содержит числа из соответствующего столбца таблицы.
14. Дан текстовый файл, представляющий собой таблицу, состоящую из трех столбцов с целыми числами. В начале и в конце каждой строки таблицы, а также между ее столбцами располагается символ-разделитель. Ширина столбцов таблицы и способ их выравнивания являются произвольными. Создать файл целых чисел, содержащий сумму чисел из каждой строки исходной таблицы.
15. Дан текстовый файл. Создать новый файл, содержащий все символы, встретившиеся в тексте, включая пробел и знаки препинания (без повторений). Символы располагать в порядке [возрастания их кодов]₁ | [убывания их кодов]₂ | [их первого появления в тексте]₃.
16. Дан текстовый файл с именем *NameT*. Подсчитать число повторений в нем строчных русских букв ("а"—"я") и создать строковый файл с именем *NameS*, элементы которого имеют вид: «<буква>—<число повторений данной буквы>». Буквы, отсутствующие в тексте, в файл не включать. Строки упорядочить по [возрастанию кодов букв]₁ | [убыванию числа повторений букв, а при равном числе повторений — по возрастанию кодов букв]₂.
17. Дано целое число N и текстовый файл с именем *Name1*, содержащий один абзац текста, выровненный по левому краю. Отформатировать текст так, чтобы его ширина не превосходила N позиций, и выровнять текст по левому₁|правому₂ краю. Пробелы в конце строк удалить. Сохранить отформатированный текст в новом текстовом файле с именем *Name2*.
18. Дано целое число N и текстовый файл *Name1*, содержащий текст, выровненный по левому краю. Абзацы текста отделяются друг от друга одной пустой строкой. Отформатировать текст так, чтобы его ширина не превосходила N позиций, и выровнять текст по левому₁|правому₂ краю, сохранив деление на абзацы. Пробелы в конце строк удалить. Сохранить отформатированный текст в новом текстовом файле *Name2*.
19. Дана строка K , состоящая из 10 цифр, и файл с русским текстом. Зашифровать файл, выполнив циклическую замену каждой русской буквы, стоящей на i -й позиции строки, на букву того же регистра, расположенную в алфавите на $K[i]$ -м месте после шифруемой буквы (символы строки K также перебираются циклически: для $i = 11$ снова используется смещение $K[1]$ и т.д.). Букву 'ё' в алфавите не учитывать, знаки препинания и пробелы не изменять.
20. Дана строка $S1$ и файл с русским текстом, зашифрованным по правилу, описанному в задании 39. Строка $S1$ представляет собой первую

расшифрованную строку текста. Расшифровать остальные строки и заменить в файле зашифрованный текст на расшифрованный. Если информации для расшифровки недостаточно, то исходный файл не изменять.

21. Прочитать текстовый файл. На отдельной форме вывести информацию о количестве слов, состоящих из одного символа, двух символов и т.д.
22. Прочитать текстовый файл. На отдельной форме вывести словарь, используемых слов с указанием частоты их использования.
23. Прочитать текстовый файл. На отдельной форме вывести текст, из которого удалены незначащие пробелы.
24. Прочитать текстовый файл. На отдельной форме вывести этот же текст, но в котором каждое новое предложение написано с новой строки (предложения заканчиваются точкой).
25. Прочитать текстовый файл. Реализовать функцию замены одного слова на другое.
26. Написать программу для кодирования и декодирования текстового файла с помощью слова-пароля.
27. Прочитать текстовый файл. На отдельной форме вывести текст, в котором все слова из латинских символов и цифр выделены другим цветом.
28. Прочитать текстовый файл. На отдельной форме вывести тот же текст, но в котором все числа записаны словами.
29. В текстовом файле записан английский /немецкий текст. Необходимо выделять сплошной фрагмент текста и определять, все ли буквы латинского алфавита в нём задействованы, отметить незадействованные.
30. Прочитать текстовый файл, в котором хранятся статьи уголовного кодекса. Разбить весь текст на отдельные статьи, если известно, что признаком начала статьи является предложение, состоящее из слова <Статья>, за которым следует номер статьи с точкой, написанное в начале строки.
31. Прочитать текстовый файл, в котором хранятся одна статья уголовного кодекса. Разбить эту статью на пункты, если известно, что признаком начала пункта является предложение, состоящее из номера пункта с точкой, написанное в начале строки.
32. Прочитать текстовый файл, в котором хранятся одна статья уголовного кодекса, состоящая из одного единственного пункта. Отделить от этой статьи санкцию, если известно, что она начинается со слов <наказывается> или <наказываются>.
33. Описать функцию *getInt(Name, k)* целого типа, возвращающую *k*-й элемент файла целых чисел с именем *Name* (элементы нумеруются от 0). Если файл не существует или не содержит *k*-го элемента, то функция возвращает 0. С помощью этой функции вывести пять элементов данного файла с указанными номерами.
34. Описать функцию *getLine(Name, k)* строкового типа, возвращающую *k*-ю строку текстового файла с именем *Name* (строки нумеруются от 0). Если файл не существует или не содержит *k*-й строки, то функция возвращает пустую строку. С помощью этой функции вывести пять строк данного файла с указанными номерами.
35. Описать функцию *IntFileSize(Name)* целого типа, возвращающую размер файла целых чисел с именем *Name*. Если файл не существует, то функция

возвращает -1 . С помощью этой функции определить размер трех файлов с данными именами.

36. Описать функцию *TextSize(Name)* целого типа, возвращающую число строк в текстовом файле с именем *Name*. Если файл не существует, то функция возвращает -1 . С помощью этой функции определить размер трех файлов с данными именами.
37. Описать процедуру *InvertIntFile(Name)*, меняющую порядок следования элементов файла целого типа с именем *Name* на противоположный. Если файл не существует или содержит менее двух элементов, то процедура не выполняет никаких действий. Обработать с помощью этой процедуры три файла с данными именами.
38. Описать процедуру *SplitIntFile(Name0, k, Name1, Name2)*, копирующую первые *k* (≥ 0) элементов существующего файла целых чисел с именем *Name0* в файл *Name1*, а остальные элементы — в файл *Name2* (прежнее содержимое результирующих файлов стирается). Один из результирующих файлов может оказаться пустым. Применить эту процедуру к файлу *Name0*, используя указанные значения *Name1*, *Name2* и *k*.
39. Описать процедуру *SplitText(Name0, k, Name1, Name2)*, копирующую первые *k* (≥ 0) строк существующего текстового файла с именем *Name0* в файл *Name1*, а остальные элементы — в файл *Name2* (прежнее содержимое результирующих файлов стирается). Один из результирующих файлов может оказаться пустым. Применить эту процедуру к файлу *Name0*, используя указанные значения *Name1*, *Name2* и *k*.
40. Описать процедуру *ConcatFile(NameA, NameB, NameAB)*, позволяющую объединить содержимое двух двоичных файлов *NameA* и *NameB* одного и того же типа в новом файле *NameAB*. Использовать процедуры *BlockRead* и *BlockWrite*. Применить эту процедуру к парам исходных файлов *Name1—Name2*, *Name1—Name3* и *Name2—Name3*, создав файлы с именами *Name12*, *Name13*, *Name23*.

Задача 3

Создать приложение, удовлетворяющее требованиям, приведенным в задании. Наследование применять только в тех заданиях, в которых это логически обосновано. Аргументировать принадлежность классу каждого создаваемого метода. Для каждого класса должно быть не менее 2 свойств.

1.

Создать объект класса Текст, используя классы Предложение, Слово.
Методы: дополнить текст, вывести текст, заголовок текста.

2.

Создать объект класса Автомобиль, используя классы Колесо, Двигатель.
Методы: ехать, заправляться, менять колесо, вывести марку автомобиля.

3.

Создать объект класса Самолет, используя классы Крыло, Шасси, Двигатель.
Методы: летать, задавать маршрут, вывести маршрут.

4.

Создать объект класса Государство, используя классы Область, Район, Город.
Методы: вывести столицу, количество областей, площадь, областные центры.

5.

Создать объект класса Планета, используя классы Материк, Океан, Остров.
Методы: вывести название материка, планеты, количество материков.

6.

Создать объект класса Звездная система, используя классы Планета, Звезда, Луна.
Методы: вывести количество планет в звездной системе, название звезды, добавление планеты в систему.

7.

Создать объект класса Компьютер, используя классы Винчестер, Оперативная память, Процессор, Видеокарта. Методы: включить, выключить, проверить на вирусы, вывести размер винчестера.

8.

Создать объект класса Квадрат, используя классы Точка, Отрезок.
Методы: задание размеров, растяжение, сжатие, поворот, изменение цвета.

9.

Создать объект класса Круг, используя классы Точка, Окружность.
Методы: задание размеров, изменение радиуса, определение принадлежности точки данному кругу.

10.

Создать объект класса Текстовый файл, используя классы Файл, Директория.
Методы: создать, переименовать, вывести содержимое, дополнить, удалить.

11.

Создать объект класса Одномерный массив, используя классы Массив, Элемент.
Методы: создать, вывести, выполнить операции (сложить, вычесть, перемножить).

12.

Создать объект класса Простая дробь, используя класс Число.
Методы: вывод, сложение, вычитание, умножение, деление.

13.

Создать объект класса Дом, используя классы Окно, Дверь.
Методы: закрыть на ключ, вывести на консоль количество окон, дверей.

14.

Создать объект класса Фотоальбом, используя классы Фотография, Страница.
Методы: задать название фотографии, дополнить фотоальбом фотографией, вывести количество фотографий.

15.

Создать объект класса Год, используя классы Месяц, День.
Методы: задать дату, вывести день недели по заданной дате, рассчитать количество дней, месяцев в заданном временном промежутке.

16.

Создать объект класса Сутки, используя классы Час, Минута.
Методы: вывести текущее время, рассчитать время суток (утро, день, вечер, ночь).

17.

Новогодний подарок.
Определить иерархию конфет и прочих сладостей. Создать несколько объектов-конфет. Собрать детский подарок с определением его веса. Провести сортировку конфет в подарке на основе одного из параметров. Найти конфету в подарке, соответствующую заданному диапазону содержания сахара.

18.

Домашние электроприборы.
Определить иерархию электроприборов. Включить некоторые в розетку. Подсчитать потребляемую мощность. Провести сортировку приборов в квартире на основе мощности. Найти прибор в квартире, соответствующий заданному диапазону параметров.

19.

Звукозапись.
Определить иерархию музыкальных композиций. Записать на диск сборку. Подсчитать продолжительность. Провести перестановку композиций диска на основе принадлежности к стилю. Найти композицию, соответствующую заданному диапазону длины треков.

20.

Камни.

Определить иерархию драгоценных и полудрагоценных камней. Отобрать камни для ожерелья. Подсчитать общий вес (в каратах) и стоимость. Провести сортировку камней ожерелья на основе ценности. Найти камни в ожерелье, соответствующие заданному диапазону параметров прозрачности.

21.

Транспорт.

Определить иерархию подвижного состава железнодорожного транспорта. Создать пассажирский поезд. Подсчитать общую численность пассажиров и багажа. Провести сортировку вагонов поезда на основе уровня комфортности. Найти в поезде вагоны, соответствующие заданному диапазону параметров числа пассажиров.

22.

Авиакомпания.

Определить иерархию самолетов. Создать авиакомпанию. Посчитать общую вместимость и грузоподъемность. Провести сортировку самолетов компании по дальности полета. Найти самолет в компании, соответствующий заданному диапазону параметров потребления горючего.

23.

Таксопарк.

Определить иерархию легковых автомобилей. Создать таксопарк. Подсчитать стоимость автопарка. Провести сортировку автомобилей парка по расходу топлива. Найти автомобиль в компании, соответствующий заданному диапазону параметров скорости.

24.

Страхование.

Определить иерархию страховых обязательств. Собрать из обязательств дериватив. Подсчитать стоимость. Провести сортировку обязательств в деривативе на основе уменьшения степени риска. Найти обязательство в деривативе, соответствующее заданному диапазону параметров.

25.

Мобильная связь.

Определить иерархию тарифов мобильной компании. Создать список тарифов компании. Подсчитать общую численность клиентов. Провести сортировку тарифов на основе размера абонентской платы. Найти тариф в компании, соответствующий заданному диапазону параметров.

26.

Налоги.

Определить множество и сумму налоговых выплат физического лица за год с учетом доходов с основного и дополнительного мест работы, авторских вознаграждений, продажи имущества, получения в подарок денежных сумм и имущества, переводов из-за границы, льгот на детей и материальной помощи. Провести сортировку налогов по сумме.

27.

Счета.

Клиент может иметь несколько счетов в банке. Учитывать возможность блокировки/разблокировки счета. Реализовать поиск и сортировку счетов. Вычисление общей суммы по счетам. Вычисление суммы по всем счетам, имеющим положительный и отрицательный балансы отдельно.

28.

Туристические путевки.

Сформировать набор предложений клиенту по выбору туристической путевки различного типа (отдых, экскурсии, лечение, шопинг, круиз и т.д.) для оптимального выбора. Учитывать возможность выбора транспорта, питания и числа дней. Реализовать выбор и сортировку путевок.

29.

Кредиты.

Сформировать набор предложений клиенту по целевым кредитам различных банков для оптимального выбора. Учитывать возможность досрочного погашения кредита и/или увеличения кредитной линии. Реализовать выбор и поиск кредита

30.

Фургон кофе.

Загрузить фургон определенного объема грузом на определенную сумму из различных сортов кофе, находящихся, к тому же, в разных физических состояниях (зерно, молотый, растворимый в банках и пакетиках). Учитывать объем кофе вместе с упаковкой. Провести сортировку товаров на основе соотношения цены и веса. Найти в фургоне товар, соответствующий заданному диапазону параметров качества.

Задача 4

Полиморфизм

Задание: построить класс 1-го уровня с указанными в индивидуальном задании полями и методами:

- конструктор;
- функция, которая определяет «качество» объекта – Q по заданной формуле (столбец 2);
- вывод информации об объекте.

Построить класс 2-го уровня (класс-потомок), который содержит:

- дополнительное поле P ;
- функция, которая определяет «качество» объекта класса 2-го уровня – Q_p , которая перекрывает функцию качества класса 1-го уровня (Q), выполняя вычисление по новой формуле (столбец 3).

Создать проект для демонстрации работы: ввод и вывод информации об объектах классов 1-го и 2-го уровней.

№ вар	Поля и функция «качества» (Q) класса 1-го уровня	Поле и функция «качества» Q_p класса 2-го уровня
1	Компьютер: - наименование процессора; - тактовая частота процессора (МГц); - объем оперативной памяти (Мб). $Q = (0,1 \cdot \text{частота}) + \text{память}$	P : объем винчестера (Гб) $Q_p = Q + 0,5 \cdot P$
2	Оператор мобильной связи: - название оператора; - стоимость 1 минуты разговора; - площадь покрытия. $Q = 100 \cdot \text{площадь покрытия} / \text{стоимость 1 минуты разговора}$	P : наличие платы за каждое соединение $Q_p = 0,7 \cdot Q$, если P - истина, иначе $Q_p = 1,5 \cdot Q$
3	Товар на складе: - наименование; - цена; - количество. $Q = \text{цена} / \text{количество}$	P : год выпуска товара $Q_p = Q + 0,5 \cdot (T - P)$, где T - текущий год
4	Кабель: - тип; - количество жил кабеля; - диаметр. $Q = \text{диаметр} / \text{количество жил}$	P : наличие оплетки Q_p : если P - истина, то $Q_p = 2 \cdot Q$; иначе $Q_p = 0,7 \cdot Q$

5	<p>Учебник по программированию:</p> <ul style="list-style-type: none"> - название; - количество страниц; - цена. <p>Q = цена / количество страниц</p>	<p>P: год издания</p> <p>$Qp = Q - 0,2 \cdot (T - P)$, где T - текущий год</p>
6	<p>Мобильный телефон:</p> <ul style="list-style-type: none"> - марка; - цена; - объем памяти. <p>Q = объем памяти / цена</p>	<p>P: количество SIM карт</p> <p>$Qp = Q \cdot P$</p>
7	<p>Спутниковая антенна (тарелка):</p> <ul style="list-style-type: none"> - диаметр; - материал; - цена. <p>Q = диаметр / цена</p>	<p>P: тип подвески (азимутальная, полярная, тороидальная)</p> <p>$Qp = Q$, если тип подвески азимутальный, $Qp = 2 \cdot Q$, если тип подвески полярный, $Qp = 2,5 \cdot Q$, если тип подвески тороидальный</p>
8	<p>Экзамен:</p> <ul style="list-style-type: none"> - дисциплина; - число студентов на экзамене; - продолжительность экзамена (ч). <p>Q = число студентов / продолжительность</p>	<p>P: процент двоек</p> <p>$Qp = Q \cdot (100 - P) / 100$</p>
9	<p>Спортсмен:</p> <ul style="list-style-type: none"> - фамилия; - число соревнований; - сумма мест, занятых спортсменом в соревнованиях. <p>Q = (число соревнований) / (сумма мест)</p>	<p>P: занимал ли хотя бы раз первое место</p> <p>$Qp = 1,5 \cdot Q$, если P - истина, иначе – $Qp = Q$,</p>
10	<p>Программист:</p> <ul style="list-style-type: none"> - фамилия; - число программ, написанных программистом; - число языков программирования, которыми он пишет программы. <p>Q = (число программ) * (число языков)</p>	<p>P: число программ, которые работают правильно</p> <p>$Qp = Q \cdot P$ / (число всех программ)</p>

11	<p>Спектакль: - название; $n1$ – число зрителей в начале; $n2$ – число зрителей в конце.</p> <p>$Q = (n2 - n1) / n1$</p>	<p>P: год написания пьесы</p> <p>$Qp = Q \cdot (T - P + 1)$, где T - текущий год</p>
12	<p>Алмаз:</p> <ul style="list-style-type: none"> - название; - вес (в каратах); - качество огранки в баллах (число). <p>$Q = 0,4 \cdot \text{вес} + 0,6 \cdot \text{качество огранки}$</p>	<p>P: цвет (белый, голубой, желтый, и тп)</p> <p>Qp: если цвет голубой, то $Qp = Q + 1$;</p> <p>а если желтый, то $Qp = Q - 0,5$</p> <p>иначе $Qp = Q$</p>
13	<p>Компьютерная сеть:</p> <ul style="list-style-type: none"> - название организации; - число рабочих станций; - среднее расстояние между станциями (м). <p>$Q = \text{число станций} \cdot \text{среднее расстояние}$</p>	<p>P: средняя скорость передачи данных в сети (Мб/с)</p> <p>$Qp = Q \cdot P$</p>
14	<p>Армия:</p> <ul style="list-style-type: none"> - вид войск; - численность (тыс человек); - вооруженность (баллы - число). <p>$Q = 0,3 \cdot \text{численность} + 0,7 \cdot \text{вооруженность}$</p>	<p>P: опыт (число месяцев, на протяжении которых армия вела боевые действия)</p> <p>$Qp = Q \cdot (P + 1)$</p>
15	<p>Автомобиль:</p> <ul style="list-style-type: none"> - марка автомобиля; - мощность двигателя (кВт); - число мест. <p>$Q = 0,1 \cdot \text{мощность} \cdot \text{число мест}$</p>	<p>P: год изготовления</p> <p>$Qp = Q - 1,5 \cdot (T - P)$, где T - текущий год</p>
16	<p>Партия:</p> <ul style="list-style-type: none"> - название; - численность (тыс. членов); - процент голосов на последних выборах. <p>$Q = 0,3 \cdot \text{численность} + 0,7 \cdot \text{процент гол}$</p>	<p>P: численность партии в прошлом году</p> <p>Qp: если численность в текущем году увеличилась, то $Qp = 1,2 \cdot Q$;</p> <p>а если сократилась, то $Qp = 0,8 \cdot Q$</p>

17	<p>Высшее учебное заведение:</p> <ul style="list-style-type: none"> - название заведения; - количество студентов, зачисленных на 1-й курс; - количество выпускников. <p>Q = количество выпускников / количество зачисленных</p>	<p>P: процент выпускников, которые работают по специальности</p> <p>$Qp = P \cdot Q$</p>
18	<p>Солдат:</p> <ul style="list-style-type: none"> - фамилия; - рост (м); - вес (кг). <p>Q = рост · вес</p>	<p>P: образование (начальное, среднее, высшее)</p> <p>Qp: если образование высшее, то $Qp = 2 \cdot Q$; а если начальное, то $Qp = 0,5 \cdot Q$; иначе $Qp = Q$</p>
19	<p>Телевизор:</p> <ul style="list-style-type: none"> - фирма; - диагональ экрана (дюйм); - звуковая мощность (дБ). <p>Q = диагональ + (0,05 · мощность)</p>	<p>P: страна-производитель Qp: если страна - Япония, то $Qp = 2 \cdot Q$; а если Сингапур или Корея, то $Qp = 1,5 \cdot Q$; иначе $Qp = Q$</p>
20	<p>Митинг:</p> <ul style="list-style-type: none"> - название события; $n1$ - число ораторов; $n2$ – число участников. $Q = n1/n2$ 	<p>P: число групп ораторов, которые высказывали одинаковые мысли</p> <p>$Qp = Q + P/n2$</p>
21	<p>Дом:</p> <ul style="list-style-type: none"> - номер дома; - число квартир; - год сооружения. <p>$Q = (\text{число квартир}) + 2 \cdot (T - \text{год сооружения}),$ где T - текущий год</p>	<p>P: район (центр, окраина, и тп)</p> <p>Qp: если район - центр, то $Qp = 2 \cdot Q$; иначе $Qp = 0,5 \cdot Q$</p>
22	<p>Руководитель:</p> <ul style="list-style-type: none"> - фамилия; - самооценка (в баллах - целое число); - оценка другими людьми (в баллах). <p>$Q = (\text{оценка другими}) / \text{самооценка}$</p>	<p>P: оценка потомками (в баллах)</p> <p>$Qp = 0,3 \cdot Q + 0,7 \cdot P$</p>
23	<p>Студент:</p> <ul style="list-style-type: none"> - фамилия; - средний балл; - курс. <p>$Q = 0,2 \cdot \text{средний балл} \cdot \text{курс}$</p>	<p>P: изучает дисциплины на английском языке</p> <p>$Qp = 2 \cdot Q$, если P - истина, иначе $Qp = 0,9 \cdot Q$</p>

24	<p>Антенна:</p> <ul style="list-style-type: none"> - название; - мощность; - <p>высота (м). $Q = \text{мощность} + 0.5 \cdot \text{высота}$</p>	<p>P: коэффициент излучения</p> $Q_p = Q - 0.1 \cdot P$
25	<p>Самолет:</p> <ul style="list-style-type: none"> - марка; - количество двигателей; - <p>высота полета. $Q = \text{кол-во двигателей} \cdot \text{высота полета} / 1000$</p>	<p>P: страна-производитель</p> <p>Q_p: если страна - Россия, то $Q_p = Q + 1$; а если Франция, то $Q_p = Q + 0.5$</p>
26	<p>Студент:</p> <ul style="list-style-type: none"> - фамилия; - число экзаменов; - <p>число оценок «пять». $Q = \text{число оценок «пять»} / \text{число экзаменов}$</p>	<p>P: число оценок «три»</p> $Q_p = Q - 0.5P$
27	<p>Фирма:</p> <ul style="list-style-type: none"> - название; - доход (тыс \$); - <p>рейтинг (в баллах). $Q = \text{доход} \cdot \text{рейтинг}$</p>	<p>P: инвестиции в фирму (тыс \$)</p> $Q_p = P^3 + Q$
28	<p>Военный корабль:</p> <ul style="list-style-type: none"> - название; - длина; - число пушек главного калибра. $Q =$ <p>(число пушек) / длина</p>	<p>P: крейсерская скорость (в морских узлах)</p> $Q_p = 0.25Q + P$
29	<p>Коробка спичек:</p> <ul style="list-style-type: none"> - фирма изготовитель; - число спичек в коробке; - время горения одной спички (с). $Q = (\text{число спичек}) \cdot \text{время}$	<p>P: средний % бракованных спичек в коробке</p> $Q_p = (100 - P)Q / 100$
30	<p>Полководец:</p> <ul style="list-style-type: none"> - фамилия; - число битв; - число <p>побед. $Q = (\text{число побед})^2 / (\text{число битв})$</p>	<p>P: число побед с меньшими, чем у противника, силами</p> $Q_p = P^2 / \text{битвы} + Q$

Задача 6

Для интерфейса необходимо определить 1 свойство и 2 метода.

Абстрактный класс должен содержать 3-5 свойств и 3-5 методов(включая унаследованные свойства интерфейса).

Класс должен содержать дополнительно 2 свойства и 2 метода.

В программе реализовать работу со списком объектов, который должен содержать объекты типа интерфейса.

1. interface Издание -> abstract class Книга -> class Справочник и Энциклопедия.
2. interface Абитуриент -> abstract class Студент -> class Студент-Заочник.
3. interface Сотрудник -> abstract class Инженер -> class Руководитель.
4. interface Здание -> abstract class Общественное Здание -> class Театр.
5. interface Mobile -> abstract class Siemens Mobile -> class Model.
6. interface Корабль -> abstract class Военный Корабль -> class Авианосец.
7. interface Врач -> abstract class Хирург -> class Нейрохирург.
8. interface Корабль -> abstract class Грузовой Корабль -> class Танкер.
9. interface Мебель -> abstract class Шкаф -> class Книжный Шкаф.
10. interface Фильм -> abstract class Отечественный Фильм -> class Комедия.
11. interface Ткань -> abstract class Одежда -> class Костюм.
12. interface Техника -> abstract class Плеер -> class Видеоплеер.
13. interface Транспортное Средство -> abstract class Общественный Транспорт -> class Трамвай.
14. interface Устройство Печати -> abstract class Принтер -> class Лазерный Принтер.
15. interface Бумага -> abstract class Тетрадь -> class Тетрадь Для Рисования.
16. interface Источник Света -> abstract class Лампа -> class Настольная Лампа.
17. interface Фигура -> abstract class Геометрическая фигура -> class Ромб.
18. interface Транспортное средство -> abstract class Автомобиль -> class Грузовик.
19. interface Устройство -> abstract class Фотоаппарат -> class Цифровой фотоаппарат.
20. interface Телефон -> abstract class Мобильный телефон -> class Смартфон.
21. interface Музыкальный инструмент -> abstract class Гитара -> class Электрогитара.
22. interface Вид спорта -> abstract class Командный вид спорта -> class Футбол.

23. interface Строительная техника -> abstract class Тяжелая строительная техника
-> class Экскаватор.

24. interface Участок -> abstract class Сад -> class Яблочный сад.

25. interface Хозяйство -> abstract class Животноводческая ферма -> class
Молочная ферма.

26. interface Космический объект -> abstract class Звезда -> class Красный гигант.

27. interface Вычислительное устройство -> abstract class Компьютер -> class
Ноутбук.

28. interface Коллектив -> abstract class Музыкальный коллектив -> class
Рок-группа.

29. interface Производство -> abstract class Завод -> class Кирпичный завод.

30. interface Добывающее предприятие -> abstract class Горнодобывающее
предприятие -> class Добыча руды.

Задача 7

Взять за основу задачу 6. Должно быть не менее 3 классов, которые наследуются от абстрактного класса.

Используя рефлексия реализовать возможность создания и вызова методов данных классов на форме. Написать код, который принимает путь к библиотеке классов и ищет все классы, которые реализуют интерфейс. Далее получают всю информацию о данных классах, и возвращают список из названий классов. На форме реализовать «дроплаун» с названиями классов. При выборе класса на форму должны динамически подгружаться все методы класса с возможностью ввода параметров пользователем. При нажатии кнопки «Выполнить» должен создаваться объект и выполняться выбранный метод.

Задача 8

В зависимости от задачи необходимо смоделировать ситуацию/процесс. В каждой модели есть набор возможных ситуаций. Для некоторых событий необходимо определить вероятность возникновения данного события. Интерфейс необходимо реализовать, используя 3 и более классов.

Для решения задач необходимо использовать:

1. Делегаты/события.
2. Многопоточность
3. Где необходимо рефлексия

На форме должно быть динамическое изменение моделей – все должно двигаться. Иметь возможность добавлять несколько моделей на форму.

1. Остановка – необходимо смоделировать работу остановки транспорта. Реализовать классы – Остановка, Автобус, интерфейс – Пассажир (реализовать несколько классов). События – автобус остановился на остановке (происходит посадка интерфейса пассажир), автобус переполнен (с некоторой долей вероятности).
2. Пешеходный переход – смоделировать работу пешеходного перехода. Реализовать классы – Авто, Пешеход, интерфейс – аварийная служба. События – зеленый свет для пешехода, аварийная ситуация(с некоторой долей вероятности) – должна приезжать машина аварийной службы.
3. Квадрокоптер – смоделировать полет квадрокоптера. Реализовать классы – Квадрокоптер, Оператор, интерфейс – механик. События – оператор включает пульт управления, на квадрокоптере отключается GPS(с некоторой долей вероятности). При аварийной ситуации квадрокоптер должен сесть, механик должен починить его.
4. Конвейер с деталями – смоделировать работу конвейера производства деталей. Реализовать классы – Конвейер, Погрузчик, интерфейс – Механик. События – в конвейере закончились материалы – погрузчик загружает новую партию, конвейер сломался (с некоторой долей вероятности) – механик чинит конвейер.
5. Гонки – смоделировать гонки. Реализовать классы – Болид, Механик, интерфейс – погрузчик. События – Стерлись покрышки – заезд в боксы смена колес, Столкновение (с некоторой долей вероятности) – приезжает погрузчик.
6. Троллейбус – смоделировать работу троллейбуса. Реализовать классы – Троллейбус, Водитель, интерфейс – Аварийная служба. События – сломался(с некоторой долей вероятности) – приезжает аварийная служба, соскочили штанги(с некоторой долей вероятности) – Водитель сам ставит на место.
7. Станок по производству деталей – смоделировать работу станка. Реализовать классы – Станок, Фрезеровщик, интерфейс – погрузчик. События – работать с новой деталью – Фрезеровщик берет новую деталь и работает с ней, Деталь готова – погрузчик забирает деталь.

8. Торговая точка – смоделировать работу торговой точки. Реализовать классы – Торговая точка, Покупатель, интерфейс – служба доставки. События – покупатель покупает товар, Товар закончился.
9. Спортивные соревнования – смоделировать спортивные соревнования. Реализовать классы – Соревнования, Спортсмен, интерфейс – врач. События – спортсмен получил травму (с некоторой долей вероятности) – приходит врач и лечит, Соревнования выиграл какой-то спортсмен – вручение награды.
10. Добыча нефти – смоделировать добычу нефти. Реализовать классы – Нефтяная вышка, Механик, интерфейс – погрузчик. События - возгорание платформы(с некоторой долей вероятности), отправка нефти – приезжает погрузчик.
11. Строительство дома – смоделировать строительство дома. Реализовать классы – Дом, Каменщик, интерфейс – строительная техника. События - Закончились стройматериалы, Необходимо крыть крышу.
12. Молочная ферма – смоделировать работу молочной фермы. Реализовать классы – Механик, Склад, интерфейс – погрузчик. События – Склад заполнен – приезжает погрузчик и увозит молоко, сломалось оборудование (с некоторой долей вероятности) – механик чинит всё.
13. Доменная печь – смоделировать работу доменной печи. Реализовать классы – Доменная печь, Рабочий, интерфейс – погрузчик. События – закончился материал, перегрев(с некоторой долей вероятности).
14. Производство конфет – смоделировать производство конфет. Реализовать классы – Фабрика, погрузчик, интерфейс – Техника. События - закончился сахар, случилась авария(с некоторой долей вероятности).
15. Ракета – смоделировать запуск ракеты. Реализовать классы – Ракета, Космонавт, интерфейс – страховая компания. События – успешный запуск – космонавты выходят в открытый космос, Авария(с некоторой долей вероятности) – страховая выплачивает много денег.
16. Остановка – необходимо смоделировать работу остановки транспорта. Реализовать классы – Остановка, Автобус, интерфейс – Пассажир (реализовать несколько классов). События – автобус остановился на остановке (происходит посадка интерфейса пассажир), автобус переполнен (с некоторой долей вероятности).
17. Пешеходный переход – смоделировать работу пешеходного перехода. Реализовать классы – Авто, Пешеход, интерфейс – аварийная служба. События – зеленый свет для пешехода, аварийная ситуация(с некоторой долей вероятности) – должна приехать машина аварийной службы.
18. Квадрокоптер – смоделировать полет квадрокоптера. Реализовать классы – Квадрокоптер, Оператор, интерфейс – механик. События – оператор включает пульт управления, на квадрокоптере отключается GPS(с некоторой долей вероятности). При аварийной ситуации квадрокоптер должен сесть, механик должен починить его.
19. Конвейер с деталями – смоделировать работу конвейера производства деталей. Реализовать классы – Конвейер, Погрузчик, интерфейс – Механик. События – в конвейере закончились материалы – погрузчик загружает новую партию, конвейер сломался (с некоторой долей вероятности) – механик чинит конвейер.

20. Гонки – смоделировать гонки. Реализовать классы – Болид, Механик, интерфейс – погрузчик. События – Стерлись покрышки – заезд в боксы смена колес, Столкновение (с некоторой долей вероятности) – приезжает погрузчик.
21. Троллейбус – смоделировать работу троллейбуса. Реализовать классы – Троллейбус, Водитель, интерфейс – Аварийная служба. События - сломался(с некоторой долей вероятности) – приезжает аварийная служба, соскочили штанги(с некоторой долей вероятности) – Водитель сам ставит на место.
22. Станок по производству деталей – смоделировать работу станка. Реализовать классы – Станок, Фрезеровщик, интерфейс – погрузчик. События – работать с новой деталью – Фрезеровщик берет новую деталь и работает с ней, Деталь готова – погрузчик забирает деталь.
23. Торговая точка – смоделировать работу торговой точки. Реализовать классы – Торговая точка, Покупатель, интерфейс – служба доставки. События – покупатель покупает товар, Товар закончился.
24. Спортивные соревнования – смоделировать спортивные соревнования. Реализовать классы – Соревнования, Спортсмен, интерфейс – врач. События – спортсмен получил травму (с некоторой долей вероятности) – приходит врач и лечит, Соревнования выиграл какой-то спортсмен – вручение награды.
25. Добыча нефти – смоделировать добычу нефти. Реализовать классы – Нефтяная вышка, Механик, интерфейс – погрузчик. События - возгорание платформы(с некоторой долей вероятности), отправка нефти – приезжает погрузчик.
26. Строительство дома – смоделировать строительство дома. Реализовать классы – Дом, Каменщик, интерфейс – строительная техника. События - Закончились стройматериалы, Необходимо крыть крышу.
27. Молочная ферма – смоделировать работу молочной фермы. Реализовать классы – Механик, Склад, интерфейс – погрузчик. События – Склад заполнен – приезжает погрузчик и увозит молоко, сломалось оборудование (с некоторой долей вероятности) – механик чинит всё.
28. Доменная печь – смоделировать работу доменной печи. Реализовать классы – Доменная печь, Рабочий, интерфейс – погрузчик. События – закончился материал, перегрев(с некоторой долей вероятности).
29. Производство конфет – смоделировать производство конфет. Реализовать классы – Фабрика, погрузчик, интерфейс – Техника. События - закончился сахар, случилась авария(с некоторой долей вероятности).
30. Ракета – смоделировать запуск ракеты. Реализовать классы – Ракета, Космонавт, интерфейс – страховая компания. События – успешный запуск – космонавты выходят в открытый космос, Авария(с некоторой долей вероятности) – страховая выплачивает много денег.

Задача 9

Доработать предыдущую задачу с использованием синхронизации потоков. На форме должно быть не менее 4 моделей. Ограничения накладываются на классы, которые реализуют интерфейсы. Для 4 моделей должно быть 2 объекта данных классов в сумме. При возникновении какого-то события 1 из объектов «лечится» и не доступен для использования в других моделях.