МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

МИРЭА – Российский технологический университет

Е.Н. Каширская, С.В. Антонов

сборник заданий для самостоятельной работы по процедурному программированию

практикум

Москва — 2020

УДК 004.42

ББК 32.973

К31

Каширская Е.Н. Сборник заданий для самостоятельной работы по процедурному программированию [Электронный ресурс]: Практикум / Каширская Е.Н., Антонов С.В. — М.: МИРЭА – Российский технологический университет, 2020. — 1 электрон. опт. диск (CD-ROM).

Практикум представляет собой сборник задач по программированию, которые разделены на несколько тематик согласно рабочей программе дисциплины «Процедурное программирование», которая входит в учебный план подготовки бакалавров направлений 01.03.04 «Прикладная математика», 09.03.01 «Информатика и вычислительная техника», 09.03.03 «Прикладная информатика», 09.03.04 «Программная инженерия» и 15.03.04 «Автоматизация технологических процессов и производств».

Практикум издается в авторской редакции.

Авторский коллектив: Каширская Елизавета Натановна, Антонов Сергей Валерьевич.

Рецензент:

Болбаков Роман Геннадьевич, к.т.н., доцент, заместитель заведующего кафедрой инструментального и прикладного программного обеспечения, институт информационных технологий РТУ МИРЭА

Минимальные системные требования:

Наличие операционной системы Windows, поддерживаемой производителем.

Наличие свободного места в оперативной памяти не менее 128 Мб.

Наличие свободного места в памяти хранения (на жестком диске) не менее 30 Мб.

Наличие интерфейса ввода информации.

Дополнительные программные средства: программа для чтения pdf-файлов (Adobe Reader).

Подписано к использованию по решению Редакционно-издательского совета   
МИРЭА – Российского технологического университета от \_\_\_ \_\_\_\_\_\_\_\_\_\_ 2020 г.

Объем \_\_\_ Мб

Тираж 10

© Каширская Е.Н., Антонов С.В., 2020

© МИРЭА – Российский технологический университет, 2020

**СОДЕРЖАНИЕ**

[Введение 4](#_Toc49336631)

[Линейные алгоритмы 5](#_Toc49336632)

[Разветвляющиеся алгоритмы 7](#_Toc49336633)

[Алгоритмы с множественным выбором 10](#_Toc49336634)

[Циклические алгоритмы 12](#_Toc49336635)

[Скалярные типы данных 30](#_Toc49336636)

[Форматирование вывода 30](#_Toc49336637)

[Структурированные типы данных 33](#_Toc49336638)

[Одномерные массивы 33](#_Toc49336639)

[Двумерные массивы 37](#_Toc49336640)

[Строки 41](#_Toc49336641)

[Динамические типы данных 43](#_Toc49336642)

[Указатели 43](#_Toc49336643)

[Динамические массивы 43](#_Toc49336644)

[Функции 47](#_Toc49336645)

[Функции как подпрограммы 47](#_Toc49336646)

[Рекурсивные функции 49](#_Toc49336647)

[Файлы 55](#_Toc49336648)

[Числовые файлы 55](#_Toc49336649)

[Текстовые файлы 55](#_Toc49336650)

[Обработка текстовых файлов 57](#_Toc49336651)

[Список литературы 73](#_Toc49336652)

[Сведения об авторах 74](#_Toc49336653)

Введение

Программирование сочетает в себе элементы искусства, науки, математики и инженерии. В узком смысле слова под программированием понимается кодирование, т.е. реализация одного или нескольких взаимосвязанных алгоритмов на некотором языке программирования, а в широком смысле программирование – это процесс создания программ или разработки программного обеспечения.

Большая часть работы программиста связана с написанием исходного кода на одном из языков программирования, который лучшим образом подходит для решения поставленной задачи. Различные языки программирования поддерживают различные стили программирования (парадигмы программирования), поэтому целесообразность применения того или иного языка обычно напрямую зависит от того, какую задачу нужно решить.

Однако все языки программирования используются для решения задачи, т.е. являются инструментом или способом интерпретации алгоритма решения этой задачи. Таким образом, основным навыком успешного и быстрого решения задач в области информационных технологий является способность создавать эффективные алгоритмы.

Под алгоритмом понимается точный набор инструкций, описывающих порядок действий некоторого исполнителя для достижения результата или решение некоторой задачи за конечное число шагов. Эти инструкции в составе алгоритма создаются в процессе алгоритмизации для решения поставленных прикладных задач.

Следовательно, начинающий программист для устойчивого развития своей карьеры и личности должен овладеть в первую очередь навыком алгоритмизации, а только после этого углубляться в изучение языковых инструментов в виде языков программирования, библиотек, сред разработки и пр.

Навык алгоритмизации может быть сформирован исключительно на основе решения прикладных задач различных тематик и уровней сложности, которые как раз представлены в данном практикуме. Для удобства они классифицированы по тематикам, которые в свою очередь упорядочены по возрастанию сложности и коррелируются с рабочей программой дисциплины «Процедурное программирование».

Линейные алгоритмы

1. Ввести с клавиатуры х и у и вычислить, вводя промежуточный аргумент:
   1. ;
   2. ;
   3. ;
   4. ;
   5. ;
   6. ;
   7. ;
   8. ;
   9. ;
   10. .
2. Даны x, y, z. Вычислить a, b, если:

,

.

1. Даны x, y, z. Вычислить a, b, если:

,

.

1. Даны x, y, z. Вычислить a, b, если:

,

.

1. 9 экземпляров одной книги стоят 11 рублей и несколько копеек, 13 экземпляров этой же книги стоят 15 рублей и несколько копеек. Сколько стоит один экземпляр книги?
2. Дано действительное число а. Не пользуясь никакими другими арифметическими операциями, кроме умножения, получить:
3. а3 и а10 за четыре операции;
4. а4 и а20 за пять операций;
5. а5 и а13 за пять операций;
6. а5 и а19 за пять операций;
7. а2, а5, а17 за шесть операций;
8. а4, а12, а28 за шесть операций.
9. Дан номер некоторого года (положительное целое число). Вывести соответствующий ему номер столетия, учитывая, что, к примеру, началом 20 столетия был 1901 год.
10. Написать программу, в которой по известной начальной скорости V и времени полета тела T определяется угол, под которым тело брошено по отношению к горизонту (можно воспользоваться следующим соотношением ).
11. Написать программу, которая позволяет пользователю ввести в консоль три координаты вектора x, y, и z, с основанием у начала координат. Вычислить длину этого вектора и вывести её обратно в консоль.
12. Вычислить значение выражения по формуле (все переменные принимают действительные значения): .
13. Составить программу, которая по введенной месячной зарплате, вычисляет подоходный налог.
14. Найти частное от деления и остаток используя только операции сложения и вычитания.

Разветвляющиеся алгоритмы

1. Ввести и вычислить:
2. Ввести и вычислить :









13. Ввести целое N. Определить, является ли оно:
14. не положительным;
15. нечетным;
16. превосходящим число –3;
17. степенью двойки.
18. Вести x, вычислить
19. Найти алгоритм работы чёрного ящика, который выполняет следующие преобразования входных данных: 1 → 0; 96 → 1; ЛЕС → 0; лес → 1; ворон → 6; Вова → 6; 88 → 4; 698 → …; юла → …; енот → … и т.д. Написать программу, которая определит результат для любого сочетания букв кириллицы.
20. Дана дыра в стене размером (a, b). определить пройдёт ли через эту дыру кирпич размером (x, y, z).
21. Поставить в порядке возрастания числа a, b, c.
22. Вычислить (согласно вариантам табл. 1):
23. значение функции при заданном X, вывести полученный результат и проверить принадлежность точки с координатами заштрихованной области
24. проверить принадлежность точки с координатами заштрихованной области.
25. – Варианты заданий к задаче с определением принадлежности точки заданной области.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № варианта п/п | Уравнение функции | Значение *x* | Заштрихованная область | Координаты исследуемой точки |
| 1 |  | 4,741 | *Прям* |  |
| 2 |  | 2,312 | 1 |  |
| 3 |  | 12,7409 | 1 |  |
| 4 |  | 32,872 | 1 |  |
| 5 |  | -2,6312 | 1 |  |
| 6 |  | -0,387 | 1 |  |
| 7 |  | 4,352 | 1 |  |
| 8 |  | 0,112 | 1 |  |
| 9 |  | -0,7129 | Полуромб |  |
| 10 |  | -4,4172 | 1 |  |

1. Даны вещественные координаты точки, не лежащей на координатных осях OX и OY. Вывести номер координатной четверти, в которой находится данная точка.
2. Даны координаты трёх точек на плоскости (x1, y1) (x2, y2) (x3, y3). Определить, принадлежит ли точка с координатами (x, y) треугольнику, построенному на заданных трёх точках.
3. Даны действительные числа a, b, c, d, e, f, g, h. Известно, что точки (e, f) и (g, h) различны. Известно также, что точки (a,b) и (c,d) не лежат на прямой l, проходящей через точки и . Прямая l разбивает координатную плоскость на 2 полуплоскости. Выяснить, верно ли, что точки и принадлежат одной и той же полуплоскости.
4. Определить номер четверти плоскости, в которой находится точка с координатами X и Y, причем X и Y не равны 0. Координаты вводить с клавиатуры.
5. Программа должна находить точку пересечения двух линий. Даны следующие уравнения: и , где a, b, c, d, k – это числа, которые вводятся с клавиатуры. Необходимо найти значение координат в точке(-ах) пересечения.
6. Вводятся координаты двух клеток шахматной доски. Вывести «Yes», если слон может за один ход перейти с одной клетки в другую, иначе вывести «No».

Алгоритмы с множественным выбором

1. Дано целое число в диапазоне 100 – 999. Вывести строку – словесное описание данного числа, например, 256 – «двести пятьдесят шесть», 814 – «восемьсот четырнадцать».
2. По введенному номеру дня недели вывести его название, используя оператор множественного выбора. Нумерация дней недели начинается с 1 – понедельник, 2 – вторник и т.д.
3. Единицы длины пронумерованы следующим образом: 1 – дециметр, 2 – километр, 3 – метр, 4 – миллиметр, 5 – сантиметр. Дан номер единицы длины и длина отрезка L в этих единицах (вещественное число). Вывести длину данного отрезка в метрах.
4. Элементы окружности пронумерованы следующим образом: 1 – радиус (R), 2 – диаметр (D), 3 – длина (L), 4 – площадь круга (S). Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данной окружности (в том же порядке). В качестве значения π использовать 3,14.
5. Арифметические действия над числами пронумерованы следующим образом: 1 – сложение, 2 – вычитание, 3 – умножение, 4 – деление. Дан номер действия и два числа A и B (В не равно нулю). Выполнить над числами указанное действие и вывести результат.
6. Даны два целых числа: D (день) и M (месяц), определяющие правильную дату невисокосного года. Вывести значения D и M для даты, предшествующей указанной.
7. Даны два целых числа: D (день) и M (месяц), определяющие правильную дату невисокосного года. Вывести значения D и M для даты, следующей за указанной.
8. Дано целое число в диапазоне 20 – 69, определяющее возраст (в годах). Вывести строку – словесное описание указанного возраста, обеспечив правильное согласование числа со словом «год», например: 20 – «двадцать лет», 32 – «тридцать два года», 41 – «сорок один год».
9. Локатор ориентирован на одну из сторон света («С» – север, «З» – запад, «Ю» – юг, «В» – восток) и может принимать три цифровые команды: 1 – поворот налево, -1 – поворот направо, 2 – поворот на 180 градусов. Дан символ C – исходная ориентация локатора и числа N1 и N2 – две посланные ему команды. Вывести ориентацию локатора после выполнения данных команд.
10. В восточном календаре принят 60-летний цикл, состоящий из 12-летних подциклов, обозначаемых названиями цвета: зеленый, красный, желтый, белый и черный. В каждом подцикле годы носят названия животных: крысы, коровы, тигра, зайца, дракона, змеи, лошади, овцы, обезьяны, курицы, собаки и свиньи. По номеру года вывести его название, если 1984 год был началом цикла – годом зеленой крысы.
11. На экране по очереди появляются вопросы (вопросы выбираются программистом), с вариантами ответов. В конце работы программа выдает количество заработанных баллов по результатам ответа. Допускается выполнение в формате теста.
12. Задание «Знаки Зодиака». Написать программу, которая по введённой дате некоторого дня года выдаёт на экран персонального компьютера знак Зодиака, с указанием интервала действия этого знака:

* 20.01 – 18.02: водолей;
* 19.02 – 20.03: рыбы;
* 21.03 – 19.04: овен;
* 20.04 – 20.05: телец;
* 21.05 – 21.06: близнецы;
* 22.06 – 22.07: рак;
* 23.07 – 22.08: лев;
* 23.08 – 22.09: дева;
* 23.09 – 22.10: скорпион;
* 23.11 – 21.12: стрелец;
* 22.12 – 19.01: козерог.

Циклические алгоритмы

1. Построить таблицу значений функции на отрезке с шагом с паузой через каждые N строк таблицы. Найти максимальное и минимальное значения и соответствующие им значения аргумента . Провести решение с использованием каждого из трёх основных типов циклов:
2. , при , , , ;
3. , при , , , ;
4. , при , , , ;
5. , при , , , ;
6. , при , , , ;
7. , при , , , ;
8. , при , , , ;
9. , при , , , ;
10. , при , , , ;
11. , при , , , .
12. Вычислить:
13. Сумму квадратов нечетных натуральных чисел (1, 3, 5, ...), в которую входят слагаемые, не большие 1000;
14. Максимальную сумму кубов четных чисел (2, 4, 6, ...), меньшую 8000;
15. Сумму квадратов натуральных чисел, кратных 3 (3, 6, 9, ...), в которую входят слагаемые, не большие 2500;
16. Сумму обратных значений натуральных чисел, кратных 4 (1/4, 1/8, 1/12, ...), в которую входят слагаемые, большие 0,005;
17. Сумму кубов нечетных натуральных чисел (1, 3, 5, 7, ...), в которую входят слагаемые, не большие 2000;
18. Максимальную сумму квадратов натуральных чисел, кратных 3 (3, 6, 9, 12, ...), меньшую 3000;
19. Сумму обратных значений нечетных натуральных чисел, (1, 1/3, 1/5, ...), в которую входят слагаемые, большие 0,01.
20. Максимальную сумму кубов чисел, кратных 3 (3, 6, 9, ...), меньшую 7000;
21. Сумму квадратов натуральных чисел, кратных 3 (3, 6, 9, ...), в которую входят слагаемые, не большие 5000;
22. Максимальную сумму обратных значений четных натуральных чисел (1/2, 1/4, 1/6, ...), в которую входят слагаемые, не меньшие 0,002. (Провести решение двумя способами: используя цикл с предусловием и цикл с постусловием).
23. Вычислить:
24. ;
25. ;
26. ;  
    ;
27. ;
28. ;
29. ;
30. ;
31. ;
32. ;
33. .
34. Построить таблицу функции на отрезке с шагом :
35. при , , ;
36. при , ,  
    ;
37. при , , ;
38. при , ,  
    ;
39. при , ,  
    ;
40. при , , ;
41. при , , ;
42. при , ,  
    ;
43. при , , ;
44. при , ,  
    ;
45. Вычислить:
46. ;
47. ;
48. ;
49. ;
50. ;
51. ;
52. ;
53. ;
54. ;
55. .
56. Вычислить сумму ряда:
57. Вычислить сумму ряда:
58. Дано целое число n (вводится с клавиатуры). Вычислить:
59. Дано целое число n (вводится с клавиатуры). Вычислить сумму из n

cлагаемых:

1. Дано целое число n (вводится с клавиатуры). Вычислить
2. Дано натуральное число n. Вычислить:
3. Дано вещественное число и . Последовательность образована по закону: , . Далее для выполнено:

Найти первый член () для которого выполняется условие .

1. Дано вещественное число и . Последовательность образована по закону: , , . Далее для выполнено:

Найти первый член для которого выполняется условие .

1. вводится с клавиатуры. Вычислить сумму с точностью для ряда:
2. Даны целое число *n*и вещественное, *а*(вводятся с клавиатуры). Вычислить:
3. Дано вещественное число и . Последовательность образована по закону:, , . Далее для выполнено:

Найти первый член () для которого выполняется условие .

1. Дано целое число n (вводится с клавиатуры). Вычислить
2. Дано натуральное число n. Вычислить:
3. вводится с клавиатуры. Используя рекуррентное соотношение, вычислить сумму с точностью :
4. Дано вещественное число . Вычислить ряд, учитывая только те слагаемые, в которых множитель имеет величину, не меньшую, чем :
5. Задача «Пифагоровы числа». Пифагорово число (пифагорова тройка) – комбинация из трёх целых чисел , удовлетворяющих соотношению Пифагора: . Необходимо найти и напечатать все пифагоровы числа в интервале . Например :3, 4, 5; 6, 8, 10; 9, 12, 15.
6. Задача «Автоморфные числа». Автоморфным называется число, равное последним цифрам своего квадрата. Например: 5 и 25 (52 = 25), 25 и 625 (252=625). Необходимо найти и напечатать все автоморфные числа в интервале . Например: на интервале [1,10000] найдены 1, 1; 5, 25; 6, 36; 25, 625; 76, 5776.
7. Задача «Числа Армстронга». Числом Армстронга называется число, состоящее из n цифр, если сумма его цифр, возведённых в n-ю степень, равна самому этому числу. Например, числом Армстронга является число 153, так как . Необходимо найти все n-значные числа Армстронга (n – входное данное, ). Например: при это числа 54748, 92727, 93084.
8. Задача «Числа-близнецы». Числа-близнецы (парные простые числа) – пары простых чисел, отличающихся на 2. Задание: найдите и напечатайте все числа-близнецы в интервале .
9. Числа Софи Жермен. В теории чисел простое число Софи Жермен – это такое простое число p, что число 2p + 1 также простое. Найти и напечатать все числа Софи Жермен в интервале .
10. Задача «Числа Каталана». Число Каталана выражается явной формулой:

или рекуррентной формулой:

Начало последовательности выглядит так: 1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796.

Найти и напечатать все числа Каталана в интервале .

1. Задача «Числа Нараяны». Числа Нараяны , , , формируют треугольную матрицу натуральных чисел, называемую Треугольником Нараяны. В XIV веке индийский математик Нараяна решил такую задачу: найти число коров и телок, появившихся от одной коровы за 20 лет, при условии, что корова в начале каждого года приносит телку, а телка дает такое же потомство в начале года, достигнув трех лет.

Найти и напечатать все числа Нараяны в интервале .

1. Числа Якобсталя. Числа Якобсталя определяются рекуррентным отношением:

Найти и напечатать все числа Якобсталя в интервале .

1. Найти Y, если ?. Количество X, то есть значение n вводится пользователем программы. Для каждого X значения Z, B, А, - разные (вводятся пользователем программы).
2. Найти алгебраическую сумму для выражения: . Значение и степень вводит пользователь.
3. Возвести, введенное с клавиатуры число в степень n, степень тоже вводится с клавиатуры. Возведение в степень организовать с использованием циклов, а не пользоваться стандартной функцией pow.
4. Составить алгоритм, определяющий, есть ли среди цифр введенного трехзначного числа одинаковые. Число вводится с клавиатуры.
5. Составить программу, которая на входе должна получать последовательность цифр, после чего программа показывает цифру, порядковый номер которой ввел пользователь.
6. С помощью цикла с предусловием разработать программу, которая будет вычислять сумму чисел, нацело делящихся на 5. Цикл задать от 0 до введенного с клавиатуры числа.
7. Используя показатели функции, вычислить цепную дробь:  
   . Количество элементов дроби необходимо задавать с клавиатуры.
8. По заданному вещественному x вычислить корень кубический из x по следующей итерационной формуле:

.

Начальное приближение . Итерации прекратить при . Смысл данного задания – найти корень кубический без использования специальных функций возведения в степень.

1. Определите число, полученное выписыванием в обратном порядке цифр заданного целого числа.
2. Вычислить выражение: .
3. Вычислить выражение:
4. Найти наименьшее общее кратное n и m.
5. Задание «Совершенные числа». Даны натуральные числа m и n. Найти все совершенные числа, меньшие n и большие m. Натуральное число называется совершенным, если оно равно сумме всех своих делителей, за исключением самого себя. Число 6 – совершенное, т.к. . Число 8 – не совершенное, т.к. .
6. Даны целые числа m, a1, a2, a3, …, a20. Найти три натуральных числа i, j, k, каждое из которых не превосходит 20, такие, что . Если таких чисел нет, то сообщить об этом.
7. Часовая стрелка образует угол φ с лучом, проходящим через центр и через точку, соответствующую 12 часам на циферблате, . Определить значение угла для минутной стрелки, а также количество часов и полных минут.
8. Дано натуральное число n. Определить, является ли число n палиндромом (перевёртышем, например, числа 6116, 34543 являются палиндромами).
9. В шестизначных трамвайных билетах (были такие когда-то) найти все счастливые билеты. Счастливым считался билетик, у которого сумма первых трех цифр равна сумме последних трех цифр.
10. Найти натуральное число от 1 до 10000 с максимальной суммой делителей.
11. Вычислить, не используя функцию pow, значения функции , для значений аргументов: x от -1,1 до 0,3 с шагом 0,2; m от 1 до 5 с шагом 1.
12. Запрограммировать генератор псевдослучайных чисел, используя конгруэнтный метод формирования псевдослучайных чисел. Случайные числа – возможные значения xi, непрерывной случайной величины X, распределённой равномерно на интервале [0;1). Формула определяющие конгруэнтный метод генерации псевдослучайных чисел:

где, a0, a1, ..., aj – множители, µ – значение инкремента, y1, y2, ..., yn – получаемые случайные числа, x1, x2, ..., xn – случайные числа на интервале [0;1).

Начальное значение, множитель и инкрементирующее число определяется произвольно в диапазоне от 0 до m.

1. Определить, является ли введённое число составным (не простым).
2. Вывести на экран все натуральные числа из диапазона от A до B, сумма цифр которых равна S. При отсутствии чисел с указанными свойствами выдать на экран сообщение «Требуемых чисел нет». Границы диапазона A и B и заданную сумму цифр S ввести с клавиатуры.
3. Вывести на экран все натуральные трехзначные и пятизначные числа из диапазона от A до B, значение которых кратно 13. При отсутствии чисел с указанными свойствами выдать на экран сообщение «Требуемых чисел нет». Границы диапазона A и B ввести с клавиатуры.
4. Вывести на экран все натуральные числа из диапазона от A до B, у которых совпадают старшая и младшая цифра. При отсутствии чисел с указанными свойствами выдать на экран сообщение «Требуемых чисел нет». Границы диапазона A и B ввести с клавиатуры.
5. Для всех натуральных чисел из диапазона от A до B вывести на экран сами числа и сумму цифр, имеющих нечетное значение. Границы диапазона A и B ввести с клавиатуры.
6. Для всех натуральных чисел из диапазона от A до B вывести на экран сами числа и сумму трех последних (младших) цифр. Границы диапазона A и B ввести с клавиатуры. Если диапазон не содержит трехзначных и выше чисел, выдать на экран сообщение «Диапазон указан ошибочно».
7. Ввести с клавиатуры натуральное число A. Определить и выдать на экран число, полученное путем вычитания единицы из каждой цифры исходного числа. Например, для исходного числа 3542 должно быть получено число 2431. В исходном числе не должно быть цифры 0. Если цифра 0 в исходном числе есть – выдать на экран сообщение «Нарушено условие задачи».
8. Вывести на экран все натуральные восьмизначные числа из диапазона от A до B, у которых совпадают сумма четырех младших и четырех старших цифр. При отсутствии чисел с указанными свойствами выдать на экран сообщение «Требуемых чисел нет». Границы диапазона A и B ввести с клавиатуры.
9. Вывести на экран все натуральные числа из диапазона от A до B, в записи которых цифра 7 встречается ровно N раз. При отсутствии чисел с указанными свойствами выдать на экран сообщение «Требуемых чисел нет». Границы диапазона A и B и значение N ввести с клавиатуры.
10. Вычислить и вывести на экран сумму ряда чисел, образуемых по правилу:

Сумма вычисляется до получения слагаемого, меньшего заданного значения A. Значение A ввести с клавиатуры.

1. Вычислить сумму первых N слагаемых последовательности чисел, образуемых по правилу:

Сумму вычислить двумя способами: S1 – суммирование от первого слагаемого до N-го слагаемого, S2 – суммирование от N-го слагаемого до первого слагаемого. Значение N ввести с клавиатуры. На экран вывести суммы S1 и S2 и модуль разности между ними.

1. Для заданного натурального числа N и вещественного числа A вычислить и вывести на экран:

Предел суммирования (произведения) N и значение A ввести с клавиатуры. Стандартную функцию pow не использовать.

1. Для заданного натурального числа N и вещественного числа A вычислить и вывести на экран:

где

Предел суммирования N ввести с клавиатуры. Стандартную функцию pow не использовать.

1. Для заданного натурального числа N вычислить и вывести на экран:

где

Предел произведения N ввести с клавиатуры.

1. Для заданного натурального числа N вычислить и вывести на экран:

Предел суммирования N ввести с клавиатуры. Стандартную функцию Pow не использовать.

1. Для заданного натурального числа N вычислить и вывести на экран:

Предел суммирования N ввести с клавиатуры

1. Для заданного натурального числа N вычислить и вывести на экран:

Предел суммирования N ввести с клавиатуры.

1. Для заданных натуральных чисел N, M и вещественного числа X вычислить и вывести на экран:

Пределы суммирования N, M и значение X ввести с клавиатуры.

1. Для заданного натурального числа N и вещественного числа x вычислить и вывести на экран:

Предел суммирования N и значение X ввести с клавиатуры. Стандартную функцию pow не использовать.

1. Для заданного натурального числа N и вещественного числа A вычислить и вывести на экран:
2. Предел суммирования N и значение A ввести с клавиатуры. Стандартную функцию Pow не использовать.
3. Вывести на экран все простые натуральные числа из диапазона от A до B. Простым является число, делящееся без остатка только на 1 и на самого себя. При отсутствии простых чисел в указанном диапазоне выдать на экран сообщение «Требуемых чисел нет». Границы диапазона A и B ввести с клавиатуры.
4. Задача «Числа Фибоначчи». Разработать программу вычисления n-чисел последовательности (ряда) Фибоначчи. Последовательность Фибоначчи задается множеством чисел, каждое из которых определяется, как сумма двух предыдущих, за исключением первых двух элементов последовательности – они равны единице: ; .
5. Последовательность вещественных чисел образуется по правилу:

где

Для заданного вещественного значения X найти и вывести на экран первый элемент последовательности, для которого выполнено условие:

Значение X ввести с клавиатуры.

1. Вычислить значение целого числа, десятичные цифры которого последовательно формируются датчиком случайных чисел. Первой генерируется старшая цифра. Количество генерируемых цифр N вводится с клавиатуры и должно находиться в диапазоне от 2 до 8. В случае ввода некорректного значения N предусмотреть возможность повторного ввода. Вычисленное целое число выдать на экран.
2. Вычислить значение целого числа, десятичные цифры которого последовательно формируются датчиком случайных чисел. Первой генерируется младшая цифра. Количество генерируемых цифр N вводится с клавиатуры и должно находиться в диапазоне от 2 до 8. В случае ввода некорректного значения N предусмотреть возможность повторного ввода. Вычисленное целое число выдать на экран.
3. Проверить две логические функции и на тождественность путем сравнения их значений по таблицам истинности.

По результатам проверки выдать на экран одно из сообщений: «Тождественны» или «Не тождественны».

1. Проверить три логические функции , и на тождественность путем сравнения их значений по таблицам истинности.

По результатам проверки выдать на экран одно из сообщений: «Тождественны» или «Не тождественны».

1. Вычислить и вывести на экран таблицу значений функции двух переменных

Изменение аргументов задано в виде: начальное значение (шаг) конечное значение и составляет: x =-0.2(0.1)2.0, y = -1.0(0.2)0.4. На каждой строке экрана вывести текущее значение аргументов и значение функции.

1. Вычислить и вывести на экран таблицу значений функции одной переменной:

Изменение аргумента задано в виде: начальное значение (шаг) конечное значение и составляет: . На каждой строке экрана вывести текущее значение аргумента и значение функции.

1. Вычислить и вывести на экран таблицу значений функции трех переменных

Изменение аргументов задано в виде: начальное значение (шаг) конечное значение и составляет: «x = -1.2(0.1)1.0, y = 4.0(0.2)5.0, z = -2.2(0.1)-1.6». На каждой строке экрана вывести текущее значение аргументов и значение функции.

1. Вычислить и вывести на экран таблицу значений функции трех переменных:

где

Изменение аргументов задано в виде: начальное значение (шаг) конечное значение и составляет: x = 0(π/8)π, y = 0(0.1)0.8, z = 0(0.1)0.8. На каждой строке экрана вывести текущее значение аргументов и значение функции.

1. Написать программу, которая введёт числа N и K такие, что 0 <К < N < 1000 и вычислит количество сочетаний, то есть способов выбрать К предметов из N возможных. Формулы комбинаторики не использовать.
2. Вводится строка не более чем из 6 цифр и некоторое целое число R. Расставить знаки арифметических операций («+», «-«, «\*», «/»; минус не является унарным, т.е. не может обозначать отрицательность числа; деление есть деление нацело, т.е. ) и открывающие и закрывающие круглые скобки так, чтобы получить в результате вычисления получившегося выражения число R. Лишние круглые скобки ошибкой не являются. Например: Строка 505597, R=120: . Необходимо решить задачу, не используя рекурсию.
3. Задача «Биоритмы человека»

Составить программу для определения физической, эмоциональной и интеллектуальной активности человека. Известные данные о биоритмах (в сутках): 23,6884 – период физической активности, 28,4261 – период эмоциональной активности, 33,1638 – период интеллектуальной активности.

На экран должны быть выведены следующие результаты работы программы:

* дата рождения;
* текущая дата;
* общее количество дней, часов, минут и секунд, разделяющих обе даты;
* прогноз на месяц вперед дат, соответствующих максимумам, минимумам и нулям биоритмов каждого из трех основных видов человеческой активности;
* возможные «неблагоприятные» даты совпадения нулей двух или трех биоритмов;
* возможные «опасные» даты совпадения минимумов двух или трех биоритмов;
* графики физической, эмоциональной, интеллектуальной и суммарной активности человека, если сумеете.

1. Задача «Календарь». Введите дату рождения и подсчитайте, сколько лет, месяцев и дней прошло с неё до сегодняшнего дня.
2. Задача «Ханойская башня». Утверждается, что эту задачу сформулировали и решают до сих пор монахи каких-то монастырей Тибета. Задача состоит в том, чтобы пирамидку из колец (на манер детской игрушки), нанизанную на один из 3-х стержней, перенести на другой такой же стержень, придерживаясь строгих правил:

* пирамидка состоит из n колец разного размера, уложенных по убыванию диаметра колец одно на другое;
* перекладывать за одну операцию можно только одно кольцо с любого штыря на любой, но только при условии, что класть можно только меньшее кольцо сверху на большее, но никак не наоборот;
* нужно, в итоге, всю исходную пирамидку, лежащую на штыре № 1, переместить на штырь № 3, используя штырь № 2 как промежуточный.

Например, для 2-х колец результат получается такой вот последовательностью перекладываний: 1 => 2, 1 => 3, 2 => 3.

По преданию эту задачу по перекладыванию n = 64 колечек решают тибетские монахи, и когда они её, наконец, решат, тогда и наступит конец света.

Решение состоит в том, чтобы при необходимости переноса пирамиды из n колец с штыря с номером from на штырь с номером to последовательно сделать следующее:

* перенести (каким-то образом) меньшую пирамиду из n-1 колец временно на штырь с номером temp, чтобы не мешала;
* перенести оставшееся единственное нижнее (наибольшее) кольцо на результирующий штырь с номером to, после чего, точно так же как в первом пункте, водрузить пирамиду (n-1 колец) с номера temp поверх этого наибольшего кольца на штырь с номером to.

Здесь важно то, что мы не знаем, каким образом выполнить алгоритм, и не умеем выполнить 1-й и 3-й пункты нашей программы, но надеемся, что алгоритм будет рекурсивно раскручиваться по аналогии, то есть по тому же алгоритму, но для меньшего числа n-1 (основополагающий принцип рекурсии), пока n не станет равным 1, а там уже совсем просто.

И вот как разворачивается решение для различных n:

n=2

1 => 2 | 1 => 3 | 2 => 3 |

общее число перемещений 3

n=3

1 => 3 | 1 => 2 | 3 => 2 | 1 => 3 | 2 => 1 | 2 => 3 | 1 => 3 |

общее число перемещений 7

n=4

1 => 2 | 1 => 3 | 2 => 3 | 1 => 2 | 3 => 1 |

3 => 2 | 1 => 2 | 1 => 3 | 2 => 3 | 2 => 1 |

3 => 1 | 2 => 3 | 1 => 2 | 1 => 3 | 2 => 3 |

общее число перемещений 15

n=5

1 => 3 | 1 => 2 | 3 => 2 | 1 => 3 | 2 => 1 |

2 => 3 | 1 => 3 | 1 => 2 | 3 => 2 | 3 => 1 |

2 => 1 | 3 => 2 | 1 => 3 | 1 => 2 | 3 => 2 |

1 => 3 | 2 => 1 | 2 => 3 | 1 => 3 | 2 => 1 |

3 => 2 | 3 => 1 | 2 => 1 | 2 => 3 | 1 => 3 |

1 => 2 | 3 => 2 | 1 => 3 | 2 => 1 | 2 => 3 | 1 => 3 |

общее число перемещений 31

Для потребуется 1023 перестановки; для любого n число перестановок равно , что представляет собой очень высокую степень роста вычислительной сложности задачи – экспоненциальную.

Примечание: решить эту задачу не рекурсивными методами – очень непростое занятие!

1. Задача «Квадрат и куб». Представить целое число в виде отношения квадрата и куба каких-то целых чисел. Кстати, почему эта задача всегда имеет решение?
2. Дано натуральное число n. Получить все его натуральные делители.
3. Найти 20 первых простых чисел.
4. Задача «Решето Сундарама». Решето Сундарама – алгоритм поиска всех простых чисел в некотором заданном диапазоне. Он был разработан в 1934 году ныне безызвестным студентом из Индии С. П. Сундарамом.

Принцип работы алгоритма Сундарама сводится, как и в его знаменитом предшественнике, к последовательному отсеиванию всех ненужных чисел. Но у него есть одна небольшая особенность: результатом работы алгоритма будет последовательность простых чисел из диапазона от 2 до удвоенного значения граничного числа. Допустим необходимо получить все простые числа до некоторого N, тогда выходными данными будут все простые числа от 2 до .

Решето Сундарама из ряда натуральных чисел, не превышающих N, исключает числа вида . Результат данного выражения ни при каких значениях входящих в него переменных не превышает N (). Соблюдая это условие, а также то, что i всегда меньше или равно j, переменные i и j пробегают все натуральные значения из множеств:

После исключения всех ненужных чисел необходимо увеличить каждое оставшиеся число в два раза и прибавить единицу (). Итоговое множество будет содержать числа: 2, 3, …, .

1. Задано целое положительное число n. Разработать программу вычисления значения у по формуле:
2. Разработать программу, которая определяет цифровой корень числа. Цифровой корень числа находится через сумму составляющих его цифр до тех пор, пока эта сумма сама не станет цифрой. Например, для числа 155 цифровой корень находится так: 1+5+5=11, 1+1=2. Цифровой корень 155 равен 2.
3. Найти сумму целых положительных чисел, кратных 4 и меньших 100.
4. Найти сумму целых положительных четных чисел, меньших 100.
5. Найти сумму целых положительных нечетных чисел, меньших 200.
6. Найти сумму целых положительных чисел, больших 20, меньших 100 и кратных 3.
7. Найти сумму ряда с точностью , общий член которого .
8. Найти сумму ряда с точностью , общий член которого .
9. Найти сумму ряда с точностью, общий член которого .
10. Найти сумму ряда с точностью, общий член которого .
11. Найти сумму ряда с точностью, общий член которого .
12. Найти сумму ряда с точностью, общий член которого .
13. Найти сумму ряда с точностью, общий член которого .
14. Найти сумму ряда с точностью, общий член которого .
15. Найти сумму ряда с точностью, общий член которого .
16. Найти сумму ряда с точностью, общий член которого .
17. Найти сумму ряда с точностью, общий член которого .
18. Найти сумму 10 членов ряда, в котором .
19. Найти сумму 15 членов ряда, в котором .
20. Найти сумму 12 членов ряда, в котором .
21. Найти сумму 7 членов ряда, в котором .
22. Найти сумму 9 членов ряда, в котором .
23. Написать программу, которая определяет количество учеников в классе, чей рост превышает средний. Рекомендуемый вид экрана во время работы программы приведен далее (введенные пользователем данные выделены полужирным шрифтом):

/\*\*\*\* Анализ роста учеников \*\*\*

Введите рост (см) и нажмите <Enter>.

Для завершения введите 0 и нажмите <Enter>

**-> 175**

**-> 170**

**-> 180**

**-> 168**

**-> 170**

**-> 0**

Средний рост: 172.6 см

У 2-х человек рост превышает средний.\*/

Скалярные типы данных

Форматирование вывода

1. Составить программу, которая будет считывать введённое пятизначное число. После чего, каждую цифру этого числа необходимо вывести в новой строке.
2. Запрограммировать следующее выражение: . Числа вводятся с клавиатуры. Организовать пользовательский интерфейс, таким образом, чтобы было понятно, в каком порядке должны вводиться числа.
3. Написать программу конвертации введённого с клавиатуры числа из метров в километры. Использовать вещественный тип, а результат округлять до десятых.
4. Написать программу, в которой в качестве входных данных принимается ASCII-код, а в консоль выводится соответствующий этому символу код.
5. Программа должна нарисовать домик. Допускается использовать следующие символы: символы слэша («/», «\», «|»), знак минуса («-»), символ подчёркивания («\_»).
6. Используя один оператор вывода cout, программа должна напечатать прямоугольный треугольник из символов знака «+».
7. Написать программу, запрашивающую имя, фамилия, отчество и номер группы студента и выводящую введённые данные в следующем виде:

/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Лабораторная работа № 1 \*

\* Выполнил(а): ст. гр. ЗИ-123 \*

\* Иванов Андрей Петрович \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

Необходимо, чтобы программа сама определяла нужную длину рамки. Сама же длина рамки зависит от длины наибольшей строки внутри рамки.

1. Составить программу, которая требует ввести два числа, а затем выполняет их сравнение и вычисляет разницу. Результат выводится в формате «X (больше/меньше) Y на (X-Y)» или «Введённые числа равны».
2. Написать программу, которая позволяет пользователю ввести в консоли латинскую букву нижнего регистра и переводит её в верхний регистр, а результат выводит в консоль.
3. Составить алгоритм увеличения всех трех, введённых с клавиатуры, переменных на 5, если среди них есть хотя бы две равные. В противном случае выдать ответ «равных нет».
4. Организовать ввод двухзначного натурального числа с клавиатуры. Программа должна определить наименьшую и наибольшую цифры, которые входят в состав данного натурального числа.
5. Выполнить ввод числа, выделить целую и дробную части, вывести на экран в формате десятичной и обыкновенной дроби (входные данные вещественного типа).
6. Написать программу, которая должна определить, пройдет ли кирпич в отверстие. Размеры отверстия (длина и высота) вводит пользователь. Тоже самое касается габаритов кирпича, пользователь вводит в программу значения длины, ширины и высоты кирпича. Программа должна изобразить кирпич и отверстие с помощью псевдографики на консоли.
7. Составить программу, которая будет печатать ромб одним из значков: «\*», «+» или «#». Выбор значков организовать оператором switch.
8. Написать программу преобразования десятичного числа в шестнадцатеричное. Допускается использовать манипулятор вывода hex.
9. Программа должна выводить таблицу квадратов чисел от 1 до 5 (необходимо использовать табуляцию и перенос на новую строку). В первом столбце должны находиться числа от 1 до 5 во втором столбце – квадраты этих чисел.
10. С помощью оператора цикла с параметром, разработать программу, которая будет выводить таблицу умножения, введённого пользователем числа с клавиатуры.
11. Требуется вывести в консоль от 1-го до 6-ти равнобедренных треугольников, в зависимости от того какое число введет пользователь. Например, если на вход в программу будет отправлено число 3, то необходимо напечатать 3 треугольника, если пользователь введет 6, то напечатать 6-ть треугольников. Треугольники нужно выводить не в столбик, а в строчку. При этом размер треугольников также формируется по введенному пользователем числу. Например, если пользователь введет число 3, то должно напечататься три равнобедренных треугольника, высота которых равна – 3-м. Для печати треугольников можно выбрать любой символ.
12. Время задается в формате час, минута, секунда. Реализовать:
13. вычитание из времени указанного пользователем количества секунд;
14. подсчёт числа секунд между двумя моментами времени, лежащими в пределах одних суток.
15. Задача «Синтезатор». Написать программу, которая генерирует звук определенной частоты в зависимости от нажатой клавиши и изображает высоту звука условными ступеньками. Таким образом, программа должна имитировать игру на синтезаторе.
16. Программа проверяет, является ли введенное с клавиатуры число – целым числом. Если введённое число нецелое – вывести ближайшее целое число.

Структурированные типы данных

Одномерные массивы

1. Пусть А – одномерный массив N вещественных чисел.
2. Поменять местами максимальный элемент и минимальный положительный элемент.
3. Проверить, является ли пятый ненулевой элемент по модулю меньшим 2, а если такого элемента нет, вывести сообщение.
4. Переставить все отрицательные элементы в начало массива.
5. Пусть C – одномерный массив вещественных чисел длины N. Ввести его через генератор случайных чисел и найти:
6. сумму элементов с четными номерами и среднее гармоническое ненулевых элементов;
7. произведение элементов с четными номерами и среднее квадратичное положительных элементов;
8. сумму элементов с нечетными номерами и среднее квадратичное положительных элементов;
9. произведение элементов с нечетными номерами и среднее геометрическое элементов, больших 2,5;
10. сумму элементов с номерами, кратными 3 и среднее квадратичное элементов с нечетными номерами;
11. произведение элементов с номерами, кратными 3 и среднее гармоническое положительных элементов;
12. сумму квадратов элементов с четными номерами и среднее квадратичное нечетных элементов;
13. сумму модулей элементов с номерами, кратными 4 и среднее геометрическое положительных элементов
14. сумму квадратов элементов с нечетными номерами и среднее квадратичное элементов с четными номерами;
15. среднее квадратичное четных элементов и произведение элементов с номерами, кратными 4.
16. Пусть A – одномерный массив из N вещественных чисел. Ввести его случайным образом. Далее выполнить (после каждой перестановки массив распечатать и оформить в виде процедур перестановку элементов и распечатку массива):
17. Поменять местами максимальный элемент и элемент, ближайший к числу -2,75; проверить, является ли четвертый ненулевой элемент отрицательным, а если такого элемента нет – вывести сообщение; переставить все нулевые элементы в конец массива.
18. Поменять местами максимальный по модулю элемент и элемент, ближайший к числу 1,33; проверить, является ли пятый ненулевой элемент числом, по модулю меньшим 2, а если такого элемента нет – вывести сообщение; переставить все отрицательные элементы в начало массива.
19. Поменять местами максимальный элемент и элемент, наиболее удаленный от числа 3,62; проверить, является ли шестой неположительный элемент числом, большим -5, а если такого элемента нет – вывести сообщение; переставить все положительные элементы в конец массива.
20. Поменять местами минимальный по модулю элемент и элемент, максимально удаленный от числа -0,93; проверить, является ли четвертый ненулевой элемент отрицательным, а если такого элемента нет – вывести сообщение; переставить все нулевые элементы в конец массива.
21. Поменять местами максимальный элемент и минимальный положительный элемент; проверить, является ли пятый ненулевой элемент числом, по модулю меньшим 2, а если такого элемента нет – вывести сообщение; переставить все отрицательные элементы в начало массива.
22. Пусть A – одномерный массив из N целых чисел. Ввести его случайным образом. Далее выполнить (после каждой перестановки массив распечатать и оформить в виде процедур перестановку элементов и распечатку массива):
23. Поменять местами минимальный по модулю элемент и максимальный отрицательный элемент; проверить, является ли шестой четный элемент положительным, а если такого элемента нет – вывести сообщение; исключить из массива все нулевые элементы.
24. Поменять местами минимальный элемент и минимальный положительный элемент; проверить, является ли третий нечетный элемент отрицательным, а если такого элемента нет – вывести сообщение; исключить из массива все положительные элементы.
25. Поменять местами минимальный элемент и минимальный положительный элемент; проверить, является ли пятый отрицательный элемент четным, а если такого элемента нет – вывести сообщение; исключить из массива все отрицательные элементы.
26. Поменять местами максимальный по модулю элемент и элемент, ближайший к числу 1,87; проверить, является ли шестой четный элемент положительным, а если такого элемента нет – вывести сообщение; исключить из массива все нулевые элементы.
27. Поменять местами минимальный элемент и максимальный отрицательный элемент; проверить, является ли третий нечетный элемент отрицательным, а если такого элемента нет – вывести сообщение; исключить из массива все положительные элементы.
28. Дан целочисленный массив A размера 10. Вывести номер первого из тех его элементов A[i], которые удовлетворяют двойному неравенству: . Если таких элементов нет, то вывести 0.
29. Пусть А, В, С, D – одномерные массивы длины N, составленные из элементов заданного типа. При этом массив А вводится с клавиатуры, массив В строится по формуле, массив С заполняется случайным образом значениями, принадлежащими отрезку [a;b], а массив D получается из этих массивов заданной линейной комбинацией. Ввести А, создать остальные массивы и распечатать все перечисленные массивы, используя процедуру вывода массивов при условиях: ; тип массивов – float; ; ; .
30. Даны действительные числа a1, a2, a3, …, a50. Получить «сглаженные» значения a1, a2, a3, …, a50, заменив в исходной последовательности все члены, кроме первого и последнего, по формуле . Считается, что после того, как получено новое значение некоторого члена, оно используется для вычисления нового значения следующего члена.
31. Даны: натуральные n и a1, a2, a3, …, an. Написать алгоритм вычисления последовательности: a1, .
32. Даны действительные числа a1, a2, a3, …, a50. Получить «сглаженные» значения a1, a2, a3, …, a50, заменив в исходной последовательности все члены, кроме первого и последнего, по формуле . Считается, что при «сглаживании» используются лишь старые значения членов последовательности.
33. Даны натуральное число n и символы s1, s2, …, sn. Известно, что среди s1, s2, …, sn есть по крайней мере одна запятая. Определить такое i, что si – последняя по порядку запятая.
34. Дан массив размера N и число k (, ). Осуществить циклический сдвиг элементов массива влево на k позиций.
35. Проверить, образуют ли элементы целочисленного массива размера N арифметическую прогрессию. Если да, то вывести разность прогрессии, если нет – вывести 0.
36. Даны множества A и B, состоящие соответственно из N1 и N2 точек. Найти минимальное расстояние между точками этих множеств и сами точки, расположенные на этом расстоянии.
37. Дано множество A из N точек. Найти наименьший периметр треугольника, вершины которого принадлежат различным точкам множества A, и сами эти точки (точки выводятся в том же порядке, в котором они перечислены при задании множества A).
38. Заданы массивы и B. Необходимо найти такие индексы i0 и j0, что , где , , .
39. Определить в программе все 12 месяцев года. Для них:
40. вывести на экран в одну строку через пробел названия всех четных месяцев и их продолжительность;
41. вывести названия месяцев с продолжительностью 31 день;
42. вывести названия месяцев, содержащие не менее пяти символов.
43. Дана точка A и множество B из N точек. Найти номер точки из множества B, наиболее близкой к А.
44. Дан массив целых чисел. Не используя других массивов, переставить элементы массива в обратном порядке.
45. Дан массив x[n], причём . Найти количество различных чисел среди элементов этого массива.
46. Дан массив размера N. Определить количество участков, на которых его элементы монотонно возрастают.
47. Дан массив размера N. Найти максимальный и |минимальный из его локальных минимумов.
48. Дано множество A из N точек с целочисленными координатами. Порядок на координатной плоскости определим следующим образом: , если либо , либо и . Расположить точки данного множества по возрастанию или убыванию в соответствии с указанным порядком.
49. Дан массив размера N. Вывести индексы массива в том порядке, в котором соответствующие им элементы образуют возрастающую или убывающую последовательность.
50. Дан массив размера N. Вывести индексы массива в том порядке, в котором соответствующие им элементы образуют возрастающую или убывающую последовательность.
51. Дан целочисленный массив размера N. Определить максимальное количество его одинаковых элементов. Решить задачу самым оптимальным способом.
52. Дан массив целых чисел , рассматриваемый как соединение двух его отрезков: начала длины m и конца длины n. Не используя дополнительных массивов, переставить начало и конец.
53. Даны две целочисленные таблицы. Разработать алгоритм, который проверяет, являются ли эти таблицы похожими. Две таблицы называются похожими, если совпадают множества чисел, встречающихся в этих таблицах.
54. Задаётся словарь. Найти в нём все анаграммы (слова, составленные из одних и тех же букв).
55. Дано n точек на плоскости. Указать (n-1)-звенную несамопересекающуюся замкнутую ломаную, проходящую через все эти точки. Соседним отрезкам ломаной разрешается лежать на одной прямой.
56. Напишите программу для слияния двух отсортированных одномерных массивов разной длины в один отсортированный.
57. Дан одномерный массив, длину массива задаёт пользователь. Вычислить сумму квадратов тех чисел, модуль которых превышает значение 2,5.

Двумерные массивы

1. Пусть B – двумерный массив целых чисел размерности N x N. Ввести его. Поменять местами (массив после каждой перестановки распечатать, используя процедуру распечатки массива):
2. Максимальный из четных элементов и минимальный из элементов побочной диагонали; 1-ый и 3-ий столбцы массива; строки с минимальной и максимальной суммой элементов.
3. Максимальный из нечетных элементов и минимальный из ненулевых элементов; 2-ую и 4-ую строки массива; главную и побочную диагонали.
4. Максимальный по модулю элемент и минимальный из элементов главной диагонали; 2-ую и 3-ю строки массива; строки с минимальным и максимальным произведением элементов.
5. Максимальный по модулю из нечетных элементов и минимальный из положительных элементов; 2-ой и 4-ый столбцы массива; строки с максимальной и минимальной суммой квадратов элементов.
6. Максимальный из ненулевых элементов и минимальный из нечетных элементов; 1-ый и 2-ой столбцы массива; строки, содержащие минимальный и максимальный элементы.
7. Максимальный из отрицательных элементов и минимальный из четных элементов; 3-ю и 4-ую строки массива; столбцы с максимальным и минимальным модулем суммы элементов.
8. Максимальный по модулю из элементов главной диагонали и минимальный из элементов массива; 1-ую и 4-ую строки массива; столбцы с минимальным и максимальным произведением элементов.
9. Максимальный из элементов массива и минимальный по модулю из элементов побочной диагонали; 1-ую и 2-ую строки массива; столбцы с максимальной и минимальной суммой квадратов элементов.
10. Максимальный отрицательный элемент массива и минимальный из нечетных элементов; 3-ий и 4-ый столбцы массива; строки с минимальным и максимальным модулем произведения элементов.
11. Максимальный из элементов побочной диагонали и минимальный по модулю элемент массива; 1-ую и 3-ю строки массива; столбцы, содержащие максимальный и минимальный элементы.
12. Дана матрица размера 5 x 9. Найти суммы элементов всех ее четных столбцов.
13. Даны целые числа . Получить матрицу для которой .
14. Пусть А, В, С, D, E – двумерные массивы размерности M x N, составленные из элементов заданного типа. При этом А вводится с клавиатуры, В строится по формуле, С заполняется случайным образом значениями, лежащими на отрезке , Е заполняется единицами, а D получается из перечисленных массивов заданной линейной комбинацией. Ввести А, создать все остальные массивы и распечатать все массивы, используя процедуру вывода массива. При условиях: ; ; тип int; ; ; .
15. Даны натуральные числа n и m, действительная матрица [n x m]. Найти среднее арифметическое каждого из столбцов и записать в вектор .
16. Дана действительная квадратная матрица порядка [n x n]. В строках с отрицательным элементом на главной диагонали найти сумму всех элементов.
17. Дана матрица целых чисел размера m x n. Подсчитать количество отрицательных элементов в нечетных строках.
18. Дана квадратная матрица размера n x n. В строках с отрицательным элементом на главной диагонали вычислить сумму всех элементов и найти максимальный элемент.
19. Дана действительная матрица размера [n x m], все элементы которой различны. В каждой строке выбирается элемент с наименьшим значением, затем среди этих чисел выбирается наибольшее. Указать индексы элемента с найденным значением.
20. Дана действительная квадратная матрица порядка [n x n]. Найти наименьшее из значений элементов побочной диагонали и двух соседних с ней линий.
21. Дана целочисленная матрица размера M x N. Вывести номер ее последней строки, содержащей максимальное количество одинаковых элементов.
22. Дана квадратная матрица порядка M. Повернуть ее на 270 градусов в положительном направлении.
23. Дана целочисленная матрица размера M x N. Различные строки (столбцы) матрицы назовем похожими, если совпадают множества чисел, встречающихся в этих строках (столбцах). Найти количество строк, похожих на последнюю строку.
24. Дана матрица размера M x N. Поменять местами ее строки так, чтобы их максимальные элементы образовывали возрастающую последовательность.
25. Пусть A, B, C, D, E – двумерные массивы размерности M x N, составленные из элементов заданного типа. При этом массив A вводится с клавиатуры, массив B строится по формуле, массив C заполняется случайным образом значениями, лежащими на отрезке , E – единичная матрица, а D получается из перечисленных массивов заданной линейной комбинацией. Ввести массив A, создать оставшиеся массивы и распечатать ВСЕ перечисленные массивы, используя процедуру вывода массива при следующих условиях:
26. , , тип – целочисленный, ,  
    , , ;
27. , , тип – вещественный, ,  
    , , ;
28. , , тип – целочисленный, ,  
    , , ;
29. , , тип – вещественный, ,  
    , , ;
30. , , тип – целочисленный, ,  
    , , ;
31. , , тип – вещественный, ,  
    , , ;
32. , , тип – целочисленный, , , , ;
33. , , тип – вещественный, ,  
    , , ;
34. , , тип – целочисленный, ,   
    , , ;
35. , , тип – вещественный, ,  
    , , .
36. Требуется написать программу, которая заполняет массив размерности n x n по заданному правилу: заполнение массива должно быть по диагонали, сверху-вниз, слева-направо. Причем заполнение еще и зигзагообразное. Например, для массива размерности 5 x 5 результат будет выглядеть следующим образом:

1 3 4 10 11

2 5 9 12 19

6 8 13 18 20

7 14 17 21 24

15 16 22 23 25.

1. В массиве 7 х 7 целого типа в каждом столбце упорядочить элементы, расположенные между максимальным и минимальным.
2. Дана матрица целых чисел размера m x n. Вычислить суммы элементов по строкам и по столбцам.
3. Транспонировать матрицу случайных чисел размерности 3 x 3.

Строки

1. Дан текст произвольной длины, оканчивающийся символом «#». Определить количество строк в тексте, каждая строка заканчивается символом перевода строки «\n».
2. Программа должна выполнять преобразования строки (длина строки 255 символов):
3. СТРОКА ЗАГЛАВНЫМИ БУКВАМИ;
4. строка в нижнем регистре;
5. С Заглавной Буквы (Первый Символ Каждого Слова В Строке);
6. пЕРВЫЙ сИМВОЛ в нИЖНЕМ рЕГИСТРЕ;
7. Как в предложении (с заглавной буквы).

Организовать в программе меню, через которое можно удобно выбирать любое действие. Программа должна выполняться пока пользователь не введет символ «f».

Чтобы преобразовать строчные буквы в прописные в С++ есть функция toupper. Для преобразования строки в нижний регистр используем функцию tolower. Остальные преобразования можно выполнить также с помощью функций библиотеки cctype.

1. Подсчитать количество повторений элементов, заданного множества символов, во введенной строке.
2. Поменять согласные буквы на гласные во введенной пользователем строке, а гласные – на соответствующий ASCII код, используя функции.
3. Написать программу, которая задает пользователю вопрос (вопрос полагает однозначный ответ). Проверить правильность ответа. Дать пользователю несколько подсказок и попыток ответить на вопрос. Если он угадал спросить у него имя, и вывести на экран поздравление, которое есть конкатенацией нескольких строк, дважды использовав его имя.
4. Дана строка символов. Признак конца строки – символ «/n» (переход на новую строку). Строка состоит из слов, которые отделены друг от друга пробелами. Вывести самое длинное слово и его порядковый номер.
5. Организовать ввод строки, каждое слово в строке отделяется от других слов пробелами, их может быть неограниченное количество. Найти самое короткое слово в строке.
6. Разработать программу, выполняющую обработку строки. Обработка строки должна осуществляться посимвольно (использование функций форматированного вывода scanf() и sscanf() не допускается). Найти сумму целых чисел, перечисленных в исходной строке через запятую. Ввод исходной строки осуществляется с клавиатуры.
7. Дана строка символов, которая обязательно заканчивается символом точки. Удалить из строки первые буквы каждого слова.
8. Из заданной строки удалить все цифры и переписать их в другую строку.
9. Определить самое длинное слово в строке.
10. Сократить число пробелов, разделяющих слова в строке до одного.
11. Имеется текстовая строка, которая содержит произвольное скобочное выражение (скобки (), [], или {}). Необходимо создать функцию check(), которая будет проверять это скобочное выражение на правильность. Например:

check(“y(x)”) -> true,

check(“[(]”) -> false,

check(“[{}]”) -> true,

check(“)(“) -> false,

check(“”) -> true,

check(“b([{}-()]{a})”) -> true.

1. Пользователь вводит строку символов и искомый символ. Посчитать, сколько раз он встречается в строке.

Динамические типы данных

Указатели

1. Изначально есть указатель на массив с одним элементом. Пользователь вводит число. Если оно больше 0 записываем его в массив. Далее пользователь вводит второе число, если оно больше 0, надо перевыделять память для 2-х элементов массива и записать в массив второе число и т.д., пока пользователь не введет отрицательное число.
2. Необходимо создать структуру, которая содержит элемент целочисленного типа строку. Объявить указатель типа структуры (объект структуры) и выделить память для хранения данных одной структуры. Предложить пользователю внести данные и записать их в элементы объекта структуры. Далее пользователю необходимо сделать выбор: внести еще данные (создать еще один объект структуры) либо выйти из программы. Если пользователь продолжает ввод – необходимо выделить новую память для указателя на объект структуры и дописать в нее введенные пользователем новые данные. Цикл выделения новой памяти продолжается, пока пользователь не выберет выход из программы.
3. Написать фрагмент программы, удаляющий i-й элемент из массива целых чисел размера N, . Фрагмент оформить в виде функции, массив передать в функцию через параметры.

Динамические массивы

1. Найти след матрицы введённой с клавиатуры матрицы (след матрицы – сумма элементов главной диагонали). Размер матрицы вводит пользователь, матрицу заполнять случайными числами.
2. Нужно вычислить произведение матрицы на вектор. В полученном векторе найти максимальный элемент. необходимо решить данную задачу тремя способами:
3. с использованием статических массивов;
4. использовать только динамические массивы с явным разыменованием указателя;
5. использовать только динамические массивы, адресацию к элементам массива выполнять с помощью индексов.
6. Составить программу, которая будет генерировать случайные числа в интервале и заполнять ими двумерный массив размером 10 на 10. В массиве необходимо найти номер строки с минимальным элементом. Поменять строки массива местами, строку с минимальным элементом и первую строку массива. Организовать удобный вывод на экран.
7. Задана матрица А размерностью n x m. Записать все элементы матрицы в одномерный массив.
8. Выполнить преобразование матрицы, а именно – перестановку строк и столбцов. Для квадратной матрицы размером n переставляйте столбцы и строки таким образом, чтобы элемент матрицы с наибольшим значением, по модулю, располагался в нижнем правом углу матрицы. Заполнение исходной матрицы организовать с клавиатуры.
9. Необходимо разработать программу, которая предоставляет удобный функционал для работы с разреженной матрицей. Разреженная матрица – матрица с большим количеством нулевых элементов. Требуется написать следующие функции:
10. Ввода матрицы;
11. Печати матрицы;
12. Суммирования двух матриц;
13. Умножения двух матриц.

Представлять матрицы можно в виде списка или в виде динамического массива.

1. Дана матрица. Найти сумму элементов в тех столбцах, которые не содержат отрицательные элементы. Количество столбцов и строк матрицы должен вводить пользователь. По введенным данным, динамически, должна выделиться память под матрицу. Матрицу заполнять случайными значениями, как отрицательными, так и положительными.
2. Разработать функцию обнуления случайного столбца в двумерном массиве. Обнулить 3 столбца. Результат вывести на печать.
3. Задание «Список группы». Составить список учебной группы, включающей 15 человек. Для каждого студента указать дату рождения, год поступления в институт, курс, группу, оценки каждого года обучения. Совокупность сведений о студентах объединить в массив.

Составить программу, которая обеспечивает ввод полученной информации, распечатку ее в виде таблицы, а также распечатку информации согласно конкретному варианту.

Варианты заданий:

1. Распечатать данные отличников.
2. Распечатать данные хорошистов.
3. Распечатать данные студентов, получивших одну тройку за все годы обучения.
4. Распечатать данные студентов, получивших в последнюю сессию двойки.
5. Распечатать данные студентов, получивших в первую сессию все пятерки.
6. Распечатать данные студентов, получивших в первую сессию все тройки.
7. Распечатать данные студентов, получивших за все время обучения одну четверку, а все остальные пятерки.
8. Распечатать список студентов, фамилии которых начинаются с буквы А, и их оценки за все время обучения.
9. Распечатать список отличников за все время обучения, в алфавитном порядке.
10. Распечатать список отличников за все время обучения и упорядочить по году рождения.
11. Распечатать список отличников за все время обучения и упорядочить по году поступления в институт.
12. Распечатать список студентов фамилия которых начинается с буквы А и упорядочить по году поступления в институт.
13. Распечатать список хорошистов за все время обучения и упорядочить по году рождения.
14. 1Распечатать список студентов, фамилии которых начинаются с буквы Б, и их даты рождения.
15. Распечатать оценки в последнюю сессию студентов, фамилии которых начинаются с букв В и Г.
16. Распечатать фамилии и даты рождения студентов, не получивших ни одной тройки за все время обучения.
17. Распечатать фамилии и даты рождения студентов, получивших все тройки за все время обучения.
18. Упорядочить список студентов по среднему балу и распечатать его.
19. Упорядочить список студентов по среднему балу последней сессии и распечатать его.
20. Вычислить средний балл группы и распечатать список студентов, имеющих средний балл выше среднего балла группы.
21. Вычислить средний балл группы и распечатать список студентов, имеющих средний балл ниже среднего балла группы.
22. Вычислить средний балл группы в последнюю сессию и распечатать список студентов, имеющих средний балл, равный среднему баллу группы (округлить до второго знака).
23. Упорядочить список студентов по году рождения и распечатать его.
24. Распечатать список студентов, упорядоченный по месяцу рождения.
25. Распечатать список студентов, упорядоченный по алфавиту.
26. Распечатать список отличников, упорядоченный по году рождения.
27. Распечатать список студентов, упорядоченный по дате рождения.
28. Найти минимальный и максимальный элементы двумерного массива действительных чисел.

Функции

Функции как подпрограммы

1. Пусть А, В, С, D – одномерные массивы длины N, составленные из элементов заданного типа. При этом массив А вводится с клавиатуры, массив В строится по формуле, массив С заполняется случайным образом значениями, принадлежащими отрезку , а массив D получается из этих массивов заданной линейной комбинацией. Ввести А, создать остальные массивы и распечатать все перечисленные массивы, используя функцию вывода массивов. При следующих условиях: ; тип float; ;  
   ; .
2. Пусть А, В, С, D, E – двумерные массивы размерности M x N, составленные из элементов заданного типа. При этом А вводится с клавиатуры, В строится по формуле, С заполняется случайным образом значениями, лежащими на отрезке , Е заполняется единицами, а D получается из перечисленных массивов заданной линейной комбинацией. Ввести А, создать все остальные массивы и распечатать все массивы, используя функцию вывода массива. При следующих условиях: ; ; тип integer; ; ; .
3. Описать функцию, находящую минимальное из двух вещественных чисел A и B. С помощью этой функции найти минимальные из пар чисел A и B, A и C, A и D, если даны числа A, B, C, D.
4. Описать функцию, записывающую в переменную A минимальное из значений A и B, а в переменную B – максимальное из этих значений. Используя четыре вызова этой процедуры, найти минимальное и максимальное из чисел A, B, C, D.
5. Даны действительные числа a, b, c. Получить .
6. Описать все необходимые функции для нахождения площади треугольника ABC по формуле Герона: , где – полупериметр. С помощью этой функции найти площади треугольников ABC, ABD, ACD, если даны координаты точек A, B, C, D.
7. Описать функцию, записывающую в переменную A минимальное из значений A и B, а в переменную B – максимальное из этих значений. Используя четыре вызова этой процедуры, найти минимальное и максимальное из чисел A, B, C, D.
8. Описать функцию, находящую сумму цифр S целого числа N. Используя эту процедуру, найти суммы цифр пяти данных чисел.
9. Описать функцию, находящую приближенное, с заданной точностью, значение функции . В сумме учитывать все слагаемые, большие заданной разрешенной погрешности eps. С помощью этой функции найти приближенное значение экспоненты для данного x с шестью различными точностями.
10. Описать функцию, находящую наибольший общий делитель (НОД) двух натуральных чисел A и B, используя алгоритм Евклида. С помощью этой функции найти наибольшие общие делители пар A и B, A и C, A и D, если даны числа A, B, C, D.
11. Даны действительные числа . Найти периметр десятиугольника, вершины которого имеют соответственно координаты . Определить и использовать функцию вычисления расстояния между двумя точками, заданными своими координатами.
12. Даны действительные числа . Получить , где u, w, v – комплексные числа , , . Определить процедуры выполнения арифметических операций над комплексными числами.
13. Описать все необходимые функции для нахождения высот hA, hB, hC треугольника ABC, проведенных соответственно из вершин A, B, C. С помощью этой функции найти высоты треугольников ABC, ABD, ACD, если даны координаты точек A, B, C, D.
14. Написать функцию, определяющую, есть ли в записи числа N цифра 3. Рассмотреть два случая: исходное число записывается посимвольно в строке; исходное число размещается в одной ячейке в формате int.
15. Разработать функцию, в которую передаются в качестве аргументов массив типа float и его размер. Функция должна возвращать среднее арифметическое элементов массива.
16. Описать функцию аddRightDigit(d,k), которая должна добавлять к целому положительному числу К справа цифру D (D – целочисленное значение в диапазоне 0–9, К – целочисленное значение, которое является одновременно входным параметром и модифицируемым значением). С помощью этой функции добавить к данному числу К цифры D1 и D2, выводя результат каждого добавления.
17. Написать функцию, подсчитывающую среднее арифметическое целого массива. Предварительно объявив и, при необходимости, проинициализировав переменные.
18. Написать функцию, которая принимает два целых числа n и k и возвращает число, содержащее k первых цифр числа n.
19. Написать функцию, которая будет возвращать значение true, если текст содержит символы только русского языка, и false при наличии недопустимых для русского текста символов, например, латинское «с» вместо русского «с».

Рекурсивные функции

1. Написать рекурсивную программу вывода на экран в виде:

1111111111111111

222222222222

33333333

4444

33333333

222222222222

1111111111111111

1. В выражении 12894\*4193\*9510\*8653\*4381\*2546\*1158\*8645\*2587 заменить звёздочки знаками «+» или «-» так, чтобы получившееся арифметическое выражение равнялось 1989.
2. В написанном выражении ((((1 ? 2) ? 3) ? 4) ? 5) ? 6 вместо каждого знака «?» вставить знак одного из четырёх арифметических действий: +, -, \*, / - так, чтобы результат вычислений равнялся 35 (при делении дробная часть в частном отбрасывается).
3. Вычислите наибольший общий делитель двух чисел, используя рекурсивную функцию.
4. Из данных предметов выбрать такие, чтобы суммарный вес был менее 30 кг, а стоимость – наибольшей.
5. Фермер хочет построить на своей земле как можно больший по площади сарай. Но на его участке есть деревья и хозяйственные постройки, которые он не хочет никуда переносить. Для простоты представим ферму сеткой размера M x N. Каждое из деревьев и построек размещается в одном или нескольких узлах сетки. Прямоугольный сарай не должен ни с чем соприкасаться (т.е. в соседних с ним узлах сетки не может ничего быть). Найти максимально возможную площадь сарая и место, где он может размещаться.
6. Даны последовательности x1, x2, …, xn и y1, y2, …, ym. Найти общую подпоследовательность наибольшей длины.
7. Задан двумерный массив . требуется найти такой путь из в , проходящий через соседние (в строке или столбце) элементы массива, ведущий вправо или вниз, чтобы сумма пройденных элементов была минимальной.
8. Написать функцию вычисления факториала числа N.
9. Написать функцию возведения в степень числа a в степень n за a рекурсивных вызовов.
10. Написать функцию возведения в степень числа a в степень n, основанный на рекуррентном соотношении:
11. Организовать рекурсивное вычисление числа Фибоначчи на позиции n (постановка задачи для реализации алгоритма на основе циклов представлена под номером 87).
12. Используя рекуррентное соотношение, для заданного x вычислить сумму с точностью :
13. Построить все правильные скобочные выражения (например, «((()())())») длины 10, т.е. те, которые содержат по 5 левых и 5 правых круглых скобок.
14. В данной последовательности действительных чисел a1, …, an выбрать возрастающую последовательность наибольшей длины.
15. Даны натуральные числа m, a1, …, an. В последовательности a1, …, an выбрать последовательность ai1, …, aik (0≤i1<…<ik≤n), такую что ai1+ …+ + aik=m. Если такую последовательность выбрать невозможно, то следует сообщить об этом.
16. Составить программу, которая печатает все различные представления числа N в виде всевозможных произведений K натуральных чисел (N, K вводятся так, что 1<K<N). Если К = 0, то выдать все возможные произведения. Представления чисел, отличающихся только порядком сомножителей, считаются одинаковыми.
17. Составить программу, которая печатает все различные представления числа N в виде всевозможных сумм K натуральных чисел (N, K – вводятся так, что 1 < K < N ). Если К = 0, то выдать все возможные суммы. Представления чисел, отличающихся только порядком слагаемых, считаются одинаковыми.
18. Данные N косточек домино по правилам игры выкладываются в прямую цепочку, начиная с косточки, выбранной произвольно, в оба конца до тех пор, пока это возможно. Построить алгоритм, позволяющий определить такой вариант выкладывания заданных косточек, при котором к моменту, когда цепочка не может быть продолжена, «на руках» останется максимальное число очков.
19. На столе в двух столбиках лежат 64 золотых и 64 серебряных монеты соответственно. Как серебряные, так и золотые монеты упорядочены в порядке убывания масс. Массы всех монет разные. За наименьшее количество взвешиваний необходимо определить 64-ую монету в порядке убывания масс среди всех 128 монет. За один раз можно взвешивать две монеты и определять, которая из них тяжелее.
20. Задано N наборов монет из различных стран. Наборы упорядочены по невозpастанию массы монет. В i-ом наборе ai монет. Необходимо за минимальное число взвешиваний на чашечных весах определить k-ую по массе монету среди всех монет.
21. На длинной перфоленте записаны N различных положительных целых чисел. Ваша ЭВМ может перематывать ленту на начало и считывать числа одно за другим. Внутренняя память машины может хранить только несколько целых чисел. Требуется найти наименьшее положительное целое число, которого нет на ленте. Сделайте это за небольшое количество перемоток ленты.
22. Обезьяна находится на лестнице из n ступенек. В руках у неё k орехов, которые она исследует на прочность, бросая их через перила лестницы на пол первого этажа. Считается, что если прочность скорлупы орехов равна Р, то:

* орехи раскалываются при сбрасывании с любой ступеньки с номером, большим Р;
* орехи не раскалываются при сбрасывании со ступенек до номера Р включительно.

Прочность Р изменяется в диапазоне 0 ... n. Требуется определить минимальное количество попыток, за которое обезьяна гарантировано сможет определить прочность орехов при заданных n и k, если будет действовать оптимальным способом.

1. Игра «Быки и Коровы». Компьютер «задумывает» четырехзначное число, не содержащее двух одинаковых цифр! Вы набираете свое число, и компьютер сообщает количество плюсов (точно угаданных цифр, т.е., стоящих на своих местах) и минусов (цифр, которые есть в задуманном числе, но на другом месте). Например, пусть задуманное число 5734, а вы набирали 0755. Результат будет 1 плюс и 2 минуса. Игра продолжается до тех пор, пока вы получите 4 плюса.
2. Имеется N городов. Для каждой пары городов можно построить дорогу, соединяющую эти два города и не заходящие в другие города. Стоимость такой дороги . Вне городов дороги не пересекаются. Написать алгоритм для нахождения самой дешевой системы дорог, позволяющей попасть из любого города в любой другой. Результаты задавать таблицей , где тогда и только тогда, когда дорогу, соединяющую города I и J, следует строить.
3. Вводится N – количество домов и К – количество дорог. Дома пронумерованы от 1 до N. Каждая дорога определяется тройкой чисел – двумя номерами домов – концов дороги и длиной дороги. В каждом доме живет по одному человеку. Найти точку – место встречи всех людей, от которой суммарное расстояние до всех домов будет минимальным. Если точка лежит на дороге, то указать номера домов - концов этой дороги и расстояние от первого из этих домов. Если точка совпадает с домом, то указать номер этого дома. Примечание: длины дорог – положительные целые числа.
4. Задача «Монахи». Для предотвращения утечки информации в монастыре Святого Павла в Калифорнии соблюдаются следующие правила:

* каждый монах известен миру только по своему личному номеру;
* у каждого монаха в течение жизни не может быть более трех учеников;
* у каждого монаха есть только один учитель.

Составить программу, которая сможет решить следующие задачи:

* по номеру монаха узнать, был ли такой монах и если был, то кто были его учитель, учитель его учителя и т.д. до самого Святого Павла;
* по двум монашеским номерам найти их общего ближайшего учителя.

Исходные данные о монахах представляют собой строки, в каждой из которых указан номер монаха и после него три номера его учеников. Если у монаха было менее трех учеников, то на соответствующих местах будут стоять нули.

Числовые данные для решения конкретной задачи – это строки, в каждой из которых стоит код строки и один или два монашеских номера. Если код строки равен единице, то за единицей будет стоять только один номер. Если код строки равен двум, то за двойкой будет стоять два номера.

Указания:

* монашеский номер Святого Павла равен единице;
* номера всех остальных монахов – целые числа от 2 до 600;
* номера монахов не обязательно последовательны, но номер ученика всегда больше, чем номер его учителя.

Подсказки:

* поиск учителей по номеру монаха рекомендуется делать с помощью рекурсии;
* для ввода исходных данных о монахах следует использовать двумерный массив 600 х 3, так чтобы, например, A[45,2] означало: 2-о1 ученик 45-ого монаха;
* для решения заданного варианта следует воспользоваться одномерным массивом из 600 элементов, так чтобы, например, A[45] означало: учитель 45-ого монаха.

Пример:

Исходные данные о монахах:

32 41 42 43

25 31 32 0

24 33 34 0

13 24 25 0

11 21 22 23

1 11 12 13

Вариант задания:

1 31

1 51

2 34 41

2 31 51

Результаты работы программы:

31 – монах, его учителя 25, 13;

51 – не монах;

34 и 41 – оба монахи, и их общий учитель 13;

51 – не монах.

1. Задача «Гипотеза Коллатца». С помощью рекурсивных формул задан закон образования числовой последовательности:

Задано целое число Max и значение x принадлежит отрезку [1; Max].

Найти n, для которого .

Например, для существует :

Указания:

* + - для вычисления использовать функцию;
    - для нахождения n использовать функцию от x и Max;
    - таблицы результатов печатать в два параллельных столбца с полями значений x и n.

Файлы

Числовые файлы

1. Записать в файл «NUMBERS.DAT» все четырехзначные натуральные числа:
2. в которых все цифры стоят по убыванию и не повторяются;
3. в которых цифра «7» встречается не менее трех раз;
4. в которых по крайней мере три цифры одинаковы;
5. которые слева направо и справа налево читаются одинаково;
6. в которых нет цифры «8»;
7. в которых все цифры стоят по возрастанию и не повторяются;
8. в которых цифра «0» встречается не менее двух раз;
9. составленные из неповторяющихся цифр;
10. в которых цифра «1» встречается не менее двух раз;
11. в которых хотя бы одна из цифр повторяется.

Текстовые файлы

1. Задание «Текстовый файл». Исходный файл содержит некоторое количество строк. Первый знак в строке определяет тип действия, которое нужно произвести со стоящими в строке целыми числами. Расшифровка знаков указана в табл. 2. Ввод и вывод осуществлять с помощью файлов.
2. – Соответствие знаков и действий над строкой.

|  |  |  |
| --- | --- | --- |
| ЗНАК | КОЛИЧЕСТВО ЧИСЕЛ В СТРОКЕ | ДЕЙСТВИЯ НАД ЧИСЛАМИ |
| + | Любое количество чисел, среди которых есть ноль | Просуммировать числа, стоящие в строке слева от нуля |
| \* | Два числа | Найти произведение этих чисел |
| / | Одно число | Найти число, обратное заданному |
| ! | Одно число | Найти факториал этого числа |
| М | Не менее пяти чисел | Найти наибольшие среди первых пяти чисел в строке |
| Х | Любое кол-во чисел | Найти порядковый номер наименьшего числа в строке |

1. Пусть в некоторый файл «A.TXT» построчно записаны предложения. Найти и распечатать:
2. количество восклицательных предложений;
3. предложение с наивысшим процентом гласных букв;
4. номера предложений, начинающихся со слова «Итак»;
5. предложение с наименьшим количеством знаков препинания;
6. количество вопросительных предложений;
7. предложение с наивысшим количеством гласных букв;
8. номера предложений, оканчивающихся многоточием;
9. предложения, содержащие двоеточия или тире;
10. первое восклицательное предложение;
11. предложение с наибольшим количеством знаков препинания.
12. Задача «Текстовый файл. Вариант 1». Создать текстовый файл, введя туда осмысленный текст на русском или английском (немецком, французском и т.п. языках).

Организовать вывод данных из файла по следующим правилам:

* в начале каждой строки напечатать номер строки;
* выводить символы из файла, печатая перед каждым из них его ASCII-код;
* после каждых десяти символов строки и при окончании строки исходного файла переходить на новую строку;
* в конце каждой строки исходного файла печатать «конец строки» и количество знаков в этой строке;
* при окончании исходного файла напечатать «конец файла»;
* после того, как выведен на экран весь файл, надо указать:

1. сколько символов было в исходном файле;
2. сколько строк было в исходном файле;
3. какой символ имел максимальный и какой – минимальный код (эти коды указать).

Наименьший из возможных кодов – 0, наибольший – 255. Всегда печатать код в поле величиной 3 позиции. Символ с кодом 32 – это пробел, которого не видно на экране.

1. Задача «Текстовый файл. Вариант 2». Напечатать содержимое текстового файла, выписывая литеры каждой его строки в обратном порядке.
2. Задача «Текстовый файл. Вариант 3». Содержимое текстового файла, разделённое на строки, переписать в другой текстовый файл, перенося при этом в конец каждой строки все входящие в неё цифры (с сохранением исходного взаимного порядка как среди цифр, так и среди остальных литер строки).
3. Задача «Окружности». В прямоугольной системе координат задаются координаты двух окружностей и 10 точек. Задача программы состоит в нахождении положения точек относительно окружностей. Принцип работы заключается в том, чтобы после ввода координат окружностей и точек программа выводит на экран координаты центра окружности, ее радиус и в виде таблицы 20 пронумерованных точек. Затем программа создает файл, в который записывает результаты исследования, и файл распечатывается на экране.
4. Создать внешний файл, содержащий сведения об игрушках: указывается: название игрушки, ее стоимость, возрастные границы (например, игрушка предназначается для детей от 2-х до 5-ти лет).

Получить следующие сведения:

1. названия игрушек, цена которых не превышает 400 р., и которые подходят детям 8 лет;
2. цену самой дорогой игрушки (игрушек);
3. названия и цену игрушек, которые подходят одновременно детям 4-х и 10-и лет.
4. Создать файл, содержащий сведения в библиотеке о книгах: ФИО автора, название, год издания. Данные вводить с клавиатуры.
5. найти название книги, автор и год издания которой вводятся вручную;
6. определить имеется ли книга, в названии которой есть слово «Паскаль». Если «да», то сообщить автора и год издания.

Обработка текстовых файлов

1. Записать в некоторый файл «GROUP.DAT» данные о студентах Вашей группы: ФИО, дата рождения, номер оконченной школы, сумма баллов на вступительных экзаменах в Университете, номер телефона. Считать информацию из файла в массив записей и упорядочить содержимое файла «GROUP.DAT»:
2. в алфавитном порядке по фамилиям студентов;
3. в порядке уменьшения возраста студентов;
4. в порядке возрастания номеров школ;
5. в порядке уменьшения суммы баллов на экзаменах;
6. по возрастанию номеров телефона;
7. в обратном алфавитном порядке по фамилиям студентов;
8. в порядке возрастания возраста студентов;
9. в порядке невозрастания номеров школ;
10. в порядке возрастания суммы баллов на экзаменах;
11. по убыванию номеров телефона.
12. Распечатать содержимое файла «GROUP.DAT» после обработки. Переписать в файл «GROUP1.DAT» данные о тех студентах, у которых:
13. четырехзначный номер оконченной школы;
14. фамилия начинается на гласную букву;
15. номер телефона начинается с девятки;
16. год рождения – 1980 или позже;
17. сумма баллов на экзаменах – от 21 до 30;
18. номер школы начинается с двойки;
19. фамилия начинается на одну из первых десяти букв;
20. номер телефона начинается с единицы или двойки;
21. год рождения – с 1979 по 1983;
22. четная сумма баллов на экзаменах.
23. Найти в файле «GROUP.DAT» и вывести на экран запись или записи по конкретному признаку, вводимому с клавиатуры:
24. фамилия студента;
25. первые три цифры номера телефона;
26. месяц рождения;
27. сумма баллов;
28. первая цифра номера телефона;
29. имя студента;
30. год рождения;
31. номер телефона;
32. первая цифра номера оконченной школы;
33. день рождения.
34. В текстовом файле, содержащем текст программы на языке Си, проверить соответствие открывающихся и закрывающихся фигурных скобок («{« и «}»). Результат проверки вывести на экран и записать в виде фразы в текстовый файл. Результат работы программы (вывод) поместить в отдельный текстовый файл (например, «out.txt»), продублировав результат на экране.
35. Дан текст, который пользователь должен ввести в программу, найти наибольшее количество идущих подряд букв. Реализовать следующие возможности в программе в виде функций:
36. меню пользователя, состоящее как минимум из 4-х пунктов:
37. ввод данных (текст вводится пользователем);
38. обработка данных (поиск наибольшего количества, подряд идущих, букв);
39. вывод результата на стандартное устройство вывода (экран) (с возможностью промежуточного хранения);
40. выход из программы;
41. организовать промежуточное хранение.
42. Написать программу, которая сортирует записи в файле в порядке возрастания номера телефона. В текстовом файле находятся записи о номерах телефонов. В каждой строке записана информация: номер телефона, ФИО, адрес.
43. Есть файл, в котором записан некоторый текст на английском языке. Требуется вывести в алфавитном порядке все буквы, которые встречаются в файле. Каждую букву выводить только один раз.
44. Программа «Жизнь» (описывающая жизнь микроорганизмов).

Введение. Жили-были микробы. Жили они долго и счастливо, но вот одна беда: после жизни всегда приходит смерть. А микробы были разные: маленькие и не очень, старички и малютки – и у каждого из них был свой жизненный уровень. Так, например, у только что родившегося микробчика он был равен 1, а по мере его взросления жизненный уровень тоже рос (от 1 до 12). Когда же микроб достигал последней ступени (т.е. 12), он, увы, погибал (теперь его жизненный уровень равен 0). Если же у микроба уровень был равен 0, то он рождался заново и проходил опять жизнь бодрым шагом от 1 до 12.

Всего же микробов в мире, как вы знаете, очень много, а жизнь их интересна. Поэтому появилась идея написать программу, описывающую их жизнь. Эта программа должна сообщать нам количество микробов в каждом поколении и «нарисовать» их на поле (экране).

Программа должна работать следующим образом.

1) Сначала создаётся файл work.dat (жилище микробов) и файл work.out (описывающий текущее поколение микробов, их развитие). Файл work.dat состоит из различных символов, среди которых «обитают» микробы.

2) Необходимо создать массивы «настоящее» и «будущее». В массиве «настоящее» записывается текущее поколение микробов, а в файл «будущее»- следующее. Массивы создаются размером 21х21.

3) Программа выдаёт пользователю на экран запрос: «Введите количество поколений». Именно столько поколений программа будет описывать.

4) Теперь программа создаёт в файле work.out поколение под номером 1. Для этого программа открывает и проверяет на наличие микробов файл (каждый символ, среди которых может затеряться микроб). Если под символом скрывается микроб (например, символ «Х»), то в массив «настоящее» записывается единица (1, т.е. микроб только что родился), а если не микроб – то ноль (0, т. е. там никто не живёт). Теперь в массиве «настоящее» находиться поле из 1 и 0 (он состоит из новорожденных младенцев и пустых мест). Все последующие поколения тоже записываются в файл work.out. Файл work.dat закрывается, и работа теперь ведётся только с файлом work.out.

5) Теперь описывается следующее поколение. Оно создаётся после проверки массива «настоящее». Проверяются микробы и их соседи (результат записывается в массив «будущее»):

Если жизненный уровень микроба от 1 до 11 то:

* если соседей 2 или 3, то микроб продолжает жить и подрастает (т.е. его жизненный уровень возрастает на 1),
* иначе микроб погибает (=0), т.к. он задыхается или умирает от скуки;
* если жизненный уровень микроба 0, то микроб рождается заново (т.е. его жизненный уровень равен 1);
* если жизненный уровень микроба равен 12, то микроб погибает (от старости).

6) После проверки подсчитывается количество жизней данного поколения (т.е. сколько единиц). Затем программа проверяет, есть ли кто «живой на поле» или все погибли (т.е. нули).

7) Теперь программа производит замену поколений: массив «будущее» становится «настоящим».

8) Программа продолжает работу с пункта 5 до тех пор, пока количество поколений (которые описывает программа) не станет равным введенным пользователем в пункте 3 или пока не погибнут все микробы (т.е. везде одни нули).

1. Эмуляция развития популяций животных. Задание: написать программу типа «Жизнь», но с некоторыми изменениями в начальных условиях.

Условия таковы, что в эмуляции должны участвовать две популяции: хищники и травоядные, которые взаимодействовали бы друг с другом путем поедания травоядных хищниками.

Дополнительные параметры:

* возраст животных;
* минимальный и максимальный репродуктивный возраст животных;
* количество пищи нужный животным для поддержания жизни;
* количество травы;
* процент восстановления травы;
* вероятность природных катаклизмов влияющих на популяции животных.

Методика взаимодействий хищника и травоядного заключается в том, что и хищники, и травоядные представлены в виде точек, которые передвигаются по экрану с шагом в один пиксель. При этом заданно условие: если в радиусе один пиксель от точки, принадлежащей хищнику, появляется точка, принадлежащая травоядному, то считается, что хищник съел травоядного.

Способ передвижения точек на экране организуется по алгоритму случайного блуждания, т.е. передвижение по осям Х и Y с шагом в один пиксель выбирается случайным образом.

Умершие своей смертью травоядные считаются съеденными хищниками.

При недоедании обеими популяциями, особи умирают в процессе увеличения возраста, т.е. чем больше возраст животного, тем больше вероятность погибнуть от голода. Из-за больших промежуточных расчетов учет по недоеданию выбирается так, что хищники учитываются один раз в год, а травоядные – двенадцать раз в год.

Программу построить на обработке массивов, имеющих следующие параметры: x – расположение по координате Х экрана,

* y – расположение по координате Y экрана,
* age – возраст точки,
* col – цвет вывода на экран.

Программа должна обеспечивать следующие операции: задание параметров популяции травоядных,

* задание параметров популяции хищников,
* задание параметров окружающей среды,
* просмотр взаимодействия животных в графическом режиме,
* индикация результатов по выходу из режима просмотра взаимодействия животных,
* выход из программы.

1. Задание «Список товаров». Данные о товаре заданы списком: название товара, цена товара, количество товара, год выпуска.

Написать программу, подготавливающую данные о N товарах, упорядоченных определенным образом по заданному признаку, и программу, добавляющую к ранее введенным товарам еще М товаров.

Общий список также должен быть упорядочен по указанному в варианте признаку. Если среди добавляемых товаров встречается товар, сведения о котором в файле уже есть, то необходимо их обновить, т.е. старую запись исключить.

Варианты заданий представлены в табл. 3.

1. – Варианты заданий к задаче «Список товаров».

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **№ вар.** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** |
| **F** | 1 | 2 | 3 | 4 | 5 | 6 | 1 | 2 | 3 | 4 | 5 | 6 | 1 | 2 |
| **N** | 10 | 9 | 11 | 9 | 7 | 12 | 13 | 10 | 13 | 12 | 11 | 10 | 7 | 8 |
| **M** | 3 | 4 | 4 | 6 | 3 | 5 | 4 | 3 | 6 | 4 | 3 | 3 | 5 | 5 |
| **№ вар.** | **15** | **16** | **17** | **18** | **19** | **20** | **21** | **22** | **23** | **24** | **25** | **26** | **27** | **28** |
| F | 3 | 4 | 5 | 6 | 1 | 2 | 3 | 4 | 5 | 6 | 1 | 2 | 3 | 4 |
| **N** | 7 | 8 | 9 | 8 | 8 | 11 | 11 | 10 | 9 | 11 | 9 | 7 | 12 | 13 |
| **M** | 6 | 5 | 4 | 6 | 3 | 5 | 5 | 4 | 3 | 6 | 4 | 5 | 3 | 3 |
| **№ вар.** | **29** | **30** | **31** | **32** | **33** | **34** | **35** | **36** | **37** | **38** | **39** | **40** | **41** | **42** |
| **F** | 3 | 4 | 5 | 6 | 1 | 2 | 3 | 4 | 5 | 6 | 1 | 2 | 3 | 4 |
| **N** | 10 | 13 | 12 | 11 | 10 | 8 | 7 | 6 | 7 | 8 | 9 | 10 | 9 | 9 |
| **M** | 4 | 4 | 6 | 2 | 2 | 4 | 3 | 5 | 6 | 2 | 4 | 3 | 2 | 2 |

В таблице указаны обозначения: F-принцип, по которому упорядочивается товар (1 – неубывание цены, 2 – неубывание года выпуска, 3 – неубывание количества, 4 – невозрастание цены, 5 – невозрастание года выпуска, 6 – невозрастание количества), N – количество товаров в первоначальном списке и M – количество добавляемых товаров.

1. Задача «Отдел кадров». Разработать программу, с помощью которой можно создать и частично корректировать базу данных, содержащую следующие сведения: фамилию, имя, отчество, должность, образование, стаж работы на предприятии, количество детей.

Для каждого из k сотрудников () заработная плата – массив Z(k) состоит из основной и надбавки. Числовые значения основной заработной платы Z0 задать в интервале от 30000 до 50000 рублей с помощью встроенного генератора случайных чисел.

Рассчитать для каждого сотрудника надбавку к зарплате Zd и полную заработную плату ZZ, которая определяется по формуле

,

,

где k – коэффициент надбавки, зависящий от должности и образования.

Увеличение этого коэффициента в зависимости от количества детей определяется по формуле (см. табл. 4):

,

где n – количество детей, n не более трех.

1. – Варианты заданий к задаче «Отдел кадров».

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Должность | Образование | | | |
| Высшее | Н/высшее | Ср. спец. | Н/ср. спец. |
| Руководитель | 3 | 2 | 1,8 | 0,5 |
| Секретарь | 1,3 | 1 | 0,5 | 0 |
| Инженер | 1,5 | 1 | 0,2 | 0 |
| Экономист | 1,8 | 1 | 0,8 | 0,1 |
| Гл. бухгалтер | 2 | 1,5 | 1 | 0,3 |

1. Организовать файл 1 компонентами которого являются 10 целочисленных одномерных массивов.
2. Максимальные и минимальные элементы всех массивов заменить на нули. Полученные массивы сохранить в файл 2.
3. Каждый массив преобразовать в квадратную матрицу размера 10 x 10. Если количество элементов недостаточно, то добавить их с использованием генератора случайных чисел. Результат сохранить в файл 3.
4. Иметь возможность просмотра полученных файлов.
5. Организовать файл, элементами которого являются слова. Упорядочить в нем слова по алфавиту. Добавить в файл произвольное слово с сохранением сортировки в файле.
6. Создать файл, элементами которого являются 5 целочисленных матриц m x n. Для каждой матрицы вывести номер строки и номер столбца для элемента матрицы, который одновременно является наименьшим в своей строке и наибольшим в своем столбце.
7. Создать файл из натуральных чисел. В файле натуральных чисел найти наименьший и наибольший общие делители, также определить все простые числа и их количество. Все простые числа сохранить в другой файл. Предусмотреть возможность просмотра содержимого всех файлов.
8. Создать файл, элементами которого являются 10 целочисленных матриц m x n. Определить для каждой матрицы номера строк столбцов всех одинаковых элементов, их значения. Полученные результаты для каждой матрицы сохранить в другом файле.
9. В файле организовать создание двух квадратных матрицы А и В порядка N. Получить матрицу , где Е – единичная матрица порядка N, а элементы матрицы С вычисляются по формуле:

Все созданные и вычисленные матрицы также сохранить в файле. Все матрицы вывести на экран в порядке заполнения и выполнения действий.

1. Создать два файла А и В. Компонентами файлов являются целые числа, которые следует упорядочить по возрастанию. Объединить содержимое файлов в новый файл С с сохранением сортировки всех элементов.
2. Ввести с клавиатуры фамилии студентов и их шифры, сохраняя информацию в файле. Упорядочить данные по фамилии или по шифру в зависимости от пожелания пользователя. После ввода данных иметь возможность просмотреть введенную информацию. Программу желательно реализовать с использованием подпрограмм(ы).
3. Ввести в файл 4 целочисленных матрицы порядка 8 х 8. В другом файле получить эти матрицы в транспонированном виде. В третьем файле для каждой матрицы найти сумму элементов в каждой строке и отсортировать каждую матрицу по уменьшению сумм элементов в строке.

Создать файл из т вещественных чисел. Найти длину k самой длинной «пилообразной (зубьями вверх)» последовательности идущих подряд чисел: .

1. Создать файл из N целых чисел. Найти число, повторяющееся максимальное количество раз. Если таких чисел несколько, то все из них. Сохранить эти числа в другой файл и отсортировать. Все файлы до и после обработки вывести на печать.
2. Создать файл из N целых чисел. Найти отрезок массива максимальной длины, в котором первое число равно последнему, второе – предпоследнему и т. д. Напечатать длину этого отрезка и весь файл.
3. Создать файл из N вещественных чисел в десятичной системе счисления. В другой файл записать эти числа, переведенные из исходного файла в любую систему счисления по желанию пользователя в пределах от 2-ой до 9-ой.
4. Создать файл, в котором хранятся записи целых чисел в шестнадцатеричной системе счисления. Организовать перевод чисел исходного файла в любую систему счисления по желанию пользователя в пределах от 2-ой до 9-ой. Результаты сохранить в другом файле.
5. Создать файл из n из вещественных чисел. Отсортировать его элементы по сумме цифр дробной части. Результат сохранить в другом файле. Программу написать с использованием подпрограмм(ы).
6. Написать программу, которая создаст файл phone.txt с информацией: фамилия и номер телефона нескольких ваших товарищей. Программа должна запрашивать фамилию человека и выводить его телефон. Если в справочнике есть одинаковые фамилии, то программа должна вывести список всех людей, имеющих эти фамилии. В другом файле организовать отсортированные по фамилиям данные исходного файла.
7. Написать программу, которая создаст файл phone.txt с информацией с данными: фамилия и номер телефона нескольких ваших товарищей. Рекомендуемый вид экрана во время работы программы приведен ниже. Вывести все данные товарищей, у которых в телефонный номер «счастливый» т.е. сумма цифр левой и правой частей равен (без учета симметричной позиции). Сохранить эти данные в другой файл и отсортировать их по фамилии.
8. Написать программу, которая создаст файл phone.txt с информацией с данными: фамилия и номер телефона нескольких ваших товарищей. Рекомендуемый вид экрана во время работы программы приведен ниже. Вывести все данные товарищей, у которых в телефонных номерах все цифры четные. Сохранить эти данные в другой файл и отсортировать по номерам телефонов.
9. Написать программу подведения итогов Олимпийских игр. В программу пользователь должен ввести количество медалей разного достоинства, завоеванное каждой командой-участницей, а программа – подсчитать общее число медалей и соответствующее число очков, после чего упорядочить список в соответствии с набранным количеством очков, которое определяется по следующему правилу: за золотую медаль команда получает 7 очков, за серебряную – 6, за бронзовую – 5.

Вся информация должна быть сохранена в файле, где одной записью будут являться все данные по одной стране.

Вся информация должна быть сохранена в файле, где одной записью будут являться все данные по одной стране.

Итоги Олимпийских игр: количество золотых, серебряных и бронзовых медалей могут быть представлены в следующей форме на основе табл. 5:

Австрия -> 3 5 9

Германия -> 12 9 8

Канада -> 6 5 4

Китай -> 0 6 2

Корея -> 3 1 2

Норвегия -> 10 10 5

Россия -> 9 6 3

США -> 6 3 4

Финляндия -> 2 4 6

Япония -> 5 1 4.

1. – Пример исходных данных результатов Олимпийских игр.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Cтрана | Золото | Серебро | Бронза | Всего | Очков |
| 1 | Германия | 12 | 9 | 8 |  |  |
| 2 | Новергия | 10 | 10 | 5 |  |  |
| 3 | Россия | 9 | 6 | 3 |  |  |
| 4 | Австрия | 3 | 5 | 9 |  |  |
| 5 | Канада | 6 | 5 | 4 |  |  |
| 6 | США | 6 | 3 | 4 |  |  |
| 7 | Финлядия | 2 | 4 | 6 |  |  |
| 8 | Япония | 5 | 1 | 4 |  |  |
| 9 | Китай | 0 | 6 | 2 |  |  |
| 10 | Корея | 3 | 1 | 2 |  |  |

Варианты заданий:

1. Отсортировать в файле страны по общему количеству набранных очков и вывести всю информацию на экран.
2. Отсортировать файл по количеству золотых медалей и вывести всю информацию на экран.
3. Отсортировать файл по сумме количества золотых и серебряных медалей и вывести всю информацию на экран.
4. Отсортировать файл по странам в алфавитном порядке и вывести всю информацию на экран
5. Организовать создание текстового файла. Подсчитать в текстовом файле число непустых строк, в которых символы упорядочены по возрастанию.
6. Создать файл, данными которого являются: номер зачетной книжки, ФИО студента, список из 5 предметов с оценками в сессии. Варианты заданий:
7. Отсортировать файл по среднему баллу каждого студента в сессии
8. Отсортировать файл по фамилиям студентов.
9. Отсортировать файл по результат второго экзамена.
10. Подсчитать среднюю успеваемость группы и вывести список всех студентов, у которых личный средний балл выше среднего балла группы.
11. Сохранить список отличников и хорошистов в отдельном файле и вывести их список с результатами сессии, отсортированный по фамилиям студентов.
12. Дана строка S из n символов, в которой символом «пробел» разделяются слова. Вывести на экран 3 слово в обратном порядке.
13. Даны целые числа с1, с2, …, с95. Подсчитать количество трех идущих подряд отрицательных чисел.
14. Найти сумму цифр целого числа n (водится с клавиатуры).
15. В строке символов записать (вывести на экран) в обратном порядке пятое слово, если оно присутствует в строке.
16. Определить, сколько различных цифр входят в запись целого числа n, которое водится с клавиатуры.
17. Дано целое число n (вводится с клавиатуры). Определить, входит ли цифра 3 в запись числа n2.
18. Дано натуральное число x. Выбросить из записи числа х цифры 0 и 5, оставив прежним порядок остальных цифр. Например, из числа 590155069 должно получиться 9169.
19. Даны числа с1, с2, …, с15. Подсчитать количество всех попарных сочетаний чисел, сумма которых образует значение «13» (предусмотреть допустимые случаи с11 + с12 =13, с7 + с12,=13).
20. Дана строка S из n символов. Определить, содержатся ли в ней все символы (в произвольном порядке), входящие в слово «студенчество».
21. Дана строка S из n символов. Подсчитать максимальное число подряд идущих пробелов.
22. Дана строка S из n символов, в которой символом «пробел» разделяются слова. Определить длину самого короткого слова.
23. Дана строка S из n символов, в которой символом «пробел» разделяются слова. Подсчитать количество слов, начинающихся с буквы «к» и заканчивающихся буквой «н».
24. Дано натуральное число m (). Получить все трехзначные целые числа, сумма цифр которых равна m. (указание: использовать полный перебор).
25. Написать программу, которая подсчитывает количество возрастающих последовательностей во введенном с клавиатуры массиве чисел.
26. Написать программу, которая подсчитывает количество убывающих последовательностей в введенном с клавиатуры массиве чисел.
27. Написать программу, которая вычисляет, сколько раз введенное с клавиатуры число встречается в массиве и в каких позициях.
28. Написать программу, которая проверяет, есть ли во введенном с клавиатуры массиве элементы с одинаковым значением. Если «ДА», то посчитать их количество, определить их положение в массиве и отсортировать в отдельном массиве.
29. Написать программу, которая объединяет два упорядоченных по возрастанию массива в один, также упорядоченный по возрастанию массив.
30. Написать программу, которая определяет количество учеников в классе, чей рост превышает средний. Найти количество пар учеников с одинаковым ростом.
31. Написать программу, которая вводит по строкам с клавиатуры двумерный массив и вычисляет сумму его элементов по столбцам.
32. Написать программу, которая вводит по строкам с клавиатуры двумерный массив и вычисляет сумму его элементов по строкам.
33. Написать программу, которая вычисляет сумму диагональных элементов квадратной матрицы.
34. Написать программу, которая вводит с клавиатуры двумерный массив по строкам и вычисляет среднее арифметическое его элементов в каждой строке и сортирует строки согласно уменьшения средних по строкам.
35. Написать программу, которая проверяет, является ли введенная с клавиатуры квадратная матрица магическим квадратом. Магическим квадратом называется матрица, сумма элементов которой в каждой строке, в каждом столбце и по каждой диагонали одинакова.
36. Написать программу, которая вводит по строкам с клавиатуры двумерный массив и вычисляет сумму его элементов по столбцам.
37. Написать программу, которая вычисляет определитель квадратной матрицы второго порядка. Рекомендуемый вид экрана во время работы программы приведен далее:

Введите матрицу второго порядка.

После ввода элементов строки нажимайте <Enter>

-> 5 -7

-> 1 3

Определитель матрицы

5.00 -7.00

1.00 3.00 Равен 22.00

1. Написать программу, которая вводит с клавиатуры последовательность из пяти дробных чисел и после ввода каждого числа выводит среднее арифметическое полученной части последовательности. Рекомендуемый вид экрана во время работы программы приведен далее:

Обработка последовательности дробных чисел. После ввода каждого числа нажимайте

<Enter> -> 12.3

Введено чисел: 1 Сумма: 12.30 Сред. арифметическое: 12.30

-> 15

Введено чисел: 2 Сумма: 27.30 Сред. арифметическое: 13.65

-> 10

Введено чисел: 3 Сумма: 37.30 Сред. арифметическое: 12.43

-> 5.6

Введено чисел: 4 Сумма: 42.90 Сред. арифметическое: 10.73

-> 11.5

Введено чисел: 5 Сумма: 54.40 Сред. арифметическое: 10.88

Для завершения работы программы нажмите <Enter>.

1. Написать программу, которая вычисляет среднее арифметическое последовательности дробных чисел, вводимых с клавиатуры. После того, как будет введено последнее число, программа должна вывести минимальное и максимальное число последовательности. Количество чисел должно задаваться во время работы программы. Рекомендуемый вид экрана приведен ниже. Данные, введенные пользователем, выделены полужирным шрифтом. Пример работы программы приведён далее:

Обработка последовательности дробных чисел. Введите количество чисел последовательности -> 5

Вводите последовательность. После ввода каждого числа нажимайте <Enter> -> 5.4 -> 7.8

-> 3.0 -> 1.5 -> 2.3

Количество чисел: 5

Среднее арифметическое: 4.00

Минимальное число:

Максимальное число:

Для завершения нажмите <Enter>

1. Дана символьная матрица N x N. найти номер последнего по порядку столбца, содержащего наименьшее число букв Ш и Щ.
2. Ввести с клавиатуры массив строк. Отсортировать его по возрастанию количества слов в каждой строке.
3. Создать массив, содержащий сведения в библиотеке о книгах: ФИО автора, название, год издания. Данные вводить с клавиатуры.
4. Найти название книги, автор и год издания которой вводятся вручную;
5. Определить имеется ли книга, в названии которой есть слово «Паскаль».

Если «да», то сообщить автора и год издания.

1. Ввести 2 массива. Объединить эти 2 массива в один с сохранением упорядоченности по возрастанию.
2. Ввести целочисленную матрицу m x n. вывести номер строки и номер столбца для элемента матрицы, который одновременно является наименьшим в своей строке и наибольшим в своем столбце.
3. Дано натуральное N, действительные .

Получить:

,

где

,

,

.

1. Желательно программу реализовать с использованием подпрограмм(ы).
2. В массиве натуральных чисел найти наименьший и наибольший общие делители.
3. Ввести целочисленную матрицу m x n. Определить номера строк столбцов всех одинаковых элементов, их значения. Подсчитать количество элементов, лежащих в диапазоне от 5 до 9 включительно.
4. В целочисленном массиве найти число, повторяющееся максимальное количество раз. Если таких чисел несколько, то одно из них.
5. Задан числовой массив . Найти отрезок массива максимальной длины, в котором первое число равно последнему, второе – предпоследнему и т. д. Напечатать длину этого отрезка.
6. Задан массив из вещественных чисел в десятичной системе счисления. В другом массиве организовать перевод исходного массива в любую систему счисления по желанию пользователя в пределах от 2-ой до 9-ой.
7. Задан массив из чисел в двоичной системе счисления. В другом массиве организовать перевод исходного массива в любую систему счисления по желанию пользователя в пределах от 3-ой до 6-ой.
8. Задан массив из вещественных чисел. Отсортировать его элементы по сумме цифр дробной части. Программу написать с использованием подпрограмм(ы).
9. Задан массив из чисел в двоичной системе счисления. В другом массиве организовать перевод исходного массива в 4 систему счисления.
10. Задан массив из чисел в двоичной системе счисления. В другом массиве организовать перевод исходного массива в 3 систему счисления.
11. Задан массив из чисел в двоичной системе счисления. В другом массиве организовать перевод исходного массива в 5 систему счисления.
12. Задан массив из чисел в троичной системе счисления. В другом массиве организовать перевод исходного массива в 6 систему счисления.
13. Задан массив из чисел в семеричной системе счисления. В другом массиве организовать перевод исходного массива в 3 систему счисления.
14. Задан массив из чисел в пятеричной системе счисления. В другом массиве организовать перевод исходного массива в 2 систему счисления.
15. Преобразование разделителей: замена пробелов на другие(определенные) символы.
16. Преобразование разделителей: превращение строк в столбцы слов.
17. Преобразование разделителей: превращение строк в столбцы фраз.
18. Поиск в тексте слова максимальной длины.
19. Поиск в тексте фразы максимальной длины.
20. Поиск в тексте слова минимальной длины.
21. Поиск в тексте фразы минимальной длины.
22. Преобразование текста в цепочку ASCII-кодов.
23. Преобразование текста в столбец ASCII-кодов.
24. Обработка текста по слогам: вставка разделителей между слогами.
25. Обработка текста по словам: вставка разделителей между словами.
26. Чтение текста из текстового файла.
27. Запись текста в текстовый файл.
28. Добавление текста в конец текстового файла.
29. Поиск определенного слова в текстовом файле.
30. Поиск определенного сочетания слов в текстовом файле.
31. Сортировка слов в текстовом файле по алфавиту.
32. Поиск вхождения подстроки в строку текстового файла.
33. Статистическая обработка текстового файла: поиск наиболее часто встречающегося символа.
34. Статистическая обработка текстового файла: поиск наименее часто встречающегося символа.
35. Статистическая обработка текстового файла: поиск наиболее часто встречающейся гласной буквы.
36. Статистическая обработка текстового файла: поиск наименее часто встречающейся гласной буквы.
37. Статистическая обработка текстового файла: поиск наиболее часто встречающейся согласной буквы.
38. Статистическая обработка текстового файла: поиск наименее часто встречающейся согласной буквы.
39. Поиск подстроки в строке по заданному условию.
40. Посимвольная замена элементов текстового файла.
41. Вычисление частоты повтора символа в текстовом файле.
42. Преобразование текста в массив символов.
43. Посимвольная сортировка.
44. Шифрование текстового файла заменой символов.
45. Шифрование текстового файла перестановкой символов.
46. Шифрование текстового файла сдвигом в алфавите.
47. Поиск элемента в текстовом файле по заданному условию.
48. Поиск определенных словосочетаний в текстовом файле.
49. Составление словаря для слов текстового файла.
50. Исключение из текстового файла заданных символов.
51. Поиск вхождения цифр в текстовом файле.
52. Проверка баланса скобок в текстовом файле.
53. Подсчет числа вхождений символов в текстовый файл.
54. Расположение слов текстового файла в алфавитном порядке.

Список литературы

1. Прата С. Язык программирования С. Лекции и упражнения: Учебник / Пер. с англ. – СПб.: ООО «ДиаСофтЮП», 2002. – 896 с.
2. Васильев А.Н. Самоучитель С++ с примерами и задачами. – СПб.: Наука и техника, 2010. – 480 с.
3. Культин Н.Б. Основы программирования в Microsoft Visual C++ 2010. Самоучитель. – СПб.: БХВ-Петербург, 2010. – 384с.
4. Павловская Т.А., Щупак Ю.А. С/С++. Структурное и объектно-ориентированное программирование: Практикум: Учебное пособие. – СПб.: Питер, 2010. - 352 с.
5. Культин Н.Б. Microsoft Visual C++ в примерах и задачах. – СПб.: БХВ-Петербург, 2011. – 272 с.
6. Подбельский В.В., Фомин С.С. Курс программирования на языке Си: Учебник. – М.: ДМК Пресс, 2012. – 384 с.
7. Магда Ю.С. Программирование и отладка С/С++ приложений для микроконтроллеров АРМ. – М.: ДМК Пресс, 2012. – 168 с.
8. Основы программирования на языках Си и C++ для начинающих [Электронный ресурс]. Режим доступа: http://cppstudio.com/.
9. Основы программирования на С++ для начинающих [Электронный ресурс]. Режим доступа: https://purecodecpp.com/.
10. Портал о программировании начинающих [Электронный ресурс]. Режим доступа: https://code-live.ru/.
11. Клуб программистов [Электронный ресурс]. Режим доступа: http://www.programmersclub.ru/.
12. SoloLearn Inc. [Электронный ресурс]. Режим доступа: https://www.sololearn.com/Course/CPlusPlus/.
13. Программирование C и C++ [Электронный ресурс]. Режим доступа: https://prog-cpp.ru/.

Сведения об авторах

Каширская Елизавета Натановна, кандидат технических наук, доцент, доцент, кафедра промышленной информатики, институт информационных технологий.

Антонов Сергей Валерьевич, ассистент, кафедра промышленной информатики, институт информационных технологий.