

Сборник задач

по программированию

к авторской дополнительной образовательной программе

«Системное и прикладное программирование»

Тольятти

2015

Оглавление

1.	Линейные алгоритмы	4
2.	Логические выражения	7
3.	Условные операторы	11
4.	Оператор выбора	15
5.	Операторы цикла	19
6.	Обработка последовательностей	23
7.	Минимумы и максимумы	27
8.	Одномерные массивы	28
9.	Двумерные массивы (матрицы)	33
10.	Символы и строки	37
11.	Процедуры и функции (вычисление)	43
12.	Процедуры и функции (массивы)	47
13.	Процедуры и функции (строки)	51
14.	Процедуры и функции (файлы)	55
15.	Процедуры и функции (простейшие рекурсивные алгоритмы)	58
16.	Процедуры и функции (разбор выражений)	61
17.	Процедуры и функции (деревья)	64
18.	Двоичные файлы	66
19.	Текстовые файлы	74

1. Линейные алгоритмы

- 1.1. Даны два ненулевых числа. Найти их сумму, разность, произведение и частное.
- 1.2. Даны два числа. Найти среднее арифметическое их квадратов и среднее арифметическое их модулей.
- 1.3. Скорость лодки в стоячей воде V км/ч, скорость течения реки U км/ч ($U < V$). Время движения лодки по озеру T_1 ч, а по реке (против течения) — T_2 ч. Определить путь S , пройденный лодкой.
- 1.4. Скорость первого автомобиля V_1 км/ч, второго — V_2 км/ч, расстояние между ними S км. Определить расстояние между ними через T часов, если автомобили удаляются друг от друга.
- 1.5. Скорость первого автомобиля V_1 км/ч, второго — V_2 км/ч, расстояние между ними S км. Определить расстояние между ними через T часов, если автомобили первоначально движутся навстречу друг другу.
- 1.6. Найти периметр и площадь прямоугольного треугольника, если даны длины его катетов a и b .
- 1.7. Дана длина ребра куба. Найти площадь грани, площадь полной поверхности и объем этого куба.

- 1.8. Найти длину окружности и площадь круга заданного радиуса R . В качестве значения P_i использовать 3.14.
- 1.9. Найти площадь кольца, внутренний радиус которого равен R_1 , а внешний радиус равен R_2 ($R_1 < R_2$). В качестве значения P_i использовать 3.14.
- 1.10. Дана сторона равностороннего треугольника. Найти площадь этого треугольника и радиусы вписанной и описанной окружностей.
- 1.11. Дана длина окружности. Найти площадь круга, ограниченного этой окружностью. В качестве значения P_i использовать 3.14.
- 1.12. Дана площадь круга. Найти длину окружности, ограничивающей этот круг. В качестве значения P_i использовать 3.14.
- 1.13. Найти периметр и площадь равнобедренной трапеции с основаниями a и b ($a > b$) и углом α при большем основании (угол дан в радианах).
- 1.14. Найти периметр и площадь прямоугольной трапеции с основаниями a и b ($a > b$) и острым углом α (угол дан в радианах).
- 1.15. Найти расстояние между двумя точками с заданными координатами (x_1, y_1) и (x_2, y_2) .
- 1.16. Даны координаты трех вершин треугольника (x_1, y_1) , (x_2, y_2) , (x_3, y_3) . Найти его периметр и площадь.

- 1.17. Найти корни квадратного уравнения $Ax^2 + Bx + C = 0$, заданного своими коэффициентами A , B , C (коэффициент A не равен 0), если известно, что дискриминант уравнения неотрицателен.
- 1.18. Найти решение системы уравнений вида $A_1x + B_1y = C_1$, $A_2x + B_2y = C_2$, заданной своими коэффициентами A_1 , B_1 , C_1 , A_2 , B_2 , C_2 , если известно, что данная система имеет единственное решение.
- 1.19. Дано целое четырехзначное число. Используя операции `div` и `mod`, найти сумму его цифр.
- 1.20. Дано целое четырехзначное число. Используя операции `div` и `mod`, найти произведение его цифр.

2. Логические выражения

Во всех заданиях данного пункта требуется вывести логическое значение True, если приведенное высказывание для предложенных исходных данных является истинным, и значение False в противном случае. Все числа, для которых указано количество цифр (двузначное число, трехзначное число и т.д.), считаются целыми.

- 2.1. Проверить истинность высказывания: "Квадратное уравнение $Ax^2 + Bx + C = 0$ с данными коэффициентами A, B, C имеет вещественные корни".
- 2.2. Проверить истинность высказывания: "Данные числа x, y являются координатами точки, лежащей во второй координатной четверти".
- 2.3. Проверить истинность высказывания: "Данные числа x, y являются координатами точки, лежащей в первой или третьей координатной четверти".
- 2.4. Проверить истинность высказывания: "Точка с координатами (x, y) лежит внутри прямоугольника, левая верхняя вершина которого имеет координаты (x_1, y_1) , правая нижняя — (x_2, y_2) , а стороны параллельны координатным осям".
- 2.5. Проверить истинность высказывания: "Данное целое число является четным двузначным числом".
- 2.6. Проверить истинность высказывания: "Данное целое число является нечетным трехзначным числом".

- 2.7. Проверить истинность высказывания: "Среди трех данных целых чисел есть хотя бы одна пара совпадающих".
- 2.8. Проверить истинность высказывания: "Среди трех данных целых чисел есть хотя бы одна пара взаимно противоположных".
- 2.9. Проверить истинность высказывания: "Сумма цифр данного трехзначного числа является четным числом".
- 2.10. Проверить истинность высказывания: "Сумма двух первых цифр данного четырехзначного числа равна сумме двух его последних цифр".
- 2.11. Проверить истинность высказывания: "Данное четырехзначное число читается одинаково слева направо и справа налево".
- 2.12. Проверить истинность высказывания: "Все цифры данного трехзначного числа различны".
- 2.13. Проверить истинность высказывания: "Цифры данного трехзначного числа образуют возрастающую последовательность".
- 2.14. Проверить истинность высказывания: "Цифры данного трехзначного числа образуют возрастающую или убывающую последовательность".

- 2.15. Проверить истинность высказывания: "Цифры данного трехзначного числа образуют арифметическую прогрессию".
- 2.16. Проверить истинность высказывания: "Цифры данного трехзначного числа образуют геометрическую прогрессию".
- 2.17. Даны координаты (как целые от 1 до 8) двух различных полей шахматной доски. Если ладья за один ход может перейти с одного поля на другое, вывести логическое значение True, в противном случае вывести значение False.
- 2.18. Даны координаты (как целые от 1 до 8) двух различных полей шахматной доски. Если король за один ход может перейти с одного поля на другое, вывести логическое значение True, в противном случае вывести значение False.
- 2.19. Даны координаты (как целые от 1 до 8) двух различных полей шахматной доски. Если слон за один ход может перейти с одного поля на другое, вывести логическое значение True, в противном случае вывести значение False.
- 2.20. Даны координаты (как целые от 1 до 8) двух различных полей шахматной доски. Если ферзь за один ход может перейти с одного поля на другое,

вывести логическое значение True, в противном случае вывести значение False.

2.21. Даны координаты (как целые от 1 до 8) двух различных полей шахматной доски. Если конь за один ход может перейти с одного поля на другое, вывести логическое значение True, в противном случае вывести значение False.

3. Условные операторы

- 3.1. Даны три целых числа. Возвести в квадрат отрицательные числа и в третью степень — положительные (число 0 не изменять).
- 3.2. Из трех данных чисел выбрать наименьшее.
- 3.3. Из трех данных чисел выбрать наибольшее.
- 3.4. Из трех данных чисел выбрать наименьшее и наибольшее.
- 3.5. Перераспределить значения переменных X и Y так, чтобы в X оказалось меньшее из этих значений, а в Y — большее.
- 3.6. Значения переменных X , Y , Z поменять местами так, чтобы они оказались упорядоченными по возрастанию.
- 3.7. Значения переменных X , Y , Z поменять местами так, чтобы они оказались упорядоченными по убыванию.
- 3.8. Даны две переменные целого типа: A и B . Если их значения не равны, то присвоить каждой переменной сумму этих значений, а если равны, то присвоить переменным нулевые значения.
- 3.9. Даны две переменные целого типа: A и B . Если их значения не равны, то присвоить каждой переменной

максимальное из этих значений, а если равны, то присвоить переменным нулевые значения.

- 3.10. Даны три переменные: X , Y , Z . Если их значения упорядочены по убыванию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное.
- 3.11. Даны три переменные: X , Y , Z . Если их значения упорядочены по возрастанию или убыванию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное.
- 3.12. Даны целочисленные координаты точки на плоскости. Если точка не лежит на координатных осях, то вывести 0. Если точка совпадает с началом координат, то вывести 1. Если точка не совпадает с началом координат, но лежит на оси OX или OY , то вывести соответственно 2 или 3.
- 3.13. Даны вещественные координаты точки, не лежащей на координатных осях OX и OY . Вывести номер координатной четверти, в которой находится данная точка.
- 3.14. На числовой оси расположены три точки: A , B , C . Определить, какая из двух последних точек (B или C) расположена ближе к A , и вывести эту точку и ее расстояние от точки A .

- 3.15. Даны четыре целых числа, одно из которых отлично от трех других, равных между собой. Вывести порядковый номер этого числа.
- 3.16. Дан номер некоторого года (положительное целое число). Вывести соответствующий ему номер столетия, учитывая, что, к примеру, началом 20 столетия был 1901 год.
- 3.17. Дан номер некоторого года (положительное целое число). Вывести число дней в этом году, учитывая, что обычный год насчитывает 365 дней, а високосный — 366 дней. Високосным считается год, делящийся на 4, за исключением тех годов, которые делятся на 100 и не делятся на 400 (например, годы 300, 1300 и 1900 не являются високосными, а 1200 и 2000 — являются).
- 3.18. Для данного x вычислить значение следующей функции f . $f(x) = -1$, если $x \leq 0$, $f(x) = x$, если $0 < x < 2$, $f(x) = x^2$, если $x \geq 2$.
- 3.19. Дано целое число, лежащее в диапазоне от -999 до 999 . Вывести строку — словесное описание данного числа вида "отрицательное двузначное число", "нулевое число", "положительное однозначное число" и т.д.

3.20. Дано целое число, лежащее в диапазоне от 1 до 9999. Вывести строку — словесное описание данного числа вида "четное двузначное число", "нечетное четырехзначное число" и т.д.

4. Оператор выбора

- 4.1. Дан номер месяца (1 — январь, 2 — февраль, ...). Вывести название соответствующего времени года ("зима", "весна" и т.д.).
- 4.2. Дан номер месяца (1 — январь, 2 — февраль, ...). Вывести число дней в этом месяце для невисокосного года.
- 4.3. Дано целое число в диапазоне 0..9. Вывести строку — название соответствующей цифры на русском языке (0 — "ноль", 1 — "один", 2 — "два", ...).
- 4.4. Дано целое число в диапазоне 1..5. Вывести строку — словесное описание соответствующей оценки (1 — "плохо", 2 — "неудовлетворительно", 3 — "удовлетворительно", 4 — "хорошо", 5 — "отлично").
- 4.5. Арифметические действия над числами пронумерованы следующим образом: 1 — сложение, 2 — вычитание, 3 — умножение, 4 — деление. Дан номер действия и два числа А и В (В не равно нулю). Выполнить над числами указанное действие и вывести результат.
- 4.6. 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.

- 4.7. Единицы массы пронумерованы следующим образом: 1 — килограмм, 2 — миллиграмм, 3 — грамм, 4 — тонна, 5 — центнер. Дан номер единицы массы и масса тела M в этих единицах (вещественное число). Вывести массу данного тела в килограммах.
- 4.8. Робот может перемещаться в четырех направлениях ("С" — север, "З" — запад, "Ю" — юг, "В" — восток) и принимать три цифровые команды: 0 — продолжать движение, 1 — поворот налево, -1 — поворот направо. Дан символ C — исходное направление робота и число N — посланная ему команда. Вывести направление робота после выполнения полученной команды.
- 4.9. 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.
- 4.10. 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 P_i использовать 3.14.

4.11. Элементы равнобедренного прямоугольного треугольника пронумерованы следующим образом: 1 — катет (a), 2 — гипотенуза (c), 3 — высота, опущенная на гипотенузу (h), 4 — площадь (S). Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данного треугольника (в том же порядке).

4.12. Элементы равностороннего треугольника пронумерованы следующим образом: 1 — сторона (a), 2 — радиус вписанной окружности (R_1), 3 — радиус описанной окружности (R_2), 4 — площадь (S). Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данного треугольника (в том же порядке).

4.13. Даны два целых числа: D (день) и M (месяц), определяющие правильную дату невисокосного года. Вывести значения D и M для даты, предшествующей указанной.

4.14. Даны два целых числа: D (день) и M (месяц), определяющие правильную дату невисокосного года.

Вывести значения D и M для даты, следующей за указанной.

- 4.15. 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".
- 4.16. Дано целое число в диапазоне 100..999. Вывести строку — словесное описание данного числа, например: 256 — "двести пятьдесят шесть", 814 — "восемьсот четырнадцать".
- 4.17. 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.

5. Операторы цикла

- 5.1. Даны два целых числа A и B ($A < B$). Вывести все целые числа, расположенные между данными числами (включая сами эти числа), в порядке их возрастания, а также количество N этих чисел.
- 5.2. Даны два целых числа A и B ($A < B$). Вывести все целые числа, расположенные между данными числами (не включая сами эти числа), в порядке их убывания, а также количество N этих чисел.
- 5.3. Дано вещественное число A и целое число $N > 0$. Вывести A в степени N : $A^N = A \cdot A \cdot \dots \cdot A$ (числа A перемножаются N раз).
- 5.4. Дано вещественное число A и целое число $N > 0$. Вывести все целые степени числа A от 1 до N .
- 5.5. Дано вещественное число A и целое число $N > 0$. Вывести $1 + A + A^2 + A^3 + \dots + A^N$.
- 5.6. Дано вещественное число A и целое число $N > 0$. Вывести $1 - A + A^2 - A^3 + \dots + (-1)^N \cdot A^N$.
- 5.7. Дано целое число $N > 1$. Вывести наименьшее целое K , при котором выполняется неравенство $3K > N$, и само значение $3K$.

- 5.8. Дано целое число $N > 1$. Вывести наибольшее целое K , при котором выполняется неравенство $3K < N$, и само значение $3K$.
- 5.9. Дано вещественное число $A > 1$. Вывести наименьшее из целых чисел N , для которых сумма $1 + 1/2 + \dots + 1/N$ будет больше A , и саму эту сумму.
- 5.10. Дано вещественное число $A > 1$. Вывести наибольшее из целых чисел N , для которых сумма $1 + 1/2 + \dots + 1/N$ будет меньше A , и саму эту сумму.
- 5.11. Дано целое число $N > 0$. Вывести произведение $1 \cdot 2 \cdot \dots \cdot N$. Чтобы избежать целочисленного переполнения, вычислять это произведение с помощью вещественной переменной и выводить его как вещественное число.
- 5.12. Дано целое число $N > 0$. Если N — нечетное, то вывести произведение $1 \cdot 3 \cdot \dots \cdot N$; если N — четное, то вывести произведение $2 \cdot 4 \cdot \dots \cdot N$. Чтобы избежать целочисленного переполнения, вычислять это произведение с помощью вещественной переменной и выводить его как вещественное число.
- 5.13. Дано целое число $N > 0$. Вывести сумму $2 + 1/(2!) + 1/(3!) + \dots + 1/(N!)$ (выражение $N!$ — "N факториал" — обозначает произведение всех целых чисел от 1 до N : $N! = 1 \cdot 2 \cdot \dots \cdot N$). Полученное число является

приближенным значением константы $e = \exp(1) = 2.71828183\dots$

- 5.14. Дано вещественное число X и целое число $N > 0$. Вывести $1 + X + X^2/2! + \dots + X^N/N!$ ($N! = 1 \cdot 2 \cdot \dots \cdot N$). Полученное число является приближенным значением функции \exp в точке X .
- 5.15. Дано вещественное число X и целое число $N > 0$. Вывести $X - X^3/3! + X^5/5! - \dots + (-1)^N X^{2N+1}/(2N+1)!$ ($N! = 1 \cdot 2 \cdot \dots \cdot N$). Полученное число является приближенным значением функции \sin в точке X .
- 5.16. Дано вещественное число X и целое число $N > 0$. Вывести $1 - X^2/2! + X^4/4! - \dots + (-1)^N X^{2N}/(2N)!$ ($N! = 1 \cdot 2 \cdot \dots \cdot N$). Полученное число является приближенным значением функции \cos в точке X .
- 5.17. Дано вещественное число X ($|X| < 1$) и целое число $N > 0$. Вывести $X - X^2/2 + X^3/3 - \dots + (-1)^{N-1} X^N/N$. Полученное число является приближенным значением функции \ln в точке $1+X$.
- 5.18. Дано вещественное число X ($|X| < 1$) и целое число $N > 0$. Вывести $X - X^3/3 + X^5/5 - \dots + (-1)^N X^{2N+1}/(2N+1)$. Полученное число является приближенным значением функции \arctg в точке X .
- 5.19. Дано целое число $N > 2$ и две вещественные точки на числовой оси: A, B ($A < B$). Отрезок $[A, B]$ разбит на

равные отрезки длины H с концами в N точках вида A , $A + H$, $A + 2H$, $A + 3H$, ..., B . Вывести значение H и набор из N точек, образующий разбиение отрезка $[A, B]$.

5.20. Дано целое число $N > 2$ и две вещественные точки на числовой оси: A , B ($A < B$). Функция $F(X)$ задана формулой $F(X) = 1 - \sin(X)$. Вывести значения функции F в N равноотстоящих точках, образующих разбиение отрезка $[A, B]$: $F(A)$, $F(A + H)$, $F(A + 2H)$, ..., $F(B)$.

5.21. Дано число $D > 0$. Последовательность чисел A_N определяется следующим образом: $A_1 = 1$, $A_2 = 2$, $A_N = (A_{N-2} + A_{N-1})/2$, $N = 3, 4, \dots$. Найти первый из номеров K , для которых выполняется условие $|A_K - A_{K-1}| < D$, и вывести этот номер, а также числа A_{K-1} и A_K .

6. Обработка последовательностей

Во всех заданиях данного пункта предполагается, что исходный набор содержит ненулевое число элементов (в частности, число N всегда больше нуля).

- 6.1. Даны десять чисел. Вывести их среднее арифметическое.
- 6.2. Дано целое число N и набор из N вещественных чисел. Вывести сумму и произведение чисел из данного набора.
- 6.3. Дано целое число N и набор из N ненулевых целых чисел. Вывести в том же порядке все четные числа из данного набора и K таких чисел.
- 6.4. Дано целое число N и набор из N ненулевых целых чисел. Вывести в том же порядке номера всех нечетных чисел из данного набора и количество K таких чисел.
- 6.5. Даны целые числа K , N и набор из N целых чисел. Если в наборе присутствует число, меньшее K , то вывести True; в противном случае вывести False.
- 6.6. Дан набор ненулевых целых чисел; признак его завершения — число 0. Вывести количество элементов в наборе.

- 6.7. Дан набор ненулевых целых чисел; признак его завершения — число 0. Вывести сумму всех положительных четных чисел из данного набора. Если требуемые числа в наборе отсутствуют, то вывести 0.
- 6.8. Дано целое число K и набор ненулевых целых чисел; признак его завершения — число 0. Вывести количество чисел в наборе, меньших K .
- 6.9. Дано целое число K и набор ненулевых целых чисел; признак его завершения — число 0. Вывести номер первого числа в наборе, большего K . Если таких чисел в наборе нет, то вывести 0.
- 6.10. Дано целое число K и набор ненулевых целых чисел; признак его завершения — число 0. Вывести номер последнего числа в наборе, меньшего K . Если таких чисел в наборе нет, то вывести 0.
- 6.11. Дано целое число N и набор из N целых чисел. Вывести номера тех чисел в наборе, которые меньше своего левого соседа, и количество K таких чисел.
- 6.12. Дано целое число N и набор из N целых чисел. Вывести номера тех чисел в наборе, которые больше своего правого соседа, и количество K таких чисел.
- 6.13. Дано целое число N и набор из N вещественных чисел. Проверить, образует ли данный набор

возрастающую последовательность. Если образует, то вывести True, если нет — вывести False.

- 6.14. Дано целое число N и набор из N вещественных чисел. Если данный набор образует убывающую последовательность, то вывести 0; в противном случае вывести номер первого числа, нарушающего закономерность.
- 6.15. Дано целое число N и набор из N целых чисел, содержащий по крайней мере два нуля. Вывести сумму чисел из данного набора, расположенных между первыми двумя нулями (если первые нули идут подряд, то вывести 0).
- 6.16. Дано целое число N и набор из N целых чисел, содержащий по крайней мере два нуля. Вывести сумму чисел из данного набора, расположенных между последними двумя нулями (если последние нули идут подряд, то вывести 0).
- 6.17. Даны целые числа K , N и набор из N вещественных чисел: A_1, A_2, \dots, A_N . Вывести K -е степени чисел из данного набора: $A_1^K, A_2^K, \dots, A_N^K$.
- 6.18. Дано целое число N и набор из N вещественных чисел: A_1, A_2, \dots, A_N . Вывести следующие числа: $A_1, A_2^2, \dots, A_{10 \cdot (N-1) + N - 1}, A_N^N$.

6.19. Дано целое число N и набор из N вещественных чисел: A_1, A_2, \dots, A_N . Вывести следующие числа: $A_{1N}, A_{2N-1}, \dots, A_{N-12}, A_N$.

7. Минимумы и максимумы

Для решения заданий из данной группы следует использовать "однопроходные" алгоритмы, позволяющие получить требуемый результат после однократного просмотра набора исходных данных. Все указания предполагают именно такие алгоритмы решения. Однопроходные алгоритмы обладают важным преимуществом: для решения задачи не требуется хранить в памяти одновременно весь набор данных. Поэтому при программировании алгоритмов не нужно использовать массивы.

- 7.1. Найти минимальный и максимальный из данных десяти элементов.
- 7.2. Найти номера минимального и максимального из данных десяти элементов.
- 7.3. Найти номера первого и последнего минимального и максимального из данных десяти целочисленных элементов.
- 7.4. Найти количество минимальных и максимальных из данных десяти целочисленных элементов.
- 7.5. Найти минимальный и максимальный четный и нечетный из данных десяти ненулевых целочисленных элементов. Если требуемые элементы отсутствуют, то вывести 0.
- 7.6. Найти [минимальный положительный] и [максимальный отрицательный] из данных десяти

элементов. Если требуемые элементы отсутствуют, то вывести 0.

- 7.7. Даны числа a , b ($0 < a < b$) и набор из десяти элементов. Найти минимальный и максимальный из элементов, содержащихся в интервале (a, b) . Если требуемые элементы отсутствуют, то вывести -1 .
- 7.8. Дан набор из десяти целочисленных элементов. Найти количество элементов, расположенных перед минимальным и после максимального.
- 7.9. Найти номер первого экстремального (то есть минимального или максимального) из данных десяти целочисленных элементов.
- 7.10. Дан набор из десяти целочисленных элементов. Найти количество элементов, содержащихся между первым и последним минимальным. Если в наборе имеется единственный минимальный элемент, то вывести 0.
- 7.11. Найти два наибольших из данных десяти элементов.
- 7.12. Дан набор из десяти целочисленных элементов. Найти максимальное количество подряд идущих минимальных и максимальных элементов.

8. Одномерные массивы

- 8.1. Дан массив размера N . Вывести его элементы в обратном порядке.

- 8.2. Дан массив размера N . Вывести вначале его элементы с четными индексами, а затем — с нечетными.
- 8.3. Дан целочисленный массив A размера 10. Вывести номер первого из тех его элементов $A[i]$, которые удовлетворяют двойному неравенству: $A[1] < A[i] < A[10]$. Если таких элементов нет, то вывести 0.
- 8.4. Дан целочисленный массив размера N . Преобразовать его, прибавив к четным числам последний элемент. Последний элемент массива не изменять.
- 8.5. Дан целочисленный массив размера N . Вывести вначале все его нечетные элементы, а затем — четные.
- 8.6. Поменять местами минимальный и максимальный элементы массива размера 10.
- 8.7. Заменить все положительные элементы целочисленного массива размера 10 на значение минимального.
- 8.8. Дан массив размера 10. Переставить в обратном порядке элементы массива, расположенные между его минимальным и максимальным элементами.
- 8.9. Дан массив размера N . Осуществить циклический сдвиг элементов массива вправо на одну позицию.
- 8.10. Дан массив размера N и число k ($0 < k < 5$, $k < N$). Осуществить циклический сдвиг элементов массива влево на k позиций.

- 8.11. Проверить, образуют ли элементы целочисленного массива размера N арифметическую прогрессию. Если да, то вывести шаг прогрессии, если нет — вывести 0.
- 8.12. Дан массив размера N . Найти количество его минимумов.
- 8.13. Дан массив размера N . Определить количество участков, на которых его элементы монотонно возрастают.
- 8.14. Дан массив размера N . Определить количество его промежутков монотонности (то есть участков, на которых его элементы возрастают или убывают).
- 8.15. Дано вещественное число R и массив размера N . Найти элемент массива, который наиболее близок к данному числу.
- 8.16. Дано вещественное число R и массив размера N . Найти два элемента массива, сумма которых наиболее близка к данному числу.
- 8.17. Дан массив размера N . Найти номера двух ближайших чисел из этого массива.
- 8.18. Дан целочисленный массив размера N . Определить максимальное количество его одинаковых элементов.
- 8.19. Дан целочисленный массив размера N . Если он является перестановкой, то есть содержит все числа

от 1 до N , то вывести 0, в противном случае вывести номер первого недопустимого элемента.

- 8.20. Дан целочисленный массив размера N . Назовем серией группу подряд идущих одинаковых элементов, а длиной серии — количество этих элементов (длина серии может быть равна 1). Вывести массив, содержащий длины всех серий исходного массива.
- 8.21. Даны два массива A и B размера 5 , элементы которых упорядочены по возрастанию. Объединить эти массивы так, чтобы результирующий массив остался упорядоченным.
- 8.22. Упорядочить массив размера N по убыванию.
- 8.23. Дан массив размера N . Вывести индексы массива в том порядке, в котором соответствующие им элементы образуют возрастающую последовательность.
- 8.24. Дана точка A и множество B из N точек. Найти номер точки из множества B , наиболее близкой к точке A .
- 8.25. Дано множество A из N точек. Среди всех точек этого множества, лежащих в первой четверти, найти точку, наиболее близкую от начала координат. Если таких точек нет, то вывести точку с нулевыми координатами.

- 8.26. Дано множество A из N точек. Найти такую точку из данного множества, сумма расстояний от которой до остальных его точек минимальна, и саму эту сумму.
- 8.27. Даны множества A и B , состоящие соответственно из N_1 и N_2 точек. Найти максимальное расстояние между точками этих множеств и сами точки, расположенные на этом расстоянии.

9. Двумерные массивы (матрицы)

- 9.1. Дано число k ($0 < k < 11$) и матрица размера 4×10 . Найти сумму и произведение элементов k -го столбца данной матрицы.
- 9.2. Дана матрица размера 5×9 . Найти суммы элементов всех ее четных столбцов.
- 9.3. Дана матрица размера 5×10 . Найти минимальное значение в каждой строке.
- 9.4. Дана матрица размера 5×10 . В каждой строке найти количество элементов, меньших среднего арифметического всех элементов этой строки.
- 9.5. Дана матрица размера 5×10 . Преобразовать матрицу, поменяв местами минимальный и максимальный элемент в каждой строке.
- 9.6. Дана матрица размера 5×10 . Найти минимальное значение среди сумм элементов всех ее столбцов и номер столбца с этим минимальным значением.
- 9.7. Дана матрица размера 5×10 . Вывести номер ее первой строки, содержащей только положительные элементы. Если таких строк нет, то вывести 0.
- 9.8. Дана целочисленная матрица размера $M \times N$. Различные строки (столбцы) матрицы назовем похожими, если совпадают множества чисел, встречающихся в этих

строках (столбцах). Найти количество строк, похожих на первую строку.

- 9.9. Дана целочисленная матрица размера $M \times N$. Найти количество ее столбцов, все элементы которых различны.
- 9.10. Дана квадратная матрица порядка M . Найти сумму элементов ее главной диагонали.
- 9.11. Дана квадратная матрица порядка M . Найти сумму элементов ее побочной диагонали.
- 9.12. Дана квадратная матрица порядка M . Найти суммы элементов ее диагоналей, параллельных главной (начиная с одноэлементной диагонали $A[1,M]1|A[1,1]2$).
- 9.13. Дана квадратная матрица порядка M . Вывести минимальные из элементов каждой ее диагонали, параллельной побочной (начиная с одноэлементной диагонали $A[1,M]3|A[1,1]4$).
- 9.14. Дана квадратная матрица порядка M . Заменить нулями элементы матрицы, лежащие выше побочной диагонали.
- 9.15. Дана квадратная матрица порядка M . Заменить нулями элементы, лежащие одновременно выше главной диагонали (включая эту диагональ) и ниже побочной диагонали (также включая эту диагональ).

- 9.16. Дана квадратная матрица порядка M . Зеркально отразить ее элементы относительно горизонтальной оси симметрии матрицы.
- 9.17. Дана квадратная матрица порядка M . Повернуть ее на 90 градусов в положительном направлении.
- 9.18. Дана матрица размера 5×10 . Вывести количество строк¹, элементы которых монотонно возрастают.
- 9.19. Дана матрица размера 5×10 . Найти максимальный среди элементов тех столбцов, которые упорядочены либо по возрастанию, либо по убыванию. Если такие столбцы отсутствуют, то вывести 0 .
- 9.20. Даны два числа k_1 и k_2 и матрица размера 4×10 . Поменять местами строки матрицы с номерами k_1 и k_2 .
- 9.21. Дана матрица размера 5×10 . Поменять местами столбцы, содержащие минимальный и максимальный элементы матрицы.
- 9.22. Дана матрица размера 5×10 . Поменять местами столбец с номером 1 и последний из столбцов, содержащих только положительные элементы.
- 9.23. Дана целочисленная матрица размера $M \times N$. Найти элемент, являющийся максимальным в своей строке и минимальным в своем столбце. Если такой элемент отсутствует, то вывести 0 .

9.24. Дана матрица размера $M \times N$. Поменять местами ее столбцы так, чтобы их максимальные элементы образовывали убывающую последовательность.

10. Символы и строки

- 10.1. Вывести строку длины N (N — четное), которая состоит из чередующихся символов $C1$ и $C2$, начиная с $C1$.
- 10.2. Дана строка. Вывести строку, содержащую те же символы, но расположенные в обратном порядке.
- 10.3. Дана строка. Вывести коды ее первого и последнего символа.
- 10.4. Дана строка. Подсчитать количество содержащихся в ней цифр.
- 10.5. Дана строка. Преобразовать все прописные латинские буквы в строчные.
- 10.6. Дана строка. Если она представляет собой запись целого числа, то вывести 1; если вещественного (с дробной частью), то вывести 2; если строку нельзя преобразовать в число, то вывести 0.
- 10.7. Дано целое число. Вывести набор символов, содержащий цифры этого числа в обратном порядке.
- 10.8. Дана строка S , изображающая вещественное число в формате с плавающей точкой, и целое число $N > 0$. Вывести набор символов, изображающих первые N цифр дробной части этого вещественного числа (без округления).

- 10.9. Дана строка, изображающая двоичную запись целого числа. Вывести строку, изображающую десятичную запись этого же числа.
- 10.10. Дана строка, изображающая целое число. Вывести сумму цифр этого числа.
- 10.11. Дана строка S и число N . Преобразовать строку S в строку длины N следующим образом: если длина строки S больше N , то отбросить первые символы, если длина строки S меньше N , то в ее начало добавить символы "." (точка).
- 10.12. Даны два числа: $N1$ и $N2$, и две строки: $S1$ и $S2$. Получить из этих строк новую строку, объединив $N1$ первых символов строки $S1$ и $N2$ последних символов строки $S2$.
- 10.13. Даны две строки: $S1$ и $S2$. Проверить, содержится ли строка $S2$ в строке $S1$. Если да, то вывести номер позиции, начиная с которой $S2$ содержится в $S1$, если нет, то вывести 0.
- 10.14. Даны две строки: $S1$ и $S2$. Определить количество вхождений строки $S2$ в строку $S1$.
- 10.15. Дана строка S и символ C . Удвоить каждое вхождение символа C в строку S .
- 10.16. Даны строки $S1$, $S2$ и символ C . после каждого вхождения символа C в строку $S1$ вставить строку $S2$.

- 10.17. Даны две строки: S1 и S2. Удалить из строки S1 все подстроки, совпадающие с S2. Если таких подстрок нет, то вывести S1 без изменений.
- 10.18. Даны три строки: S1, S2, S3. Заменить в строке S1 все вхождения строки S2 на S3.
- 10.19. Дана строка. Вывести подстроку, расположенную между первой и второй точками исходной строки. Если в строке менее двух точек, то вывести всю исходную строку.
- 10.20. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Определить количество слов в строке.
- 10.21. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Определить количество слов, которые начинаются и заканчиваются одной и той же буквой.
- 10.22. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Определить количество слов, которые содержат ровно три буквы "А".
- 10.23. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Определить длину самого длинного слова.

- 10.24. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова, но разделенные одним символом "." (точка). В конце точку не ставить.
- 10.25. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова (разделенные одним пробелом), но расположенные в обратном порядке.
- 10.26. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова (разделенные одним пробелом), но расположенные в алфавитном порядке.
- 10.27. Дана строка-предложение на русском языке. Преобразовать строку так, чтобы каждое слово начиналось с заглавной буквы.
- 10.28. Дана строка-предложение на русском языке. Подсчитать количество содержащихся в строке знаков препинания.
- 10.29. Дана строка-предложение на русском языке. Вывести самое короткое слово в предложении (если таких слов несколько, то вывести первое из них).
- 10.30. Дана строка-предложение, содержащая избыточные пробелы. Преобразовать ее так, чтобы между словами был ровно один пробел.

- 10.31. Дана строка, содержащая полное имя файла, то есть имя диска, список каталогов (путь), собственно имя и расширение. Выделить из этой строки расширение файла.
- 10.32. Дана строка, содержащая полное имя файла. Выделить из строки название последнего каталога (без символов "\"). Если файл содержится в корневом каталоге, то вывести символ "\".
- 10.33. Дана строка-предложение на русском языке. Зашифровать ее, выполнив циклическую замену каждой буквы на следующую за ней в алфавите и сохраняя при этом регистр букв ("А" перейдет в "Б", "а" — в "б", "Б" — в "В", "я" — в "а" и т.д.). Букву "ё" в алфавите не учитывать ("е" должна переходить в "ж"). Знаки препинания и пробелы не изменять.
- 10.34. Дана строка-предложение на русском языке и число k ($0 < k < 10$). Зашифровать строку, выполнив циклическую замену каждой буквы на букву того же регистра, расположенную в алфавите на k -й позиции после шифруемой буквы (например, для $k = 2$ "А" перейдет в "В", "а" — в "в", "Б" — в "Г", "я" — в "б" и т.д.). Букву "ё" в алфавите не учитывать, знаки препинания и пробелы не изменять.
- 10.35. Дано зашифрованное предложение на русском языке (способ шифрования описан в предыдущем задании) и

кодовое смещение k ($0 < k < 10$). Расшифровать предложение.

- 10.36. Дана строка-предложение. Зашифровать ее, поместив вначале все символы, расположенные на четных местах, а затем, в обратном порядке, все символы, расположенные на нечетных местах (например, строка "Программа" превратится в "ргамамрП").
- 10.37. Дано предложение, зашифрованное по правилу, описанному в предыдущем задании. Расшифровать это предложение.
- 10.38. Дана строка, содержащая несколько круглых скобок. Если скобки расставлены правильно (то есть каждой открывающей соответствует одна закрывающая), то вывести число 0. В противном случае вывести или номер позиции, в которой расположена первая ошибочная закрывающая скобка, или, если закрывающих скобок не хватает, число -1 .

11. Процедуры и функции (вычисление)

В заданиях данной подгруппы требуется реализовать процедуры или функции с числовыми параметрами типа `integer` и `real`. Входные параметры этих типов обычно описываются как параметры-значения.

- 11.1. Описать функцию $\text{Min2}(A,B)$ вещественного типа, 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 .
- 11.2. Описать процедуру $\text{MinMax}(A,B)$, записывающую в переменную A минимальное из значений A и B , а в переменную B — максимальное из этих значений (A и B — вещественные параметры, являющиеся одновременно входными и выходными). Используя четыре вызова этой процедуры, найти минимальное и максимальное из чисел A , B , C , D .
- 11.3. Описать функцию $\text{Fact}(N)$ целого типа, вычисляющую значение факториала $N! = 1 \cdot 2 \cdot \dots \cdot N$ ($N > 0$ — параметр целого типа). С помощью этой функции вычислить факториалы 10 данных чисел.
- 11.4. Описать функцию $\text{FactR}(N)$ вещественного типа, позволяющую вычислять приближенное значение факториала $N! = 1 \cdot 2 \cdot \dots \cdot N$ для целых $N > 0$. С помощью

этой функции вычислить факториалы пяти данных чисел.

- 11.5. Описать функцию $\text{Fact2}(N)$ целого типа, вычисляющую значение "двойного факториала": $N!! = 1 \cdot 3 \cdot 5 \cdot \dots \cdot N$, если N — нечетное, $N!! = 2 \cdot 4 \cdot 6 \cdot \dots \cdot N$, если N — четное ($N > 0$ — параметр целого типа). С помощью этой функции вычислить двойные факториалы десяти данных чисел.
- 11.6. Описать нерекурсивную функцию $\text{Fib}(N)$ целого типа, вычисляющую N -е число Фибоначчи $F(N)$ по формуле: $F(1) = F(2) = 1$, $F(k) = F(k-2) + F(k-1)$, $k = 3, 4, \dots$. С помощью этой функции вычислить 10 чисел Фибоначчи с указанными номерами.
- 11.7. Описать процедуру $\text{SumDigit}(N, S)$, находящую сумму цифр S целого числа N (N — входной, S — выходной параметр). Используя эту процедуру, найти суммы цифр пяти данных чисел.
- 11.8. Описать нерекурсивную функцию $\text{NOD2}(A, B)$ целого типа, находящую наибольший общий делитель (НОД) двух натуральных чисел A и B , используя алгоритм Евклида: $\text{НОД}(A, B) = \text{НОД}(B \bmod A, A)$, если $A \neq 0$; $\text{НОД}(0, B) = B$. 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 .

- 11.9. Описать функцию $\text{Exp1}(x, \text{eps})$ вещественного типа (параметры x , eps — вещественные, $\text{eps} > 0$), находящую приближенное значение функции $\exp(x)$:

$$\exp(x) = 1 + x + x^2 / 2! + x^3 / 3! + \dots + x^n / n! + \dots$$
 В сумме учитывать все слагаемые, большие eps . С помощью Exp1 найти приближенное значение экспоненты для данного x при шести данных eps .
- 11.10. Описать функцию $\text{Otr}(Ax, Ay, Bx, By)$ вещественного типа, находящую длину отрезка AB на плоскости по координатам его концов: $|AB| = \sqrt{(Ax - Bx)^2 + (Ay - By)^2}$ (Ax, Ay, Bx, By — вещественные параметры). С помощью этой функции найти длины отрезков AB, AC, AD , если даны координаты точек A, B, C, D .
- 11.11. Используя функцию Otr из предыдущего задания, описать функцию $\text{Perim}(Ax, Ay, Bx, By, Cx, Cy)$ вещественного типа, находящую периметр треугольника ABC по координатам его вершин (Ax, Ay, Bx, By, Cx, Cy — вещественные параметры). С помощью этой функции найти периметры треугольников ABC, ABD, ACD , если даны координаты точек A, B, C, D .
- 11.12. Используя функции Otr и Perim из предыдущих заданий, описать функцию $\text{Area}(Ax, Ay, Bx, By, Cx, Cy)$ вещественного типа, находящую площадь треугольника ABC по формуле Герона: $S_{ABC} =$

$\sqrt{p \cdot (p - |AB|) \cdot (p - |AC|) \cdot (p - |BC|)}$), где p — полупериметр. С помощью этой функции найти площади треугольников ABC , ABD , ACD , если даны координаты точек A , B , C , D .

- 11.13. Используя функции Otr и $Area$ из предыдущих заданий, описать процедуру $Dist(Px, Py, Ax, Ay, Bx, By, D)$, находящую расстояние D от точки P до прямой AB по формуле $D = 2SP_{AB} / |AB|$, где SP_{AB} — площадь треугольника PAB . С помощью этой процедуры найти расстояния от точки P до прямых AB , AC , BC , если даны координаты точек P , A , B , C .
- 11.14. Используя процедуру $Dist$ из предыдущего задания, описать процедуру $Heights(Ax, Ay, Bx, By, Cx, Cy, hA, hB, hC)$, находящую высоты hA , hB , hC треугольника ABC , проведенные соответственно из вершин A , B , C . С помощью этой процедуры найти высоты треугольников ABC , ABD , ACD , если даны координаты точек A , B , C , D .

12. Процедуры и функции (массивы)

При описании процедур и функций в заданиях данной подгруппы необходимо учитывать особенности, связанные с передачей массивов в качестве параметров. Для одномерных параметров-массивов рекомендуется использовать механизм открытых массивов, введенный в Borland Pascal 7.0. Для двумерных массивов-матриц подобный механизм использовать нельзя, поэтому предварительно требуется определить соответствующий пользовательский тип, который в дальнейшем использовать при описании параметров-матриц. Входные параметры-массивы обычно не описывают как параметры-значения, поскольку это приводит, как правило, к неоправданному расходу памяти. Если массив в ходе выполнения процедуры/функции не изменяется, его нужно описать как параметр-константу, а если изменяется, то как параметр-переменную. Вспомогательные локальные переменные-массивы в процедурах/функциях при выполнении заданий использовать не следует.

- 12.1. Описать функцию $\text{Polynom}(A, N, X)$ вещественного типа, находящую значение полинома P в вещественной точке X . Полином P задается параметрами N (степень полинома, $0 < N < 8$) и A (коэффициенты полинома, вещественный массив размера $N+1$): $P(X) = A[1] \cdot X^N + A[2] \cdot X^{N-1} + \dots + A[N] \cdot X + A[N+1]$. Используя эту функцию, найти значения заданного полинома в пяти данных точках.
- 12.2. Описать функцию $\text{Min}(A, N) | \text{Max}(A, N)$ вещественного типа, находящую минимальный элемент массива A , состоящего из N вещественных чисел. С помощью этой

функции найти минимальные элементы массивов A , B , C размера NA , NB , NC соответственно.

- 12.3. Описать функцию $NMax(A,N)$ целого типа, находящую номер максимального элемента массива A (массив состоит из N вещественных чисел). С помощью этой функции найти номера максимальных элементов массивов A , B , C размера NA , NB , NC соответственно.
- 12.4. Описать процедуру $NMinmax(A,N,NMin,NMax)$, находящую номера минимального и максимального элемента массива A из N вещественных чисел. Выходные параметры: $NMin$ (номер минимального элемента) и $NMax$ (номер максимального элемента). С помощью этой процедуры найти номера минимальных и максимальных элементов массивов A , B , C размера NA , NB , NC соответственно.
- 12.5. Описать процедуру $Invert(A,N)$, меняющую порядок следования элементов массива A из N вещественных чисел на противоположный ("инвертирование массива"). Массив A — входной и выходной параметр, N — входной параметр. С помощью этой процедуры инвертировать массивы A , B , C размера NA , NB , NC соответственно.
- 12.6. Описать процедуру $MoveRight(A,N,k)$, осуществляющую циклический сдвиг элементов вещественного массива A размера N на k позиций

вправо ($0 < k < 5$, $k < N$). Массив A — входной и выходной параметр, N и k — входные параметры. С помощью этой процедуры осуществить сдвиг элементов данного массива размера N на k_1 позиций, а затем — сдвиг элементов полученного массива на k_2 позиций (k_1 и k_2 даны). После каждого вызова процедуры выводить на экран результирующий массив.

- 12.7. Описать процедуру $\text{Smooth}(A,N)$, заменяющую каждый элемент вещественного массива A размера N на его среднее арифметическое со своими соседями ("сглаживание массива"). Массив A — входной и выходной параметр, N — входной параметр. С помощью этой процедуры выполнить пятикратное сглаживание данного массива A размера N , выводя на экран результаты каждого сглаживания.
- 12.8. Описать процедуру $\text{SortInc}(A,N)$, выполняющую сортировку по возрастанию массива A из N вещественных чисел. Массив A является входным и выходным параметром. С помощью этой процедуры отсортировать массивы A , B , C размера N_A , N_B , N_C соответственно.
- 12.9. Описать функцию $\text{Norm1}(A,M,N)$ вещественного типа, вычисляющую норму матрицы A размера $M \times N$ с вещественными элементами: $\text{Norm1}(A,M,N) = \max$

$\{|A[1,j]| + |A[2,j]| + \dots + |A[M,j]|\}$, где максимум берется по всем j от 1 до N . Для данной матрицы A размера $M \times N$ найти $\text{Norm1}(A,k,N)$, $k = 1, \dots, M$.

- 12.10. Описать функцию $\text{SumLine}(A,M,N,k)$ вещественного типа, вычисляющую сумму элементов вещественной матрицы A размера $M \times N$, расположенных в k -й строке (если $k > M$, то функция возвращает 0). Для данной матрицы A размера $M \times N$ и трех данных k найти $\text{SumLine}(A,M,N,k)$.
- 12.11. Описать процедуру $\text{SwapCol}(A,M,N,k1,k2)$, осуществляющую перемену местами столбцов вещественной матрицы A размера $M \times N$ с номерами $k1$ и $k2$ (если $k1$ или $k2$ больше N , то матрица не изменяется). Двумерный массив A — входной и выходной параметр, прочие параметры — входные. Используя эту процедуру, поменять для данной матрицы A размера $M \times N$ столбцы с номерами $k1$ и $k2$.
- 12.12. Описать процедуру $\text{Transp}(A,M)$, выполняющую транспонирование квадратной вещественной матрицы A порядка M . Двумерный массив A — входной и выходной параметр, M — входной параметр. Использовать эту процедуру для транспонирования данной матрицы A порядка M .

13. Процедуры и функции (строки)

Входные строковые параметры, если они не изменяются в теле процедуры/функции, для экономии памяти рекомендуется описывать как параметры-константы.

- 13.1. Описать функцию `IsIdent(S)` целого типа, проверяющую, является ли строка `S` допустимым идентификатором Паскаля. При утвердительном ответе возвращается `0`. Если `S` является пустой строкой, то возвращается `-1`, если строка начинается с цифры, то возвращается `-2`. Если `S` содержит недопустимые символы, то возвращается номер первого недопустимого символа. Проверить с помощью этой функции пять данных строк.
- 13.2. Описать функцию `FillStr(S,Len)` строкового типа, возвращающую строку длины `Len`, заполненную повторяющимися копиями строки-шаблона `S` (последняя копия строки-шаблона может входить в результирующую строку частично). Используя эту функцию, сформировать по данному числу `Len` и пяти данным строкам-шаблонам пять результирующих строк длины `Len`.
- 13.3. Описать процедуру `UpCase1(S)`, преобразующую все строчные буквы строки `S` в прописные (остальные

символы строки S не изменяются). Используя эту процедуру, преобразовать пять данных строк.

- 13.4. Описать процедуру $\text{TrimL}(S)$, удаляющую в строке S начальные пробелы. Используя эту процедуру, преобразовать пять данных строк.
- 13.5. Описать функцию $\text{PosLast}(\text{subS}, S)$ целого типа, возвращающую номер позиции, с которой в строке S содержится последнее вхождение подстроки subS . Если в строке S отсутствуют подстроки subS , то функция возвращает 0. Вывести значения этой функции для пяти данных пар subS и S .
- 13.6. Описать функцию $\text{PosK}(\text{subS}, S, k)$ целого типа, возвращающую номер позиции, с которой в строке S содержится k -е вхождение подстроки subS ($k > 0$). Если количество вхождений subS в строке S меньше k , то функция возвращает 0. Вывести значения этой функции для пяти данных троек: k , subS и S .
- 13.7. Описать функцию $\text{WordN}(S, k)$ строкового типа, возвращающую k -е слово строки S (под словом понимается набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки). Если количество слов в строке меньше k , то функция возвращает пустую строку. Используя эту функцию, выделить из данной строки S слова с номерами k_1, k_2, k_3 .

- 13.8. Описать процедуру `SplitStr(S,W,N)`, которая формирует по данной строке `S` набор слов `W`, входящих в `S` (`W` — выходной строковый массив; `N` — его размер; предполагается, что `N` не будет превышать 10). Под словом понимается набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки. Используя эту функцию, вывести количество слов `N`, содержащихся в данной строке `S`, и сами эти слова.
- 13.9. Описать функцию `DecToBin(N)` строкового типа, которая возвращает строковое представление целого неотрицательного числа `N` в двоичной системе счисления. Результирующая строка состоит из символов '0'-'1' и не содержит ведущих нулей (за исключением представления числа 0). Используя эту функцию, получить двоичные представления данных пяти чисел.
- 13.10. Описать функцию `BinToDec(S)` целого типа, которая определяет целое неотрицательное число по его строковому представлению `S` в двоичной системе счисления. Параметр `S` имеет тип `string`, состоит из символов '0'-'1' и не содержит ведущих нулей (за исключением значения '0'). Используя эту функцию, вывести пять чисел, для которых даны их двоичные представления.

14. Процедуры и функции (файлы)

Все входные строковые параметры, задающие имена файлов, для экономии памяти целесообразно описывать как параметры-константы.

- 14.1. Описать функцию `getInt(Name,k)` целого типа, возвращающую k -й элемент файла целых чисел с именем `Name` (элементы нумеруются от 0). Если файл не существует или не содержит k -го элемента, то функция возвращает 0. С помощью этой функции вывести пять элементов данного файла с указанными номерами.
- 14.2. Описать функцию `getLine(Name,k)` строкового типа, возвращающую k -ю строку текстового файла с именем `Name` (строки нумеруются от 0). Если файл не существует или не содержит k -й строки, то функция возвращает пустую строку. С помощью этой функции вывести пять строк данного файла с указанными номерами.
- 14.3. Описать функцию `IntFileSize(Name)` целого типа, возвращающую размер файла целых чисел с именем `Name`. Если файл не существует, то функция возвращает -1 . С помощью этой функции определить размер трех файлов с данными именами.
- 14.4. Описать функцию `LineCount(Name)` целого типа, возвращающую число строк в текстовом файле с

именем Name. Если файл не существует, то функция возвращает -1. С помощью этой функции определить размер трех файлов с данными именами.

- 14.5. Описать процедуру `InvertIntFile(Name)`, меняющую порядок следования элементов файла целого типа с именем Name на противоположный. Если файл не существует или содержит менее двух элементов, то процедура не выполняет никаких действий. Обработать с помощью этой процедуры три файла с данными именами.
- 14.6. Описать процедуру `SplitText(Name0,k,Name1,Name2)`, копирующую первые k (≥ 0) строк существующего текстового файла с именем Name0 в файл Name1, а остальные элементы — в файл Name2 (прежнее содержимое результирующих файлов стирается). Один из результирующих файлов может оказаться пустым. Применить эту процедуру к файлу Name0, используя указанные значения Name1, Name2 и k .
- 14.7. Описать процедуру `ConcatFile(NameA,NameB,NameAB)`, позволяющую объединить содержимое двух двоичных файлов NameA и NameB одного и того же типа в новом файле NameAB. Использовать процедуры `BlockRead` и `BlockWrite`. Применить эту процедуру к параметрам исходных файлов Name1-Name2, Name1-Name3 и

Name2–Name3, создав файлы с именами Name12, Name13, Name23.

- 14.8. Proc60. Описать процедуру `CodeText(Name,k)`, шифрующую текстовый файл с именем `Name`, выполняя циклическую замену каждой русской буквы на букву, расположенную в алфавите на k -й позиции после исходной ($0 < k < 11$). Например, при $k = 3$ "А" перейдет в "Г", "а" — в "г", "Я" — в "В". Букву "ё" в алфавите не учитывать, прочие символы файла не изменять. Используя эту процедуру и зная k , зашифровать и расшифровать данный файл.

15. Процедуры и функции (простейшие рекурсивные алгоритмы)

Следует заметить, что практически все задания этой подгруппы можно легко решить и без использования рекурсии. Данное обстоятельство связано с тем, что в заданиях рассматриваются действительно простейшие примеры рекурсии, легко сводимые к итерационным алгоритмам. Более того, в некоторых случаях использование рекурсии приводит к неэффективным алгоритмам (см., например, задания Proc64 и Proc65). Однако именно на подобных примерах проще всего получить первоначальные навыки разработки рекурсивных алгоритмов.

- 15.1. Описать рекурсивные функции $\text{Fact}(N)$ и $\text{Fact2}(N)$ вещественного типа, вычисляющие значения факториала $N!$ и двойного факториала $N!!$ соответственно ($N > 0$ — параметр целого типа). С помощью этих функций вычислить факториалы и двойные факториалы пяти данных чисел.
- 15.2. Описать рекурсивную функцию $\text{PowerN}(x,n)$ вещественного типа, находящую значение n -й степени числа x по формуле: $x_0 = 1$, $x_n = x \cdot x_{n-1}$ при $n > 0$, $x_n = 1 / x^{-n}$ при $n < 0$ ($x \geq 0$ — вещественное число, n — целое). С помощью этой функции найти значения XN при 5 различных значениях N для данного X .
- 15.3. Описать рекурсивную функцию $\text{SqrtK}(x,k,n)$ вещественного типа, находящую приближенное

значение корня k -й степени из числа x по формуле:
 $y(0) = 1, y(n+1) = y(n) - (y(n) - x / y(n)^{k-1}) / k$, где
 $y(n)$ обозначает $\text{SqrtK}(x,k,n)$ (x — вещественный
параметр, k и n — целые; $x > 0, k > 1, n > 0$). С
помощью этой функции найти приближенные
значения корня K -й степени из X при 6 различных
значениях N для данных X и K .

- 15.4. Описать рекурсивную функцию $C(m,n)$ целого типа, находящую число сочетаний из n элементов по m , используя формулу: $C(0,n) = C(n,n) = 1, C(m,n) = C(m,n-1) + C(m-1,n-1)$ при $0 < m < n$ (m и n — целые параметры; $n > 0, 0 \leq m \leq n$). Дано число N и пять различных значений M . Вывести числа $C(M,N)$ вместе с количеством рекурсивных вызовов функции C , потребовавшихся для их нахождения.
- 15.5. Описать рекурсивную функцию $\text{NOD}(A,B)$ целого типа, находящую наибольший общий делитель двух натуральных чисел A и B , используя алгоритм Евклида: $\text{NOD}(A,B) = \text{NOD}(B \bmod A, A)$, если $A \neq 0$; $\text{NOD}(0,B) = B$. 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 .
- 15.6. Описать рекурсивную функцию $\text{Digits}(S)$ целого типа, находящую количество цифр в строке S без использования оператора цикла. С помощью этой

функции найти количество цифр в данных пяти строках.

- 15.7. Описать рекурсивную функцию $\text{Simm}(S)$ логического типа, проверяющую, является ли симметричной строка S , без использования оператора цикла. С помощью этой функции проверить данные пять строк.

16. Процедуры и функции (разбор выражений)

- 16.1. Вывести значение целочисленного выражения, заданного в виде строки S. Выражение определяется следующим образом: $\langle \text{выражение} \rangle ::= \langle \text{цифра} \rangle \mid \langle \text{выражение} \rangle + \langle \text{цифра} \rangle \mid \langle \text{выражение} \rangle - \langle \text{цифра} \rangle$
- 16.2. Вывести значение целочисленного выражения, заданного в виде строки S. Выражение определяется следующим образом: $\langle \text{выражение} \rangle ::= \langle \text{терм} \rangle \mid \langle \text{выражение} \rangle + \langle \text{терм} \rangle \mid \langle \text{выражение} \rangle - \langle \text{терм} \rangle$
 $\langle \text{терм} \rangle ::= \langle \text{цифра} \rangle \mid \langle \text{терм} \rangle * \langle \text{цифра} \rangle$
- 16.3. Вывести значение целочисленного выражения, заданного в виде строки S. Выражение определяется следующим образом: $\langle \text{выражение} \rangle ::= \langle \text{терм} \rangle \mid \langle \text{выражение} \rangle + \langle \text{терм} \rangle \mid \langle \text{выражение} \rangle - \langle \text{терм} \rangle$
 $\langle \text{терм} \rangle ::= \langle \text{элемент} \rangle \mid \langle \text{терм} \rangle * \langle \text{элемент} \rangle$
 $\langle \text{элемент} \rangle ::= \langle \text{цифра} \rangle \mid (\langle \text{выражение} \rangle)$
- 16.4. Вывести значение целочисленного выражения, заданного в виде строки S. Выражение определяется следующим образом: $\langle \text{выражение} \rangle ::= \langle \text{цифра} \rangle \mid (\langle \text{выражение} \rangle \langle \text{знак} \rangle \langle \text{выражение} \rangle)$
 $\langle \text{знак} \rangle ::= + \mid - \mid *$

- 16.5. Вывести значение логического выражения, заданного в виде строки S. Выражение определяется следующим образом ("T" — True, "F" — False): <выражение> ::= T | F | And(<операнды>) | Or(<операнды>) <операнды> ::= <выражение>, <выражение>
- 16.6. Вывести значение логического выражения, заданного в виде строки S. Выражение определяется следующим образом ("T" — True, "F" — False): <выражение> ::= T | F | And(<операнды>) | Or(<операнды>) <операнды> ::= <выражение> | <выражение>, <операнды>
- 16.7. Вывести значение логического выражения, заданного в виде строки S. Выражение определяется следующим образом ("T" — True, "F" — False): <выражение> ::= T | F | And(<операнды>) | Or(<операнды>) | Not(<выражение>) <операнды> ::= <выражение> | <выражение>, <операнды>
- 16.8. Проверить правильность расстановки скобок в строке S. Текст в строке S определяется следующим образом: <текст> ::= <элемент> | <элемент><текст> <элемент> ::= a | b | c | (<текст>) | [<текст>] | {<текст>} Если текст составлен правильно, то вывести True, иначе вывести False.
- 16.9. Проверить правильность расстановки скобок в строке S (текст в строке S определяется по тем же правилам,

что и в предыдущем задании). Если текст составлен правильно, то вывести 0; в противном случае вывести номер первой ошибочной скобки или -1, если в строке недостаточно закрывающих скобок.

17. Процедуры и функции (деревья)

- 17.1. Дано упорядоченное дерево глубины $N > 0$, каждая внутренняя вершина которого имеет $K (< 9)$ непосредственных потомков, которые нумеруются от 1 до K . Корень дерева имеет номер 0. Записать в текстовый файл с именем Name все возможные пути, ведущие от корня к листьям (каждый путь записывается в отдельной строке файла). Перебирать пути, начиная с "самого левого" и заканчивая "самым правым", при этом первыми заменять конечные элементы пути.
- 17.2. Дано упорядоченное дерево глубины $N > 0$, каждая внутренняя вершина которого имеет $K (< 9)$ непосредственных потомков, которые нумеруются от 1 до K . Корень дерева имеет номер 0. Записать в текстовый файл с именем Name все пути, ведущие от корня к листьям и удовлетворяющие следующему условию: никакие соседние элементы пути не нумеруются одной и той же цифрой. Каждый путь записывается в отдельной строке файла. Порядок перебора путей — тот же, что в задании Proc81.
- 17.3. Дано упорядоченное дерево глубины N ($N > 0$ — четное), каждая внутренняя вершина которого имеет два непосредственных потомка: A с весом 1 и B с

весом -1 . Корень дерева S имеет вес 0 . Записать в текстовый файл с именем Name все пути от корня к листьям, удовлетворяющие следующему условию: суммарный вес элементов пути равен 0 . Каждый путь записывается в отдельной строке файла.

17.4. Дано упорядоченное дерево глубины N ($N > 0$) того же типа, что и в предыдущем задании. Записать в текстовый файл с именем Name все пути от корня к листьям, удовлетворяющие следующему условию: суммарный вес элементов для любого начального отрезка пути неотрицателен. Каждый путь записывается в отдельной строке файла.

18. Двоичные файлы

- 18.1. Дана строка S . Если S является допустимым именем файла, то вывести True и создать файл с этим именем. Если файл с именем S создать нельзя, то вывести False.
- 18.2. Даны имена четырех файлов. Вывести количество файлов с указанными именами, которые имеются в текущем каталоге.
- 18.3. Дано имя файла целых чисел. Вывести количество его элементов. Если файла с таким именем не существует, то вывести -1 .
- 18.4. Дано число k и файл, содержащий ненулевые целые числа. Вывести элемент файла с номером k (элементы файла нумеруются от нуля). Если такой элемент отсутствует, то вывести 0 .
- 18.5. Дан файл целых чисел, содержащий не менее четырех элементов. Вывести его нулевой, первый, предпоследний и последний элементы.
- 18.6. Даны имена двух файлов вещественных чисел. Известно, что один из них существует и содержит не менее двух элементов, а другой в текущем каталоге отсутствует. Создать отсутствующий файл и записать в

него нулевой и последний элементы существующего файла.

- 18.7. Дан файл целых чисел. Вывести количество содержащихся в нем серий (то есть наборов последовательно расположенных одинаковых элементов).
- 18.8. Дан файл вещественных чисел. Найти количество его локальных экстремумов.
- 18.9. Дан файл вещественных чисел. Найти количество его участков убывания.
- 18.10. Даны два файла произвольного типа. С помощью процедуры `Rename` поменять местами их содержимое.
- 18.11. Дан файл произвольного типа. С помощью процедур `BlockRead` и `BlockWrite` создать его копию с новым именем.
- 18.12. Дано целое число N (< 5) и N файлов одного и того же типа с именами `Name1`, ..., `Name`. С помощью процедур `BlockRead` и `BlockWrite` объединить содержимое этих файлов (в указанном порядке) в новом файле с именем `Name0`.
- 18.13. Даны два файла одного и того же типа. С помощью процедур `BlockRead` и `BlockWrite` добавить к первому файлу содержимое второго файла, а ко второму файлу — содержимое первого.

- 18.14. Даны три файла одного и того же типа, но разного размера. С помощью процедур BlockRead и BlockWrite заменить содержимое самого длинного файла на содержимое самого короткого.
- 18.15. Дан файл целых чисел. Создать новый файл, содержащий те же элементы, что и исходный файл, но в обратном порядке.
- 18.16. Даны три файла целых чисел одинакового размера с именами NameA, NameB и NameC. Создать новый файл с именем NameD, в котором чередовались бы элементы исходных файлов с одним и тем же номером: A0, B0, C0, A1, B1, C1, A2, B2, C2,
- 18.17. Даны четыре файла целых чисел разного размера с именами NameA, NameB, NameC и NameD. Создать новый файл с именем NameE, в котором чередовались бы элементы исходных файлов с одним и тем же номером. "Лишние" элементы более длинных файлов в результирующий файл не записывать.
- 18.18. Дан файл вещественных чисел с именем Name1. Создать два новых файла с именами Name2 и Name3, первый из которых содержит элементы исходного файла с четными номерами (0, 2, 4, ...), а второй — с нечетными (1, 3, 5, ...).

- 18.19. Дан файл, содержащий ненулевые целые числа. Создать новый файл, содержащий только положительные числа исходного файла (в том же порядке).
- 18.20. Дан файл целых чисел. Создать новый файл, содержащий длины всех серий исходного файла.
- 18.21. Дан файл вещественных чисел. Создать файл целых чисел, содержащий длины всех убывающих последовательностей его элементов.
- 18.22. Дан файл вещественных чисел. Заменить в нем все элементы на их квадраты.
- 18.23. Дан файл вещественных чисел. Заменить в файле каждый элемент, кроме начального и последнего, на его среднее арифметическое с предыдущим и последующим элементом.
- 18.24. Дан файл целых чисел с элементами $A(i)$, $i = 0, \dots, N-1$ (N — размер файла). Заменить исходное расположение его элементов на следующее: $A(0)$, $A(N-1)$, $A(1)$, $A(N-2)$, $A(2)$,
- 18.25. Дан файл целых чисел. Если его размер меньше 50, то дополнить его нулями до 50 элементов; если его размер больше 50, то урезать его до 50 элементов.
- 18.26. Дан файл целых чисел. Удалить в нем все отрицательные нечетные числа.

- 18.27. Дан файл целых чисел. Продублировать в нем все числа, принадлежащие диапазону 5..10.
- 18.28. Дан файл, содержащий ненулевые целые числа. Заменить в нем все положительные четные числа двумя нулями.
- 18.29. Дан файл вещественных чисел. Поменять в нем местами минимальный и максимальный элементы.
- 18.30. Даны два файла вещественных чисел с именами Name1 и Name2, элементы которых упорядочены по возрастанию. Объединить эти файлы в новый файл с именем Name3, сохранив упорядоченность элементов.
- 18.31. Даны два целых числа i и j и файл вещественных чисел, содержащий элементы квадратной матрицы (по строкам). Вывести элемент матрицы, расположенный в i -й строке и j -м столбце (строки и столбцы нумеруются от 1). Если требуемый элемент отсутствует, то вывести 0.
- 18.32. Даны два целых числа i и j и файл вещественных чисел, содержащий элементы прямоугольной матрицы (по строкам), причем начальный элемент файла содержит количество столбцов матрицы. Вывести элемент матрицы, расположенный в i -й строке и j -м столбце (строки и столбцы нумеруются от 1). Если требуемый элемент отсутствует, то вывести 0.

- 18.33. Дан файл вещественных чисел, содержащий элементы квадратной матрицы (по строкам). Создать файл, содержащий элементы матрицы, транспонированной к исходной.
- 18.34. Дан файл вещественных чисел, содержащий элементы прямоугольной матрицы (по строкам), причем начальный элемент файла содержит количество столбцов матрицы. Создать новый файл той же структуры, содержащий матрицу, транспонированную к исходной.
- 18.35. Даны два файла вещественных чисел с именами NameA и NameB, содержащие элементы квадратных матриц A и B (по строкам). Создать новый файл с именем NameC, содержащий элементы произведения $A \cdot B$. Если матрицы A и B нельзя перемножить, то оставить файл NameC пустым.
- 18.36. Даны два файла вещественных чисел с именами NameA и NameB, содержащие элементы прямоугольных матриц A и B (по строкам), причем начальный элемент каждого файла содержит количество столбцов соответствующей матрицы. Создать файл той же структуры с именем NameC, содержащий произведение $A \cdot B$. Если матрицы A и B нельзя перемножить, то оставить файл NameC пустым.

- 18.37. Дано целое число $N < 5$ и N файлов целых чисел разного размера с именами $Name_1, \dots, Name_N$. Объединить их содержимое в новом файле целых чисел с именем $Name_0$, используя следующий формат: в начальном элементе файла $Name_0$ хранится число N , в следующих N элементах хранятся размеры исходных файлов, а затем последовательно размещаются данные из каждого исходного файла.
- 18.38. Дан символьный файл, содержащий по крайней мере один символ пробела. Удалить все его элементы, расположенные после первого символа пробела, включая и сам этот пробел.
- 18.39. Дан символьный файл. Упорядочить его элементы по возрастанию их кодов.
- 18.40. Дано число k и строковый файл с именем $Name_1$, содержащий непустые строки. Создать два новых файла: строковый с именем $Name_2$, содержащий первые k символов каждой строки исходного файла (если строка короче k символов, то она сохраняется целиком), и символьный с именем $Name_3$, содержащий k -й символ каждой строки (если строка короче k , то в файл $Name_3$ записывается пробел).
- 18.41. Дан строковый файл, содержащий непустые строки. Создать новый файл, содержащий все строки

исходного файла наименьшей длины (в том же порядке).

- 18.42. Дан строковый файл с именем NameS, содержащий даты в формате "день/месяц/год", причем под день и месяц отводится по две позиции, а под год — четыре. Создать файлы целых чисел с именами Name1 и Name2, содержащие соответственно значения дней и месяцев для дат из исходного строкового файла (в том же порядке).
- 18.43. File50. Дан строковый файл, содержащий даты в формате "день/месяц/год", причем под день и месяц отводится по две позиции, а под год — четыре. Создать новый строковый файл, в котором даты из исходного файла располагались бы в порядке убывания.

19.Текстовые файлы

- 19.1. Дан текстовый файл. Вывести количество содержащихся в нем символов и строк (маркеры концов строк EOLN и конца файла EOF при подсчете количества символов не учитывать).
- 19.2. Дана строка S и текстовый файл. Добавить строку S в начало файла.
- 19.3. Дан текстовый файл. Удалить из него первую строку.
- 19.4. Даны два текстовых файла с именами $Name1$ и $Name2$. Создать новый текстовый файл с именем $Name3$, являющийся объединением содержимого файлов $Name1$ и $Name2$ (в указанном порядке).
- 19.5. Даны два текстовых файла с именами $Name1$ и $Name2$. Добавить в конец файла $Name1$ содержимое файла $Name2$.
- 19.6. Дан текстовый файл, содержащий более трех строк. Удалить из него три последние строки.
- 19.7. Дано число $k < 10$ и текстовый файл, содержащий более k строк. Удалить из файла последние k строк.
- 19.8. Дано число $k < 10$ и текстовый файл, содержащий более k строк. Создать новый текстовый файл, содержащий k последних строк исходного файла.

- 19.9. Дано число k и текстовый файл. Удалить из файла строку с номером k (строки нумеруются от 0). Если строки с таким номером нет, то оставить файл без изменений.
- 19.10. Дано число k и текстовый файл. Вставить пустую строку после строки с номером k (строки нумеруются от 0). Если строки с таким номером нет, то оставить файл без изменений.
- 19.11. Дан текстовый файл. Удалить из него пустые строки.
- 19.12. Дан текстовый файл. Заменить в нем все подряд идущие пробелы на один пробел.
- 19.13. Дан текстовый файл, содержащий текст, выровненный по левому краю. Выровнять его по центру, добавив в начало каждой непустой строки необходимое количество пробелов (ширину текста считать равной 50). Строки нечетной длины перед центрированием дополнять слева пробелом.
- 19.14. Дан текстовый файл. Найти количество абзацев в тексте, если абзацы отделяются друг от друга одной или несколькими пустыми строками.
- 19.15. Дан текстовый файл. Найти количество абзацев в тексте, если каждый абзац начинается с красной строки (5 пробелов). Пустые строки между абзацами не учитывать.

- 19.16. Дан текстовый файл. Абзацы выделяются в нем с помощью красной строки (5 пробелов), а пустых строк нет. Вставить между соседними абзацами по пустой строке.
- 19.17. Дан текстовый файл. Создать символьный файл, содержащий все знаки препинания, встретившиеся в текстовом файле (в том же порядке).
- 19.18. Дан текстовый файл. Вывести первое слово текста наибольшей длины (с учетом знаков препинания, расположенных в начале и в конце слов).
- 19.19. Дан символ C (прописная русская буква) и текстовый файл. Создать строковый файл, содержащий все слова из исходного файла, оканчивающиеся этой буквой (как прописной, так и строчной). Знаки препинания, расположенные в начале и в конце слов, не учитывать. Если исходный файл не содержит подходящих слов, оставить результирующий файл пустым.
- 19.20. Дано число N и текстовый файл. Удалить из файла абзац с номером N (абзацы отделяются друг от друга одной или несколькими пустыми строками и нумеруются от 1). Пустые строки, предшествующие и следующие за удаляемым абзацем, не удалять. Если абзац с данным номером отсутствует, то оставить файл без изменений.

- 19.21. Дано число N и текстовый файл. Удалить из файла абзац с номером N (абзацы выделяются с помощью красной строки (5 пробелов) и нумеруются от 1). Пустые строки между абзацами не учитывать и не удалять. Если абзац с данным номером отсутствует, то оставить файл без изменений.
- 19.22. Дан текстовый файл, каждая строка которого изображает целое число, дополненное слева и справа несколькими пробелами. Вывести сумму этих чисел и их количество.
- 19.23. Дан текстовый файл, каждая строка которого содержит изображения нескольких вещественных чисел, разделенных пробелами. Создать файл вещественных чисел, содержащий эти числа в том же порядке.
- 19.24. Даны два текстовых файла с именами $Name1$ и $Name2$. Добавить в начало каждой строки файла $Name1$ соответствующую строку файла $Name2$. Если файл $Name2$ короче файла $Name1$, то оставшиеся строки файла $Name1$ не изменять.
- 19.25. Дан текстовый файл $NameT$ и файл целых чисел $NameN$. Добавить в начало каждой строки файла $NameT$ изображение соответствующего числа из файла $NameN$. Если файл $NameN$ короче файла

NameT, то оставшиеся строки файла NameT не изменять.

- 19.26. Дан текстовый файл. Создать символьный файл, содержащий все символы, встретившиеся в тексте, включая пробел и знаки препинания (без повторений). Символы располагать в порядке возрастания их кодов.
- 19.27. Дана строка K , состоящая из 10 цифр, и файл с русским текстом. Зашифровать файл, выполнив циклическую замену каждой русской буквы, стоящей на i -й позиции строки, на букву того же регистра, расположенную в алфавите на $K[i]$ -м месте после шифруемой буквы (символы строки K также перебираются циклически: для $i = 11$ снова используется смещение $K[1]$ и т.д.). Букву "ё" в алфавите не учитывать, знаки препинания и пробелы не изменять.

