

Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра электронных вычислительных машин

Е.М. Демидович

СБОРНИК ЗАДАНИЙ

по курсу

ОСНОВЫ АЛГОРИТМИЗАЦИИ
И ПРОГРАММИРОВАНИЯ

для студентов специальности
«Вычислительные машины, системы и сети»
дневной и вечерней форм обучения

Минск 2003

УДК 681.322 (075.8)
ББК 32.973 я 73
Д30

Демидович Е.М.

Д30 Сборник заданий по курсу «Основы алгоритмизации и программирования» для студентов специальности «Вычислительные машины, системы и сети» дневной и вечерней форм обучения / Е.М. Демидович. — Мн.: БГУИР, 2003. — 51 с.

ISBN 985-444-459-7.

В сборнике приведены задания по темам: обработка одномерных и многомерных массивов с числовой и символьной информацией, использование указателей, информации командной строки и рекурсий, создание и обработка динамических списков различного типа и файлов.

УДК 681.322(075.8)
ББК 32.973 я 73

Введение

Приведенные в сборнике задания соответствуют содержанию курса «Основы алгоритмизации и программирования».

Задания могут быть использованы для разработки схем алгоритмов, которые затем будут реализованы в программах. В начале сборника приводятся задания для разработки простых программ, элементы которых в дальнейшем будут использованы в более сложных заданиях: это поиск элементов, их сортировка, обработка одномерных и многомерных массивов данных.

В последующих заданиях предполагается знание более сложных элементов алгоритмического языка: функций, указателей на них, рекурсий, списков.

При использовании структурированных данных во многих заданиях явно не указываются их тип (`class`, `struct`, и др.), какие элементы являются открытыми или закрытыми, какие конструкторы и методы должны быть реализованы. Проектирование классов и структуры программы выполняют студенты. Предполагается, что в заданиях последних тем студенты реализуют дружественный интерфейс: меню, всплывающие окна, необходимые подсказки, используют оптимальное сочетание цвета.

С учётом уровня подготовки студентов задания им могут выдаваться в другой последовательности.

1. Обработка одномерных массивов данных

1.1. Сколько чисел между 100 и 10000 состоят из нечетных цифр, сколько — из различных цифр?

1.2. Сколько палиндромов длиной «n» можно образовать, используя двадцать шесть букв? (Палиндром — выражение, которое читается одинаково как слева направо, так и наоборот).

1.3. Получить все «счастливые» номера от 0 до 999999. Номер является «счастливым», если сумма первых трех цифр равна сумме трех его последних цифр. Если в числе меньше шести цифр, то недостающие начальные цифры считаются нулями.

1.4. В диапазоне от 1 до 10000 найти все целые числа, квадраты которых являются палиндромами.

1.5. Дан массив из десяти элементов. Определить, рассортирован ли этот массив по возрастанию. Элементы массива ввести.

1.6. Определить, рассортирован ли массив вещественных чисел по убыванию. Элементов в массиве не более тридцати. Размер массива и числа ввести.

1.7. В массиве вещественных чисел поменять местами максимальный отрицательный элемент и минимальный положительный. Количество элементов в массиве не более двадцати. Элементы и их количество ввести.

1.8. В массиве вещественных чисел среди чисел, стоящих на четных местах, определить минимальный положительный элемент массива и его индекс. Количество элементов в массиве не более двадцати. Элементы и их количество ввести.

1.9. В массиве целых чисел среди чисел, стоящих на нечетных местах, определить максимальный отрицательный элемент массива и его индекс. Количество элементов в массиве не более двадцати. Элементы и их количество ввести.

1.10. В массиве вещественных чисел поменять местами минимальный отрицательный и максимальный положительный элементы. Количество элементов в массиве не более двадцати пяти. Элементы и их количество ввести.

1.11. Определить сумму элементов, имеющих нечётные значения и стоящих на чётных местах. Количество элементов в массиве не более пятнадцати. Элементы и их количество ввести.

1.12. В массиве целых чисел определить сумму элементов, стоящих на чётных позициях, и произведение элементов, имеющих чётные значения. Количество элементов в массиве не более двадцати. Элементы и их количество ввести.

1.13. В массиве определить количество элементов, кратных максимальному элементу. Количество элементов в массиве не более двадцати. Элементы и их количество ввести.

1.14. В массиве целых чисел вывести значения элементов, стоящих после элементов со значением, кратным пяти. Размер массива и элементы ввести.

1.15. В массиве вещественных чисел определить пять максимальных элементов. Размер массива и числа ввести.

1.16. В массиве целых чисел заменить нулями отрицательные элементы, стоящие после положительных. Элементов в массиве не более двадцати. Конкретный размер массива и элементы ввести.

1.17. В массиве вещественных чисел утроить значения положительных элементов, расположенных перед отрицательными элементами. Элементов в массиве не более пятнадцати. Конкретный размер массива и элементы ввести.

1.18. Выдать на экран все пятизначные числа, состоящие из несовпадающих цифр.

1.19. Составить программу для обучения переводу чисел из десятичной системы счисления в двоичную систему и обратно.

1.20. Известно, что любое натуральное число можно представить в виде суммы не более чем четырёх квадратов целых неотрицательных чисел. Дано натуральное « n ». Указать такие неотрицательные целые числа, сумма квадратов которых равна « n ».

1.21. Написать программу, которая определяла бы следующие значения:

- а) размер целого числа в байтах;
- б) размер длинного целого числа в байтах;

- в) размер вещественного числа в байтах;
- г) размер двойной точности в байтах;
- д) размер указателя в байтах.

1.22. В соревновании принимают участие восемь спортсменов. Сколько существует способов разделения медалей (золотые, серебряные, бронзовые)?

1.23. Сколько существует возможных результатов, которыми могут закончиться соревнования, в которых стартует десять человек в трех видах спорта, если каждый человек участвует в одном, произвольно выбранном виде? (Под результатом соревнования понимается распределение мест для всех спортсменов, стартующих в каждом виде).

1.24. Построить треугольник Паскаля из «n» строк. По краям строк стоят единицы, а каждое число внутри равно сумме двух, стоящих сверху в ближайших позициях строки:

$$\begin{array}{cccc} & & 1 & & \\ & 1 & 2 & 1 & \\ 1 & 3 & 3 & 1 & \end{array}$$

1.26. Вычислить и отобразить на экране таблицу значений функции

$$\sum \frac{2+n}{n!} x^n$$

от 0 до 1 с шагом 0,02.

1.27. Имеется последовательность чисел 123456789. Проставить между ними знак «+» или знак «-» таким образом, чтобы она была равна заданному числу.

2. Обработка многомерных массивов данных

2.1. Ввести матрицу чисел и транспонировать её. Определить строки, в которых значения элементов убывают, и столбцы, в которых они возрастают.

2.2. Дана матрица 10x10. Вычислить сумму тех ее элементов, расположенных на главной диагонали и выше, которые превосходят все элементы, расположенные ниже главной диагонали. Если таких элементов нет, то выдать соответствующее сообщение.

- 2.3. Составить программу, которая выводит значения элементов матрицы, образованной из исходной матрицы, если вычеркнуть все строки и столбцы, содержащие хотя бы один нулевой элемент.
- 2.4. Дана действительная квадратная матрица порядка «n». Рассмотрим те элементы, которые расположены в строках, начинающихся с отрицательного элемента. Найти суммы тех из них, которые расположены соответственно ниже, выше и на главной диагонали.
- 2.5. Транспонировать матрицу. Ввод, вывод, транспонирование производить в функциях. Обработать «к» матриц целого типа размером не более 20x20.
- 2.6. Определить сумму элементов, расположенных в чётных столбцах матрицы ниже побочной диагонали. Размеры матрицы не более 20x20. Конкретные размеры и элементы матрицы ввести. Элементы — целые числа.
- 2.7. Определить максимальный элемент и его позицию в каждой строке матрицы вещественных чисел. Размеры матрицы не более 30x10. Конкретные размеры и элементы матрицы ввести.
- 2.8. Определить максимальный элемент и его позицию среди элементов, расположенных выше главной диагонали матрицы. Размеры матрицы не более 10x10 элементов. Конкретные размеры и элементы матрицы ввести.
- 2.9. Поменять местами максимальный элемент среди всех отрицательных элементов матрицы на минимальный элемент среди всех положительных. Размеры матрицы вещественных чисел не более 20x20. Конкретные размеры и элементы матрицы ввести.
- 2.10. Рассортировать положительные элементы каждой строки матрицы по убыванию. Отрицательные элементы оставить на своих местах. Размеры матрицы целых чисел не более 10x10. Размеры матрицы и её элементы ввести.
- 2.11. Рассортировать отрицательные элементы каждого столбца матрицы по возрастанию. Положительные элементы оставить на своих местах. Размеры матрицы целых чисел не более 20x10. Размеры матрицы и её элементы ввести.
- 2.12. Определить количество элементов, равных минимальному эле-

менту матрицы и расположенных выше главной диагонали. Размеры матрицы вещественных чисел не более 10×10 . Конкретные размеры и элементы матрицы ввести.

2.13. В матрице целых чисел определить максимальный элемент на главной диагонали, и есть ли такой среди элементов ниже главной диагонали. Размеры матрицы не более 10×10 . Конкретные размеры и элементы матрицы ввести.

2.14. Определить минимальный положительный элемент среди элементов, расположенных ниже главной диагонали матрицы. Размеры матрицы не более 10×10 . Конкретные размеры и элементы матрицы ввести.

2.15. В матрице целых чисел элементы главной диагонали записать в обратном порядке. Размеры матрицы не более 15×15 . Конкретные размеры и элементы матрицы ввести.

2.16. Даны две действительные квадратные матрицы порядка N . Получить новую матрицу:

а) умножением элементов каждой строки первой матрицы на наибольшее из значений элементов соответствующей строки второй матрицы;

б) прибавлением к элементам каждого столбца первой матрицы произведения элементов соответствующих строк второй матрицы.

2.17. В матрице вещественных чисел вывести на экран номера элементов со значением, кратным семи, и находящихся в строках с нечетными номерами и столбцах с четными номерами. Размеры матрицы не более 15×10 . Конкретные размеры матрицы и элементы ввести.

2.18. Определить сумму положительных элементов, расположенных выше главной диагонали матрицы. Размеры матрицы не более 10×10 . Конкретные размеры и элементы матрицы ввести. Элементы — целые числа.

2.19. Определить минимальный элемент и его позицию в каждом столбце матрицы вещественных чисел. Размеры матрицы не более 10×30 . Конкретные размеры и элементы матрицы ввести.

2.20. Дана действительная квадратная матрица порядка n . Построить последовательность действительных чисел $a[1], a[2], a[3], \dots, a[n]$ по правилу: если в i -й строке матрицы элемент, принадлежащий главной диагонали, отрицателен, то $a[i]$ равно сумме элементов i -й строки, предшествующих первому отрицательному элементу, иначе $a[i]$ равно сумме по-

следних элементов i -й строки, начиная с первого по порядку неотрицательного элемента.

2.21. Дана матрица размером 10×10 . Вычислить сумму элементов, расположенных на главной диагонали и выше, которые превосходят все элементы, расположенные ниже главной диагонали. Если таких элементов нет, то выдать соответствующее сообщение.

2.22. Дано « n » матриц, каждая матрица размерностью не более 5×10 . Используя указатели, рассортировать элементы столбцов матрицы следующим образом: первый столбец — по возрастанию, второй — по убыванию и т.д.

2.23. Задана последовательность целых чисел $a[1], a[2], \dots, a[64]$. Построить целую квадратную матрицу восьмого порядка, которая будет заполнена по следующему правилу:

```
a[ 1] a[ 3] a[ 4] a[10] a[11] a[21] a[22] a[36]
a[ 2] a[ 5] a[ 9] a[12] a[20] a[23] a[35] a[37]
a[ 6] a[ 8] a[13] a[19] a[24] a[34] a[38] a[49]
a[ 7] a[14] a[18] a[25] a[33] a[39] a[48] a[50]
a[15] a[17] a[26] a[32] a[40] a[47] a[51] a[58]
a[16] a[27] a[31] a[41] a[46] a[52] a[57] a[59]
a[28] a[30] a[42] a[45] a[53] a[56] a[60] a[63]
a[29] a[43] a[44] a[54] a[55] a[61] a[62] a[64]
```

3. Обработка массивов строковых данных

3.1. Написать программу, преобразующую текст, состоящий только из прописных букв, в текст, состоящий из прописных и строчных букв. Первая буква после точки — прописная, остальные — строчные.

3.2. Введите текст, заканчивающийся знаком *. Каждое предложение заканчивается точкой. В функции в каждом предложении поменять местами самое длинное и самое короткое слово. Глобальные переменные, библиотечные функции, кроме функций ввода-вывода, не использовать.

3.3. Дано « n » матриц. Элементы матрицы — строки. Подсчитать сумму кодов символов каждого слова и, если сумма оказалась четной, развернуть зеркально это слово в строке. Полученные матрицы вывести.

3.4. Дан массив строк. Все русские буквы привести к верхнему регистру, латинские — заменить «?», все одинаковые буквы (русские) в слове — удалить. Вывести результат на экран.

3.5. Дан текст. Если в нем нет малых латинских букв, то оставить его без изменения, в противном случае каждый из символов, следующих за первой группой малых латинских букв, заменить точкой.

3.6. Обработать массив строк (длина каждой не более 255 символов). Если в строке встречена последовательность одинаковых символов, заменить их кодом 255, за которым следуют код этого символа и количество одинаковых символов.

3.7. Дан массив символов — текст. Каждый символ текста может быть малой буквой, цифрой или одним из знаков: «+», «-», «*». Группой букв называется совокупность последовательно расположенных букв, которой непосредственно не предшествует и за которой не следует буква. Аналогично определяется группа цифр и знаков. Выяснить, действительно ли, что в тексте больше групп букв, чем групп знаков. Найти самую длинную группу цифр, если эту наибольшую длину имеют несколько групп, то взять первую по порядку.

3.8. Обработать «к» массивов длинных чисел — строк. В одном массиве содержится не более ста чисел-строк. В первой функции осуществить их ввод. Во второй функции получить массив этих чисел — строк, записанный в порядке возрастания частоты их встречаемости в исходном массиве. Полученный массив вывести в головном модуле, используя функцию puts(). В третьей функции получить массив этих чисел — строк, записанный в порядке возрастания значимости соответствующих чисел, полученный массив также вывести в головном модуле, используя функцию puts(). Глобальные переменные не использовать. Реализовать свои функции преобразований.

3.9. С клавиатуры задается текст. Длина текста не более 10 000 знаков. Признак конца текста точка. Текст состоит из предложений. Длина каждого предложения задается его первым словом. Слова в предложении разделены одним или несколькими пробелами. Слова могут состоять только из цифр или букв. Количество предложений не более ста. Длина предложения не более двухсот пятидесяти знаков. Необходимо:

- а) найти сумму чисел, входящих в каждое предложение, и записать её в массив. Затем вывести полученный массив чисел;
- б) рассортировать предложения по алфавиту.

3.10. В головном модуле необходимо обработать «к» матриц. Элементы матрицы — строки, длина строки не более восьмидесяти знаков. Строка состоит из фамилии, имени, отчества. Размеры матрицы не более 10x20. Конкретные размеры матрицы ввести. В функции в каждой матри-

це определить последние по алфавиту пять строк и вывести их в головном модуле. До вывода новой матрицы исходную матрицу сохранить, новых массивов строк не создавать. Все элементы матрицы сортировать не рекомендуется. Глобальные переменные и системные функции, за исключением ввода/вывода, не использовать.

3.11. На входе имеется «n» матриц. Элементы матрицы — строки, размеры матрицы не более 10x20, длина строки не более ста знаков. Из строк надо удалить все палиндромы (слова, которые слева и справа читаются одинаково). Выполнить всё на месте старой строки, новых массивов не создавать, количество пробелов между словами сохранить. Глобальные переменные, системные функции для работы со строками не использовать.

3.12. Написать программу, преобразующую текст, состоящий только из прописных букв, в текст, состоящий из прописных и строчных букв. Первая буква после точки — прописная, остальные — строчные.

3.13. Обработать текст, заканчивающийся *. Каждое предложение заканчивается точкой. В функции в каждом предложении поменять местами самое длинное и самое короткое слово. Слова разделены одним или несколькими пробелами. Рассортировать в алфавитном порядке множество всех коротких и длинных слов, найденных в тексте. Глобальные переменные, библиотечные функции, кроме функций ввода-вывода не использовать.

3.14. В головном модуле выполнить ввод «n» строк знаков, «n» ввести с клавиатуры. Каждая строка заканчивается точкой. В строке не более двухсот знаков, строка состоит не более чем из тридцати слов. Слова в строке разделены одним или несколькими пробелами. В функции преобразовать строку, разделяя слова двумя пробелами, и вывести строку на экран не более чем по шестьдесят символов в строке, не допуская переноса слов; строка на экране не должна начинаться с пробела. Библиотечные функции не использовать. Дополнительные массивы не вводить.

3.15. Дан массив строк. Все русские буквы привести к верхнему регистру, латинские — заменить «?», все одинаковые буквы (русские) в слове удалить. Вывести результат на экран.

3.16. Обработать «n» матриц. Размеры матрицы не более 10x20, элементы — строки, состоящие из цифр. Используя в работе только один указатель на строку, определить строки с четным количеством единиц. Вывести на экран эти строки, начиная с предпоследнего байта по первый. Функциями обработки строк не пользоваться. Новых переменных не вводить. Общее количество единиц не подсчитывать.

3.17. Дано «n» предложений. Длина предложения не более ста пятидесяти знаков. Необходимо сформировать массив указателей на первые по алфавиту десять слов из этих предложений. В этих словах младшая шестнадцатеричная цифра кода последнего знака слова — пять. Передать в функцию массив указателей и вывести в ней слова на экран, длина слова не более семи знаков.

3.18. Заданы натуральное число $n \leq 1000$ и символ, указывающий падеж. Вывести на экран введенное число русскими словами в заданном падеже.

3.19. Вводится массив строк. Каждую строку обработать так, чтобы все слова, состоящие только из цифр, были удалены, а их сумма стала последним словом. Дополнительные строки или массивы не создавать, стандартные функции работы со строками не использовать.

3.20. Обработать «n» матриц. Элементами матриц являются строки. Рассортировать строки матриц по возрастанию сумм кодов всех символов строки. Для упрощения можно дополнительно выделить поле для занесения в него суммы. Размеры матрицы не более 10x10, строка не более пятидесяти знаков.

3.21. В массиве строк найти среднее значение длины строки. Строки, длина которых больше среднего, — обрезать, меньше — добавить пробелы. Вывести полученные строки.

3.22. Обработать «n» матриц (n вводится с клавиатуры). Размеры матриц не более 10x15. Конкретные размеры матриц ввести. Каждый элемент матрицы — строка длиной не более ста знаков. Необходимо определить:

а) в каждом предложении количество слов, которые слева и справа читаются одинаково (палиндромы), и вывести в головном модуле на экран их количество;

б) первые по алфавиту десять палиндромов из каждой матрицы и вывести их в головном модуле.

Слова разделяются одним или несколькими пробелами, длина слов не более десяти знаков, палиндромов во всех матрицах не более пятидесяти. Глобальные переменные не использовать.

3.23. В функцию передаются размеры матрицы «m», «n» ($m \leq 20$, $n \leq 25$), два символа и матрица. Элементами матрицы являются слова. Удалить из матрицы все строки и столбцы, в которых находятся слова (хотя бы одно), включающие в себя заданные два символа. Сдвинуть строки и

столбцы к левому верхнему углу матрицы и вывести ее. Выполнить преобразования на месте исходной матрицы.

3.24. Даны две строки длиной по восемьдесят байт каждая. Используя в работе только одну переменную типа `int`, определить сходство строк. Если строки не идентичны, то вывести слова, начиная с которых строки различны. Любая строка обязательно начинается с пробела.

3.25. Программа, анализируя массив, состоящий из строк, выводит на экран количество слов женского, мужского и среднего рода.

3.26. Обработать «n» матриц размером не более 5x8. Элементы матрицы — строки длиной не более двухсот знаков. Строка содержит символьную и цифровую информацию. Слова могут состоять только из букв или только из цифр. На месте исходных матриц выполнить конкатенацию нецифровых слов, причем слова меньше четырех знаков удалить. Найти сумму всех чисел в строке и записать её в конец новой строки. Затем вывести на экран все элементы матрицы. Глобальные переменные, системные функции не использовать (кроме ввода/вывода).

3.27. Имеется массив строк. На месте исходного массива, не вводя других массивов, записать слова всех строк в обратном порядке, т.е. первое слово первой строки станет последним словом последней строки (если объединить строки, реализация проще). Необходимо учитывать, что в строке может быть недостаточно места для размещения очередного слова.

3.28. Дан массив строк, состоящих из слов. В словах есть только цифры 0–9. Преобразовать строки следующим образом: сложить первое и последнее, второе и предпоследнее слова и т.д. по цифрам, получив знаки новых чисел (или буквы «А», «В» и т.д.) Пример: «1206 381 667 80», получим: «9206 9E8». Если в одном из слов цифр меньше, недостающие разряды считать нулями.

3.29. Дан текст. Если в нем нет малых латинских букв, то оставить его без изменения, в противном случае каждый из символов, следующих за первой группой малых латинских букв, необходимо заменить точкой.

3.30. Задана матрица размером не более 15x15. Элементы матрицы — слова длиной до десяти символов. Удалить из матрицы строки и столбцы, содержащие символы «q» и «e».

4. Использование функций

4.1. В головном модуле даны массивы целых чисел. Известен адрес первого числа первого массива, другие массивы следуют друг за другом после первого массива. Первое число каждого массива задаёт его длину. Признаком конца всех массивов является нуль. В функции создать матрицу, первым элементом которой является число, входящее в какой-либо массив, второй элемент указывает, сколько раз число встретилось в исходных массивах. Всего разных чисел не более ста. Исходные массивы сохранить, дополнительных массивов, кроме матрицы, не создавать.

4.2. В функцию передаётся массив чисел. Определить пять элементов этого массива (исключая первый), сумма которых равна первому элементу массива. Вывести найденные числа в головном модуле на экран.

4.3. Ввести массив длинных целых чисел. Рассортировать по алфавиту массив строк, соответствующих записям этих чисел, и вывести эти числа в соответствии с полученным массивом. Ввод, преобразование, сортировку чисел выполнить в отдельных модулях. Глобальные переменные, системные функции, кроме функций ввода-вывода, не использовать.

4.4. Обработать «n» матриц (n задается с клавиатуры). Размеры матрицы не более 10x15. Каждый элемент матрицы — строка знаков длиной не более ста. В первой функции определить в каждом предложении количество слов, которые и слева и справа читаются одинаково (палиндромы). Во второй функции определить первые по алфавиту десять слов-палиндромов из каждой матрицы. Длина слова не более десяти знаков, в предложении слова разделены одним или несколькими пробелами. Палиндромов во всей матрице не более пятидесяти. Глобальные переменные, системные функции, кроме функций ввода-вывода, не использовать.

4.5. Дан указатель на текст, являющийся списком. Признак конца текста — знак «!». Текст состоит из предложений, которые содержат символьную и цифровую информацию. Первое слово предложения задает его длину, далее следуют фамилия, имя, отчество и зарплата по месяцам (их не более двенадцати).

В одной функции, используя указатели на фамилию, имя, отчество, рассортировать эти указатели в алфавитном порядке строк и вывести строки в головном модуле.

Во второй функции, используя эти указатели, рассортировать строки

по возрастанию общей суммы зарплаты и также вывести их в головном модуле.

Предложений не более двадцати пяти, длина предложения не более восьмидесяти знаков. Глобальные переменные и системные функции, за исключением функций ввода/вывода, не использовать, исходный текст сохранить.

4.6. В головном модуле ввести массив строк. Строк не более пятнадцати, длина строки не более восьмидесяти знаков. В первой функции, используя массив указателей на строки, осуществить ввод строк. Во второй функции найти строку, содержащую слово с наибольшим количеством знаков (самое длинное слово). Найденную строку вывести в головном модуле. В третьей функции самое длинное слово, используя рекурсию, вывести на экран, начиная с последнего символа. Все вызовы осуществить в головном модуле, глобальные переменные не использовать, написать свою функцию определения длины строки.

4.7. Обработать «n» матриц. Размеры матрицы не более 10x15. Элементы матрицы — строки символов, длина строки не более двадцати пяти знаков. В строке есть слово, состоящее только из цифр (например 1327671). Найти это слово, преобразовать в число и сохранить в массиве. Вывести на экран полученное число, а также количество чисел в матрице, значения которых больше тысячи. Ввод матрицы, поиск слова, преобразование его в число реализовать в отдельных функциях. Глобальные переменные не использовать.

4.8. В головном модуле определить матрицу, размеры которой 10x20. Элементами матрицы являются строки знаков. В первой функции ввести размеры матрицы и её элементы. Во второй функции все строки и столбцы матрицы, в которых хотя бы один из её элементов совпадает со строкой, введенной с клавиатуры, удалить. Заданная строка может повторяться в строке или столбце матрицы. Полученную матрицу уплотнить, сдвинув все её элементы к левому верхнему углу.

4.9. С клавиатуры задается «n» предложений. Каждое предложение длиной не более восьмидесяти знаков. Слова в предложении разделены одним или несколькими пробелами. Длина слова не более десяти знаков. Слов не более пятисот. В первой функции сформировать массив слов, у которых старшая шестнадцатеричная цифра кода последнего знака состоит из четного числа единиц. Во второй функции найти первые десять слов, у которых вычисленное количество двоичных единиц старшей шестнадцатеричной цифры кода последнего знака минимально. Найденные слова вывести в головном модуле. Глобальные переменные, системные функции не использовать.

4.10. Обработать «n» матриц целых чисел (типа int), задаваемых с клавиатуры. Размеры матрицы не более 10x15. Из всех элементов матрицы найти семь чисел, строковое (знаковое) представление которых по алфавиту будет первым. Зарезервировать место в оперативной памяти только под восемь строк. В головном модуле вводить матрицы и выводить найденные семь чисел (строк). В отдельных функциях реализовать перевод числа в строку, сравнение строк и, если требуется, сортировку строк. Глобальные переменные и системные функции не использовать.

4.11. В головном модуле ввести массив строк, размер массива не более 20x30. В первой функции, используя указатели на строки, ввести их. Во второй функции разбить строки на отдельные слова (выделить слова из строк). Известно, что слов не более семидесяти, а длина слова — не более семи знаков. В третьей функции рассортировать массив слов по их длинам. Рассортированный массив слов вывести в четвертой функции.

4.12. Обработать «n» матриц (n задается с клавиатуры). Размеры матрицы не более 15x10. Элементы матрицы — предложения, строки знаков длиной не более восьмидесяти. В первой функции определить все слова в предложении, состоящие только из цифр. Во второй функции определить сумму чисел во всех предложениях, составляющих строку матрицы. Сумму чисел каждой строки матрицы вывести в головном модуле. Рассортировать элементы каждой строки матрицы в алфавитном порядке и вывести в головном модуле. Глобальные переменные, системные функции, кроме функций ввода-вывода, не использовать.

4.13. В головном модуле необходимо обработать «K» массивов. Элементы массивов — строки знаков: фамилия, имя, отчество работника. Длина строки не более шестидесяти знаков. Размер массива не более 150 элементов. Конкретный размер ввести. Для каждого вышеуказанного массива вводится матрица зарплат этих работников. Первый элемент строки задаёт количество отработанных месяцев, а следующие — величину зарплаты за каждый месяц (не более двенадцати).

В одной функции в каждом массиве определить десять сотрудников, чьи фамилии будут последними в алфавитном порядке. До ввода нового массива исходный массив сохранить, новых массивов строк не создавать. Не рекомендуется сортировать все элементы массива. Во второй функции список из десяти сотрудников рассортировать по общей сумме полученных ими зарплат. Вывести список сотрудников (фамилия, имя, отчество), общую сумму зарплаты и зарплаты по месяцам. Глобальные переменные, системные функции, за исключением функций ввода/вывода, не использовать.

4.14. С клавиатуры задается вид работы: S — обработка строк, I — обработка целых чисел, F — обработка чисел с плавающей точкой, Q — выход.

Для S — с клавиатуры задается массив строк. Строк в массиве не более четырехста, длина не более восьмидесяти знаков, длина слова не более десяти знаков. На месте исходного массива создать новый массив, такой, чтобы в конце была расположена четвертая часть первых (последних) по алфавиту строк. Последовательность расположения остальных строк в массиве сохранить.

Для I и F — сделать так, чтобы были введены соответствующие числа и вычисления выполнялись по некоторой формуле.

4.15. В функции с клавиатуры вводится массив строк, являющихся изображением целых чисел. Строк не более двадцати пяти. Рассортировать строки в порядке убывания их числовых значений. Сортировку выполнить в отдельной функции, вывод результатов — в головном модуле. Глобальные переменные, системные функции не использовать.

4.16. В головном модуле определить три матрицы, размеры которых соответственно 5x8, 4x7 и 7x5. В первой функции за одно обращение к ней ввести указанные матрицы, используя одни и те же операторы. Во второй функции, так же за одно обращение к ней, найти сумму:

- а) элементов первой, третьей и пятой строк первой матрицы;
- б) элементов второй и третьей строк второй матрицы;
- в) элементов первой, третьей, четвертой, и седьмой строк третьей матрицы;
- г) общую сумму всех вышеперечисленных строк соответствующих матриц.

В третьей функции вывести названные строки в порядке увеличения сумм элементов этих строк.

4.17. В головном модуле вводится матрица. В функциях найти:

- а) сумму элементов каждой строки матрицы;
- б) максимальный и минимальный элементы в каждой строке и поменять их местами; в головной модуль вернуть найденные элементы и максимальный и минимальный элементы во всей матрице.

4.18. В головном модуле ввести матрицу целых чисел. Размеры матрицы не более 7x10. Используя функцию, рассортировать все элементы матрицы в порядке возрастания. Рассортированные элементы вывести в головном модуле.

4.19. В головном модуле дана матрица размером 10x10 элементов, элементы — целые числа. В функции записать построчно в виде вектора элементы, составляющие верхний треугольник матрицы (выше главной диагонали).

4.20. Дано «n» предложений. Длина предложения не более ста пятидесяти знаков. Необходимо сформировать массив указателей на первые по алфавиту десять слов из этих предложений (слов, в которых младшая шестнадцатеричная цифра кода последнего знака слова — пять). Передать в функцию массив указателей и вывести в ней слова на экран, длина слова не более семи знаков.

5. Использование указателей при работе с массивами данных

5.1. Обработать «n» матриц. Матрицы размером не более 10x20, элементы — строки, являющиеся записями чисел. Используя в работе только один указатель на строку матрицы, определить строки с четным количеством единиц. Вывести на экран эти строки, начиная с предпоследнего элемента по первый. Функциями работы со строкой не пользоваться, новые переменные не вводить. Общее количество единиц не подсчитывать.

5.2. Дано «n» матриц, размеры матриц не более 5x10. Используя указатели, рассортировать столбцы каждой матрицы следующим образом: первый столбец — по возрастанию, второй — по убыванию и т.д.

5.3. Ввести массив из «n» строк. Длина строки «n» знаков. Используя только указатели, транспонировать массив по диагонали.

5.4. Ввести массив строк. Удалить из каждой строки все слова (числа), состоящие только из цифр, а сумму этих чисел записать последним словом строки. Дополнительных строк или массивов не создавать, стандартные функции работы со строками не использовать.

5.5. Обработать «n» массивов строк. Число «n» задается с клавиатуры. Длина строки не более пятидесяти знаков. В функции рассортировать массив строк, перемещая указатели на них. Результат вывести в головном модуле.

5.6. Обработать «k» матриц. Число «k» задается с клавиатуры. Размерность матриц не более 15x15. Элементы матрицы — строки знаков длиной не более двадцати пяти символов. Создать матрицу указателей на строки. Строки разместить в динамической памяти. В функции рассортировать элементы побочной диагонали матрицы в алфавитном порядке, перемещая указатели на них. Написать свою функцию сравнения строк. Преобразованную матрицу вывести в головном модуле.

5.7. Создать матрицу указателей на строки. Строки размещены в динамической памяти; размеры матрицы не более 10x10. Конкретный размер матрицы ввести с клавиатуры. Длина строки не более сорока знаков. В функции рассортировать строки, принадлежащие главной диагонали матрицы, в порядке возрастания. Преобразованную матрицу вывести в головном модуле.

5.8. Ввести пять символьных строк, используя массив указателей на них. Найти строку, содержащую слово с наибольшим числом символов. Поиск реализовать в функции, в которую в качестве параметра передается указатель на строку. Найденное слово вывести на экран, начиная с последнего символа.

5.9. В головном модуле ввести «n» матриц (n задается с клавиатуры). Размеры матрицы не более 10x20. Элемент матрицы — строка знаков. Длина строки не более пятидесяти знаков. Используя функцию, рассортировать элементы каждой строки матрицы в алфавитном порядке, перемещая указатели на элементы матрицы. Глобальные переменные, системные функции, за исключением функций ввода/вывода, не использовать.

5.10. С клавиатуры задаются «n» предложений, длина предложения не более ста пятидесяти знаков. Необходимо сформировать массив указателей на первые по алфавиту десять слов, у которых младшая шестнадцатеричная цифра кода последнего знака слова пять. Передать в функцию массив указателей и вывести слова на экран. Длина слова не более восьми знаков. Глобальные переменные, системные функции не использовать.

6. Рекурсивное использование функций

6.1. Ввести текст, состоящий из «n» предложений (n не более 50). Длина предложения не более ста двадцати знаков. Слова разделены одним или несколькими пробелами. Записать без повторения в новый массив слова, встречающиеся в тексте более одного раза. Используя рекурсию, каждое слово в массиве записать в обратном порядке и затем массив рассортировать в алфавитном порядке. Исходный текст сохранить. Использовать один результирующий массив. В предложении не более пятисот таких слов. Длина слова не более восьми знаков. Глобальные переменные, библиотечные функции, кроме функций ввода-вывода, не использовать.

6.2. Получить все возможные перестановки из пяти символов.

6.3. С клавиатуры задается «n» значений x и ε . Вычислить с точностью ε значение y :

$$y_m = 1 / \left(m! + \frac{x^m}{(m-2)! - \frac{x^{m-2}}{(m-4)! + \frac{x^{m-4}}{1+x}}} \right)$$

О

$$5! + \frac{7! - \frac{x^7}{x^5}}{3! - \frac{x^3}{1+x}}$$

Вычисления выполнять до тех пор, пока $|y_m - y_{m-2}| > \varepsilon$. Формат вывода результатов x , ε и y задается с клавиатуры. В формат вывода могут входить слова, разделенные любым количеством пробелов. Ввод — вывод выполнить в головном модуле, вычисления — в функции. Глобальные переменные не использовать.

6.4. На входе имеется «n» предложений, представляющих собой арифметические выражения (длина выражения не более ста двадцати знаков). Создать массив, включающий в себя идентификаторы из всех выражений. Идентификаторов не более двухсот, длина идентификатора не более восьми знаков. Используя рекурсию, записать идентификаторы в обратном порядке и рассортировать их в алфавитном порядке. Глобальные переменные, библиотечные функции, кроме функций ввода-вывода, не использовать.

6.5. Дан массив чисел. Все числа, граничащие с цифрой «1», заменить нулями.

6.6. Ввести значение «n» и вычислить n!

7. Использование в функциях информации командной строки

7.1. В командной строке задаются размеры матрицы ($m, n \leq 100$). В первой функции ввести заданную матрицу (элементы — целые числа). Во второй функции на месте старой матрицы сформировать новую матрицу таким образом, чтобы в каждой строке сначала были записаны отрица-

тельные числа, затем — положительные числа и в конце — нули. Последовательность расположения чисел относительно друг друга среди положительных и отрицательных чисел сохранить. Вывод сформированной матрицы выполнить в головном модуле. Глобальные переменные, дополнительные массивы не использовать.

7.2. Вычислить значение тригонометрической функции. Название функции и значение аргумента передаются в качестве параметров функции `main()`.

7.3. Составить программу "Калькулятор", которая выполняла бы арифметические действия над операндами. Арифметическое выражение, являющееся аргументом функции `main()`, должно анализироваться посимвольно.

7.4. Аргументом командной строки является выражение в виде обратной польской записи, например: `20 10 5+* -> 20*(10+5)`. Вычислить значение выражения.

7.5. В командной строке задаются имена функций, которые требуется выполнить. Функции вызывать, используя указатель на них. При завершении программы вывести на экран, сколько раз данная функция выполнялась. Подсчет количества обращений к функции реализовать в самой функции. В функции ввести:

а) массив символов и определить символ, у которого будет наибольшее количество единиц в битовом представлении;

б) квадратную матрицу и рассортировать её элементы, расположенные выше главной диагонали, в порядке возрастания, ниже — в порядке убывания.

8. Обработка структурированных типов данных

8.1. В головном модуле ввести структуры. Элементами структуры являются вторая структура (фамилия, имя, отчество студента и номер семестра) и объединение. В объединении в зависимости от семестра записаны номера экзаменов и результаты, а список наименований экзаменов хранится отдельно. В зависимости от запроса в командной строке вывести информацию о студентах за определённый семестр.

8.2. В командной строке задаются два признака. Первый признак определяет тип вводимой информации: просто структура или структура, включающая объединение и имена функций, которые надо выполнить. Функции выводят информацию (из просто структуры или из структуры,,

включающей объединение), в зависимости от второго признака в командной строке.

В структурах хранится информация о студентах: фамилия, имя, отчество.

В объединении структура хранит один из типов информации:

- а) отец, мать, брат;
- б) отец, мать, брат, сестра;
- в) мать, брат, сестра.

8.3. В структурах хранится информация о студентах. Одним из элементов структуры является объединение, в котором в зависимости от места жительства (иностранец студент или нет) информация задаётся в виде:

- а) Минск, ул. ..., д. ..., кв. ...;
- б) область, город ..., ул. ..., д. ..., кв. ...;
- в) область, ... район, ... город, ул. ..., д. ... кв. ...;
- г) область, ... район, ... деревня ..., дом

По запросу из командной строки в зависимости от вида информации вывести данные о тех или иных студентах, используя указатели на функции.

8.4. Имеется массив структур о студентах. В структуре в качестве подструктуры задаются фамилия, имя, отчество студентов. Их медицинские параметры задаются в виде объединения в этой же структуре. По запросу из командной строки выдать информацию о студентах с соответствующими признаками.

Объединения включают:

- а) рост, вес;
- б) рост, вес, два — три других параметра.

8.5. В головном модуле в зависимости от запроса в командной строке выполнить ввод структуры или ее обработку в зависимости от ключа.

Ключи:

- /e — ввод структуры;
- /p — распечатать структуру;
- /x — поиск максимальной средней оценки;
- /n — поиск минимальной средней оценки.

Структура включает:

- а) фамилию студента, оценки;
- б) номер группы, номер семестра;
- в) оценки за экзамены. Список экзаменов зависит от номера семестра.

8.6. В головном модуле ввести массив структур. Одним из элементов

структуры является объединение. В нём хранится количество валюты, которым обладает человек. В зависимости от запроса в командной строке вывести: кто самый богатый, бедный, у кого больше долларов.

8.7. Для каждого факультета по каждой специальности подготовить приказы об окончании студентами высшего учебного заведения. Обработку информации по факультету прекратить, когда на запрос введено слово «Выход». Специальностей на факультете не более пяти, групп на специальности не более десяти, студентов в группе не более тридцати, количество сданных экзаменов не более двадцати пяти. Списки студентов по специальности вводятся по группам в алфавитном порядке.

В приказе первыми в алфавитном порядке следуют выпускники, получающие диплом с отличием (нет троек, четверок менее 25%), затем в алфавитном порядке следуют остальные выпускники данной специальности. Исходные массивы с данными сохранить, но выходных массивов, копирующих исходные данные, не создавать. Глобальные переменные, системные функции, кроме ввода/вывода, не использовать. Ввод данных, вывод, поиск отличников, расположение по алфавиту выполнить в отдельных функциях.

8.8. Обработать информацию о фирмах городов. Обработку прекратить, когда на запрос будет введено слово «end». Фирм в городе не более пятидесяти. Информация следующая:

- название фирмы (не более тридцати знаков);
- величина налогообложения (не более 1млн. р. — в виде строки);
- дата (месяц — в виде строки) последнего срока внесения налога;
- дата его фактического внесения (строка).

В одной функции ввести названия фирм, в другой — величину налога, в третьей — предельную дату внесения налога и дату, когда налог погашен (если не внесен, то вводится ноль). В головном модуле для заданной даты (месяц) вывести в алфавитном порядке пять фирм, имеющих максимальную задолженность. Глобальные переменные, системные функции, кроме функций ввода-вывода, не использовать. Исходный массив сохранить, новых массивов структур не создавать. Можно объявлять и вводить другую необходимую информацию.

8.9. В головном модуле вводится список сотрудников (не более 25): фамилия, имя, отчество и зарплата в виде чисел по месяцам (не более 12) для каждого сотрудника. В первой функции рассортировать список каждого отдела по алфавиту. Во второй функции для каждого отдела создать матрицу из полученного списка сотрудников. Строка матрицы соответствует одному сотруднику: сначала идет фамилия и инициалы, а затем — зарплата по месяцам. Полученную матрицу вывести на экран в го-

ловном модуле. Глобальные переменные, системные функции, за исключением ввода/вывода, не использовать.

8.10. Обработать информацию о студентах не более чем пяти факультетов. На каждом факультете не более пятидесяти групп, в группе не более тридцати человек. С клавиатуры отдельно по каждому факультету вводится информация о студентах в виде строки знаков: фамилия, имя, отчество, затем номер группы и оценки за экзамены. Длина строки не более восьмидесяти знаков, количество оценок может быть менее пяти. В первой функции сформировать списки по группам в алфавитном порядке. Во второй функции, используя созданные списки, к информации о студенте добавить его средний балл по сессии (два знака после точки). Список вывести на экран в головном модуле и сохранить в памяти. Глобальные переменные, системные функции, за исключением ввода/вывода, не использовать.

8.11. Реализовать программу премирования по каждому факультету десяти студентов, которые учатся на «4» и «5» и имеют наивысший средний балл. Обработку данных прекратить после того, как введено слово «вывод». Специальностей на факультете не более пяти, групп на каждой специальности не более десяти, студентов в группе не более тридцати, экзаменов не более пяти. Списки вводятся по алфавиту, кодировка оценок: 0 — неявка, 1 — недопуск, 2, 3, 4, 5 — остальные. Список к оплате для каждого факультета вывести в алфавитном порядке. Новых списков не составлять. Ввод, вывод, поиск данных реализовать в отдельных функциях. Исходный список сохранить.

8.12. В головном модуле обработать информацию о сотрудниках нескольких отделов: в первой функции вводятся фамилия, инициалы (ФИО), год рождения сотрудников, во второй функции — зарплата по месяцам для каждого сотрудника. Количество зарплат не более двенадцати. Если человек не работал, то зарплата за этот месяц равна нулю. В третьей функции рассортировать списки сотрудников вместе с зарплатой по алфавиту, вывести в головном модуле. В четвертой функции найти среднюю годовую зарплату каждого сотрудника; месяцы, когда они не работали, не учитываются. Вывести в головном модуле ФИО и среднюю зарплату каждого сотрудника в виде строки знаков. Обработку прекратить, когда введено слово «конец». Средняя зарплата — целое число.

8.13. С клавиатуры вводится список из «k» студентов: сначала идет ФИО, затем номер группы и оценки за экзамены. Длина строки не более восьмидесяти знаков. Количество оценок может быть не более пяти. В первой функции сформировать список по группам. Во второй функции каждую группу рассортировать в алфавитном порядке и вывести с ука-

занием среднего балла каждого студента. Глобальные переменные, системные функции, кроме ввода/вывода, не использовать.

8.14. Райисполком готовит списки оказания помощи малообеспеченным семьям, в список должно войти не более ста семей. Организации района (их не более пятидесяти) подают в райисполком списки (не более двадцати пяти человек) с информацией: фамилия, имя, отчество, суммарный доход, количество членов семьи. Списки составлены по алфавиту. Райисполком из всех списков определяет сто наиболее нуждающихся семей. На эти сто семей составляется по алфавиту свой список и в исходных списках отмечается, кому оказана помощь. Вывести по алфавиту список райисполкома и список каждой организации. Глобальные переменные, системные функции, кроме ввода/вывода, не использовать. Ввод, вывод, поиск и другие операции выполнять в отдельных функциях. Разрешается включать в данные свои дополнительные переменные.

8.15. Ввести список студентов по специальности, групп не более пяти, студентов в группе не более двадцати пяти. Списки вводятся по алфавиту. Информация о студентах: ФИО, домашний адрес, дата рождения. Всё вводится в виде символьных строк. Вывести в алфавитном порядке список студентов, которым исполнилось восемнадцать лет. Ввод, поиск, вывод, другие преобразования выполнить в отдельных модулях. Глобальные переменные, системные функции не использовать.

8.16. Обработать информацию о трех сотрудниках фирмы. Информация следующая: ФИО — строка не более пятидесяти знаков, в виде матриц сведения о заработной плате сотрудников: первого — за пять лет по десять месяцев, второго — за семь лет по пять месяцев, третьего — за четыре года по семь месяцев. В первой функции за одно обращение к ней одним и тем же циклом ввести информацию о сотрудниках. Во второй функции, также за одно обращение к ней, найти суммарную зарплату сотрудников: за первый, второй, пятый годы первого сотрудника; за первый, второй, пятый, седьмой годы — второго сотрудника; за первый, четвёртый годы — третьего сотрудника, а также их суммарную зарплату за упомянутые годы. В третьей функции вывести фамилии сотрудников, заработную плату каждого из них за указанные годы в порядке возрастания всех найденных сумм.

8.17. В головном модуле имеется информация о студентах пяти групп, в каждой группе учится не более двадцати пяти человек (но для каждой группы количество учащихся своё). Информация о студентах следующая: указатель на строку с ФИО, строка с адресом местожительства, слово признаков: первый бит — минчанин или нет, второй бит — областной центр, третий бит — райцентр, четвёртый бит — сельская местность, пя-

тый бит — отличник, шестой бит — наличие троек, седьмой — наличие братьев, сестёр. В первой функции в каждой группе получить отдельно списки минчан-отличников, имеющих братьев или сестер, студентов из сельской местности, не имеющих братьев, сестёр. Каждый такой список включает информацию не более чем о семи студентах. Во второй функции рассортировать полученные списки и вывести их в головном модуле. Исходный список сохранить, глобальные переменные не использовать, реализовать свою функцию сравнения.

9. Связные списки. Стеки

9.1. Ввести «m», «n» ($n \leq 200$) и строку из «n» символов (строка оканчивается либо «.», либо «:», либо «;»). В строке среди знаков могут встречаться открывающиеся и закрывающиеся круглые, квадратные и фигурные скобки. Проверить, предшествует ли каждая открывающаяся скобка соответствующей закрывающейся, т. е. правильность расстановки скобок во всех «m» строках.

9.2. В командной строке заданы арифметические выражения, которые разделяются пробелами. В функции проверить правильность расстановки квадратных, круглых и фигурных скобок. В случае обнаружения ошибки в головном модуле вывести само выражение и во второй строке номера позиций, в которых обнаружено первое несоответствие скобок. Если скобки расставлены правильно, то вывести само выражение и сообщение: «Ошибок не обнаружено». Динамические структуры, индексы, глобальные переменные и библиотечные функции не использовать.

9.3. В командной строке задаются имена функций, которые необходимо выполнить. Одна из функций в качестве параметра получает указатель на очередь, содержащую массив указателей на стеки. Элементами стеков являются структуры. Путем слияния исходных стеков создать результирующий стек, упорядоченный в алфавитном порядке строк.

9.4. Имеется стек, элементами записи которого являются указатель на очередь и признак упорядочения очереди. Элементом записи очереди является фамилия студента. На экране с задержкой поочередно вывести состояние стека и очередей. При нажатии F1 вывод информации на экран приостанавливается и выполняется следующая операция: если «+» — добавляются элементы в очередь, если знак «-», то удаляются заданные элементы, «!» — наращивается стек.

9.5. Реализовать программу учёта занятых и свободных мест в общежитии, а также лиц, нуждающихся в общежитии. Одним из элементов созданной очереди является номер этажа, другим — общее количество

мест на этаже, количество свободных мест, список лиц, проживающих на этаже. Если свободных мест нет, то студент ставится в очередь на вселение на соответствующий этаж.

9.6. Имеется очередь с информацией о больницах. Элементами очереди являются количество всех мест в больнице, количество свободных мест и указатель на функцию, вычисляющую (условно) расстояние от больного до больницы. Программа ведет учет свободных мест и распределяет больных в ближайшие больницы. Реализовать операции приёма и выписки больного: если 0 — больной выписывается, 1 — поступает в ближайшую больницу, задаётся фамилия больного. По F4 — выдается состояние мест в больнице, по F5 — завершить выполнение программы.

9.7. Дана очередь, задающая операции обработки стеков. Элементами очереди являются номер стека и один из знаков: «d» — удалить элемент из стека, «a» — добавить. Элемент помещается в стек после элемента с номером, задаваемым с клавиатуры. Дан указатель на место положения добавляемого элемента. Добавляемый элемент — ФИО. По нажатию F1 на экране появляется запрос: добавить или удалить элемент в очереди; по F2 — ведётся обработка стеков; по F3 — завершить выполнение программы.

9.8. Имеется автостоянка с одним въездом. Вести работу стоянки: ставить машины на свободные места, выдавать машины по номеру. При отсутствии свободных мест организовать очередь, а при освобождении места — заполнять его машинами из очереди.

9.9. Имеется очередь, соответствующая 6 дням. Одним из элементов очереди является указатель на вторую очередь, соответствующую количеству учебных пар в день. Одним из элементов второй очереди является указатель на список наличия используемых аудиторий и количества рабочих мест в них. По запросу с клавиатуры выделить аудитории для проведения занятий, показать загруженность аудиторий; по F1 — завершить выполнение программы.

9.10. Имеется перечень (очередь) автостоянок города. В очереди указываются общее количество мест, количество свободных мест и список номеров машин на этой автостоянке. Обеспечить функционирование автостоянок. По клавише F1: если «+» появляется новая машина и ставится на свободное место, если «-», то машина убывает. Въезд — выезд со стоянки единственный. По клавише F2 создаётся новая автостоянка. При нажатии на F1 с клавиатуры задается номер автостоянки, на которую же-

лаем поставить автомобиль. При отсутствии мест машина ставится в очередь ожидания свободного места на соответствующей стоянке.

9.11. Дан стек. Одним из его элементов является указатель на очередь, элемент которой содержит фамилию. При нажатии любой клавиши, если количество значащих бит во введённом символе четное, то добавляется новый элемент в очередь; если количество значащих бит в символе нечетное, то заданный элемент удаляется. Элемент добавляется в ту очередь, на которой остановится указатель стека при вычитании по единице из кода символа до достижения нуля. По F1 вывести на экран состояние стека, по F2 завершить выполнение программы.

9.12. Дан указатель на очередь. Элементом записи очереди является массив указателей (не более десяти) на строки знаков. Длина строки — не более ста знаков. Слова в строке разделены произвольным количеством пробелов. Определить все строки, которые с начала и конца читаются одинаково, и вывести их без повторения в файл, имя файла задается с клавиатуры. Разрешается использовать функции работы с файлами.

9.13. Дан указатель на очередь, записи которой содержат указатель на кольцо и указатель на функцию. Запись кольца содержит размеры матрицы и указатель на матрицу, элементы матрицы целые числа.

На месте старой матрицы получить новую матрицу: вычеркнув из исходной все строки и столбцы, содержащие хотя бы один ноль, сдвинув строки и столбцы к верхнему левому углу. Для каждого кольца в командной строке, начиная с ее третьего элемента, задается имя файла, в который необходимо записать результирующую матрицу и ее размеры.

Из созданных файлов читать матрицы и, используя указатель на функцию, осуществлять транспонирование строк или столбцов матрицы. Все преобразованные матрицы и их размеры записать в первый файл командной строки. Вывести файл матриц на экран по пять чисел в строке, число печатать в поле из десяти позиций вплотную к левому краю. Перед печатью файла в первой строке выводить имя файла.

9.14. Дан указатель на очередь, записи которой содержат размеры (не более чем 20x20) матрицы и указатель на саму матрицу. Элементами матрицы являются строки знаков (не более двадцати знаков). Выбрать пять самых коротких строк. Каждую строку записать в отдельный файл, имена файлов задаются в командной строке. Матрицы сохранить и не дублировать.

9.15. Дан массив указателей на стеки, их не более пятнадцати. Элементы стеков могут быть двух типов: первый тип — включает фамилию, имя и дату рождения; второй тип — полученный из первого типа, допол-

ненный массивом чисел с плавающей точкой и числом, задающим количество элементов в массиве. Длина массива — не более двенадцати элементов. В функции рассортировать все элементы стека в алфавитном порядке фамилий, перемещая элементы стека. Содержимое стеков вывести в головном модуле. Написать свою функцию сравнения строк. Глобальные переменные не использовать.

10. Списки в виде кольца

10.1. В головном модуле дан стек, содержащий указатели на кольца. Записи кольца содержат или m , или ε , или и m и ε . Кроме предыдущей информации записи кольца содержат указатель на массив вещественных чисел, число, задающее длину этого массива, и n — степень извлекаемого корня. Необходимо из каждого вещественного числа массива извлечь корень n -й степени по нижеприведенной формуле. В командной строке задается, какую функцию использовать для вычисления корня: для одной функции задается количество выполняемых итераций (m), для второй — точность вычислений корня (ε). Глобальные переменные не использовать.

$$x = \sqrt[n]{a}; \quad x_{i+1} = \frac{x_i}{n^2}(n^2 - 1) + \frac{1}{2}(n + 1)\frac{a}{x_i^n} - \frac{1}{2}(n - 1)\frac{x_i^n}{n},$$

где $x_0 = a$, $|x_{i+1} - x_i| \leq \varepsilon$.

10.2. Дано кольцо. Одним из элементов записи кольца является указатель на стек, в котором посимвольно (в виде строки) записано арифметическое выражение с операциями «+», «-», «*», «/» над обыкновенными дробями. Вывести на экран в виде строки (текста) выражение, имеющее максимальное значение. Реализовать операции «/» и «-».

Например: $\frac{5}{6} - \frac{7}{12} = \frac{3}{12}$; $\frac{60}{6 \cdot 12} - \frac{42}{6 \cdot 12} = \frac{18}{6 \cdot 12} = \frac{3}{12}$.

Глобальные переменные, библиотечные функции не использовать.

10.3. Дана очередь. Одним из её элементов является указатель на кольцо. Реализовать следующие операции с указанными списками: в заданное кольцо подсоединить новый элемент; удалить; вывести на экран содержимое кольца; удалить элемент из очереди; элементы заданного кольца распределить по другим кольцам.

10.4. Дан указатель на кольцо. Элементами его являются: массив из

пяти указателей на функции, размеры матрицы и сама матрица. Матрица не более 50x25 элементов. Одна из функций преобразует матрицы таким образом, что в начале каждой строки идут отрицательные числа, затем положительные и нулевые. Последовательность расположения чисел относительно друг друга среди положительных и отрицательных чисел сохранить. В программе должна быть реализована эта функция. Глобальные переменные не использовать, новые массивы и другие структуры данных (стеков и т.д.) не создавать.

10.5. Дан указатель на кольцо. Одним из элементов записи кольца является указатель на стек, в котором посимвольно хранится имя функции. Вторым элементом записи кольца являются параметры функции, которыми могут быть или целое и вещественное числа, или два вещественных числа, или два указателя на строки, или целое, задающее количество массивов, и массив указателей на массивы целых чисел. Первый элемент каждого массива задает длину массива. В командной строке задаются имена функций, которые, используя указатель на функцию, необходимо выполнить. Если задан последний тип параметров функции, то необходимо найти частоту встречаемости чисел во всех массивах. Разных чисел не более пятисот. Реализовать данную функцию. Глобальные переменные, библиотечные функции не использовать.

10.6. Дано кольцо. Записи кольца содержат указатель на стек и символ. Нулевой бит символа указывает, рассортирован стек или нет, первый бит символа — признак упорядочения стека (возрастание или убывание). Записи стека содержат строку знаков и целое число, используемое для упорядочения стека. Получить на месте исходных стеков один общий упорядоченный стек, исключив записи с повторяющимися строками, оставив по одному экземпляру. Разрешается сортировать не рассортированные стеки. Для сравнения строк использовать свою функцию. Стек с одной связью.

10.7. Дано кольцо. Элементы кольца содержат указатель на стек. Элементы стеков рассортированы в лексикографическом убывании слов, в них содержащихся. Выполнить слияние стеков в один результирующий стек, располагая в нем слова в лексикографическом возрастании. Используя рекурсию, слова в результирующем стеке записать в обратном порядке на том же месте. Разрешается создать один результирующий стек, исходные стеки сохранить. Глобальные переменные и библиотечные функции не использовать.

10.8. Дано кольцо. Элементами записей кольца является указатель на функцию и указатель на список параметров этой функции. На экране должно отображаться состояние кольца с задержкой. При нажатии F1

вывод приостанавливается, указывается элемент кольца, на котором приостановилось движение, и запрашивается информация для выполнения функции. Если данные не вводятся, функция выполняется для стандартных значений.

10.9. Имеется кольцо. Элементы кольца — указатели на очереди. Каждая очередь рассортирована по некоторому признаку (признак сортировки содержится в очереди). При нажатии на F1 добавляется элемент в очередь, на которую показывает указатель; на F2 — удаляется заданный элемент; на F3 — общий упорядоченный список всех очередей выводится на экран.

10.10. Организовать справочную информацию о наличии свободных мест в гостиницах, для чего создать кольцо с двумя связями. В элементах кольца указать общее количество мест в гостинице и наличие свободных мест, третьим элементом кольца является список лиц, проживающих в гостинице. При нажатии на клавишу F1 и «+» — вселить человека в гостиницу, F1 и «-» — выселить, «?» — вывести список лиц, проживающих в гостинице. Выбираться элемент из кольца должен кратчайшим путем.

10.11. Дано кольцо, одним из элементов записи кольца является указатель на очередь вкладчиков. Элементами очереди являются сумма вклада, фамилия, имя и отчество вкладчика. Имеется стек магазинов. При нажатии на клавишу F2 завершить выполнение программы, на F1 — выполнить следующие операции в кольце: по клавише «↑» указанному вкладчику начисляется сумма, заданная с клавиатуры; по «↓» — снимается сумма и переводится в указанный магазин. В стеке магазинов должен вестись учет покупателей (ФИО, сумма).

10.12. Имеется кольцо, а в нём — указатели на функции, которые при выборе данного элемента кольца могут быть выполнены. Предусмотреть возможность наращивания кольца. Двигаясь по кольцу, при нажатии любой клавиши нужно показать функции, которые могут быть выполнены.

10.13. Имеется кольцо с двумя связями. При очередном нажатии на F1 изменяется направление движения по кольцу, на F2 — движение приостанавливается. Если затем нажать «?», то выводится содержимое кольца. К элементу кольца присоединена очередь из списка студентов. В списке указано, получает студент стипендию или нет. Ввести список выдачи стипендий студентам. При очередном нажатии на F3 и затем на знак «+» студенту добавляется некоторая сумма, если «-» — снимается. Если данный студент не находится в очереди, на которой остановился в данный момент указатель кольца, то выводится сообщение: «Такого студента нет».

10.14. Дано кольцо, содержащее указатели на стеки. Запись стека содержит: фамилию (не более пятнадцати знаков), имя (не более десяти знаков), отчество (не более двенадцати знаков), дату рождения, а также для мужчины — воинское звание и специальность, для женщины — количество детей. Записи стека рассортированы по возрастанию или убыванию фамилий. Признак сортировки задаётся в соответствующей записи кольца: «в» — возрастание, «у» — убывание. Для мужчин получить одну очередь, рассортированную в алфавитном порядке, путем слияния исходных стеков на том же месте (сортировки не использовать). Записи для женщин поместить в файл, имя которого задается в командной строке.

10.15. Дан указатель на кольцо. Элементом записи кольца является указатель на стек, где посимвольно хранится имя обрабатываемого файла (последний символ на вершине стека). В записях файла (их не более пятисот) хранятся фамилия, имя, отчество, жилая площадь и код длиной в один байт, первые четыре бита кода определяют номер института, следующие четыре бита — номер подразделения. В командной строке первый по порядку элемент определяет, что требуется выполнить: А — вывести записи каждого файла в алфавитном порядке фамилий; П — вывести в порядке возрастания площади. Второй по порядку элемент командной строки (двухзначное десятичное число) определяет, к какому подразделению должна относиться выводимая информация; первая цифра является номером института, вторая цифра — номером подразделения. Обработать все элементы командной строки.

10.16. Дана очередь. Одним из её элементов является указатель на кольцо, которое имеет две связи. В записях кольца — фамилии, рассортированные в алфавитном порядке. При вводе цифры кольцо, относящееся к соответствующему элементу очереди, отображается на экране, при этом если нажата ←, то от А до Я, если →, то от Я до А.

10.17. Дано кольцо. Записи кольца содержат указатель на массив указателей (их не более двадцати пяти). Каждая указатель массива задает местоположение слова. Слово начинается и оканчивается произвольным числом пробелов. Рассортировать слова по убыванию их длин (игнорируя пробелы). Системные функции и процедуры не использовать.

10.18. Дан указатель на кольцо. Записи кольца содержат указатель на стек, в котором посимвольно записано имя файла (снизу вверх), и указатель на кольцо. Каждая запись кольца содержит указатель на очередь, посимвольно хранящую строку знаков. Вывести в соответствующий файл указатели на строки, встречающиеся в кольце более од-

ного раза (без повторения строк). Исходные кольца сохранить, строки не дублировать.

10.19. Дано кольцо, его записи содержат указатель на стек. Элементом записи стека является число, используя которое, стеки упорядочены. В записи кольца указывается признак упорядочения чисел («В» — по возрастанию, «У» — по убыванию). Получить один упорядоченный по возрастанию стек путем слияния исходных стеков. Сортировки не использовать, результирующий стек получить на месте исходных стеков.

10.20. Дано кольцо с одной связью. Элемент кольца содержит: строку, являющуюся шифром устройства; указатель на строку с описанием данного устройства. Поменять местами элементы первой половины кольца на вторую. Новые массивы, списки не создавать.

10.21. Дано кольцо с одной связью. Элемент кольца содержит указатель на строку с названием прибора и указатель на текст с описанием прибора. На старом месте получить новое кольцо, в котором направление обхода поменять на противоположное. Новые массивы, списки не создавать.

10.22. Дано кольцо. Элементом кольца является указатель на стек, а элементом стека — указатель на текстовую информацию. Стеки с четными номерами (0, 2, 4, ...) объединить в одну цепочку, а с нечетными номерами — во вторую так, чтобы при обращении из кольца в (n-1)-й стек последовательно обходились все последующие стеки с четными или соответственно с нечетными номерами. Кольцо состоит, по крайней мере, из четырёх элементов. Стеки для некоторых элементов в кольце могут отсутствовать.

10.23. Дан массив указателей на кольца. В элементе кольца есть указатель на фамилию. В функции с переменным числом параметров (параметры — указатели на кольца) из каждого кольца вывести на экран без повторения фамилии, которые в одном кольце встречаются неоднократно. В командной строке задаётся, сколько колец просматривать, с какого кольца начать просмотр и шаг, на который увеличивается номер следующего просматриваемого кольца. Например: 2, 3, 5; 3, 1, 1; 5, 3, 2; 1, 2, 3; 3, 5, 7. Глобальные переменные, системные функции, за исключением функций ввода/вывода, не использовать.

10.24. Элементами кольца являются указатель на массив указателей на строки и число, задающее количество строк. Рассортировать строки в массиве в алфавитном порядке, а элементы кольца (после первой

сортировки) — в порядке возрастания первой строки массива (в алфавитном порядке).

10.25. Дано кольцо. Элементом кольца является указатель на стек, а элементом стека — указатель на текстовую информацию. Объединить стеки в единую цепочку так, чтобы при вхождении в $(n-1)$ -й стек последовательно проходились все последующие стеки.

10.26. Дан массив указателей на кольца, элементов в этом массиве не более пятидесяти, последний элемент — указатель NULL. Элементами кольца являются указатель на массив строк и число, задающее длину этого массива. В кольцах с чётными номерами массив строк в элементах на чётных местах рассортировать в порядке возрастания, а в кольцах с нечётными номерами массив строк в элементах на нечётных местах рассортировать в порядке убывания.

10.27. При заданных порядке расположения солдат в круге, числе « n » и известном солдате, с которого начинается отсчет, определить солдата, отправляющегося за помощью. Счет начинается с солдата с заданным именем. Когда счетчик достигает значения « n », солдат удаляется, отсчет начинается сначала со следующего солдата. Реализовать список в виде кольца, на экране отображать изменения в кольце.

11. Обработка списков в виде бинарных деревьев

11.1. Написать рекурсивную функцию поиска узла бинарного дерева с максимальным значением (число с плавающей точкой). Ключ дерева — число с фиксированной точкой.

11.2. В командной строке задан адрес вершины бинарного дерева в формате: сегмент: смещение (по четыре шестнадцатеричные цифры). В узлах дерева расположены целые числа. Создать два стека: в один поместить все четные числа, в другой — нечетные.

11.3. В командной строке задан адрес вершины бинарного дерева в формате: сегмент: смещение (по четыре шестнадцатеричные цифры). В узлах дерева расположены целые числа (0-50000). Записать в текстовый файл все числа, меньшие 3000, и удалить из структуры дерева.

11.4. В командной строке задан адрес вершины бинарного дерева в формате: сегмент: смещение (по четыре шестнадцатеричные цифры). В узлах дерева расположены строки до восьмидесяти символов. Создать стек и поместить в него все строки, содержащие хотя бы одну букву «А».

11.5. Параметрами функции являются указатели на массив и целое число, задающее количество записей в этом массиве. Функция должна создать сбалансированное бинарное дерево минимальной длины (сбалансированное бинарное дерево — это дерево, в котором длины ветвей могут отличаться не более чем на один элемент). В записях исходного массива есть ключ — целое число и строка длиной в сто знаков.

11.6. В узлах бинарного дерева имеется элемент, определяющий частоту обращения к нему. В записях дерева хранятся следующие данные: фамилия, имя, отчество работника, его адрес, место работы, должность, дата рождения. Создать новое бинарное дерево. В качестве ключа использовать частоту обращения к нему, путь к узлам дерева должен быть оптимальным (наикратчайшим). Записи из старого дерева в новое дерево не перемещать. Рекурсии и библиотечные функции не использовать. В дереве не более пятисот узлов. Частоты в узлах дерева не совпадают.

11.7. Создать бинарное дерево. Реализовать поиск элемента бинарного дерева в диалоговом режиме.

11.8. В узлах бинарного дерева имеется элемент, определяющий частоту обращения к нему. Создать новое дерево таким образом, чтобы путь к узлам был кратчайшим (рекурсии не использовать). В качестве ключа использовать частоту обращений к узлу. Частоты в узлах дерева не совпадают. В дереве не более семисот узлов.

11.9. Дана ссылка на бинарное дерево. Определить поддереву максимальной длины, в качестве ключа используется целое число.

11.10. Дано бинарное дерево, в качестве ключа используется строка, её длина не более пяти знаков. Записи дерева содержат указатель на слово, его длина не более пятнадцати знаков. В поддереву минимальной длины определить самое длинное слово.

11.11. Дано бинарное дерево, ключом является целое число. Записи бинарного дерева содержат строки знаков (не более пятидесяти знаков). Определить все симметричные поддеревья данного бинарного дерева и строки знаков, встречающиеся в нем более одного раза, вывести без повторения в файл. Имя файла ввести с клавиатуры.

11.12. Дано бинарное дерево, в качестве ключа используется строка знаков (не более восьми знаков). В записях дерева хранятся указатели на кольцо и на имя файла. В записях кольца есть указатели на строки знаков (длина не более двухсот), строка заканчивается точкой. Слова в строке разделены одним или несколькими пробелами. Определить частоту

встречаемости слов в каждом кольце. Сформировать новое дерево, используя частоту слова в качестве ключа и помещая в узел само слово. Слова и их частоту каждого кольца записать в соответствующий файл. Рекурсии, глобальные переменные, системные функции не использовать. Самое длинное слово в кольце не более десяти знаков.

11.13. Дано бинарное дерево, содержащее указатель на стек, в который посимвольно записано имя файла (не более десяти знаков), и указатель на кольцо. Записи кольца содержат указатель на очередь, посимвольно хранящую строку знаков. Вывести в соответствующий файл без повторения указатели на все строки, встречающиеся в кольце более одного раза. Записывать указатель на последний экземпляр повторяющейся строки.

11.14. Дана ссылка на бинарное дерево, записи которого содержат указатель на стек, в котором посимвольно хранится строка, являющаяся ключом бинарного дерева. Сформировать кольцо из двадцати пяти записей, в которых находятся первые в алфавитном порядке ключи бинарного дерева (ключ не более десяти знаков). Указатели на стеки, содержащие выбранные строки (ключи бинарного дерева), записать в файл. Имя файла ввести. Рекурсии и системные функции не использовать.

11.15. Дано бинарное дерево. Записи бинарного дерева содержат указатель на функцию и указатель на стек, хранящий строку знаков. Одна из функций определяет правильность расстановки скобок ($\{ [] \}$) в строке. Вторая функция переставляет самую длинную и самую короткую строки в соответствующих стеках.

11.16. Дано бинарное дерево, в качестве ключа используется строка знаков (длина не более десяти знаков). В записях дерева хранятся указатели на кольцо. В записях кольца есть указатели на строки знаков (не более ста знаков). Слова в строке разделены произвольным количеством пробелов. Определить частоту встречаемых слов во всех строках кольца. Вывести в файл слова и соответствующую им частоту. Имя файла для каждого кольца задается с клавиатуры. Глобальные переменные, системные функции не использовать, исходные строки сохранить. Самое длинное слово не более восьми знаков.

11.17. Дан стек. Элементами стека являются число N и указатель на бинарное дерево. В узле бинарного дерева записаны указатели на строку и на массив целых чисел. На N -м уровне каждого дерева найти его средний элемент (узел) и, используя рекурсию, на старом месте записать строку знаков в обратном порядке. Рекурсивно вывести строки с хвоста.

Узлов на одном уровне не более пятидесяти. Глобальные переменные не использовать.

11.18. Дано бинарное дерево. Определить количество узлов на N-м уровне бинарного дерева (N ввести с клавиатуры).

11.19. Дан массив бинарных деревьев. Элементами дерева являются фамилия, оценки, адрес, указатель на текст с характеристикой студента, признак — военнообязанный или нет. Сделать копии бинарных деревьев с целью воссоздания информации, если исходные деревья и вся принадлежащая им информация будут уничтожены. Структура новых деревьев должна полностью совпадать со структурами исходных деревьев. Рекурсивные функции и глобальные переменные не использовать.

12. Создание, копирование и корректировка файлов

12.1. На вход первой функции поступает последовательность целых длинных (четыре байта) чисел. Последнее число — 0. Чисел не более тысячи. В функции определить все числа, которые и слева и справа читаются одинаково. Записать в файл полученные в функции символьные представления этих чисел и рассортировать в файле в порядке возрастания их значений. Имя файла ввести в головном модуле. Во второй функции вывести представления этих чисел на экран, располагая их симметрично. Использовать только библиотечные функции работы с файлами. Глобальные переменные не использовать.

12.2. С клавиатуры вводится имя файла. Запись файла содержит ФИО и место жительства студента. Записать файл на дискету. Рассортировать записи в алфавитном порядке:

- а) в новый файл;
- б) в этом же файле.

Осуществить поиск указанного студента в файле информации.

12.3. С клавиатуры вводится имя файла, содержащего информацию о жителях района (ФИО, жилая площадь и количество человек). Написать программу, которая:

- а) сортирует файл по величине жилой площади, приходящейся на одного человека;
- б) добавляет, удаляет и выводит на экран информацию файла.

12.4. В записях файла заданы наименование сельскохозяйственной культуры, а также имя файла, в котором хранится информация о выращивании культуры, и имя файла, в котором хранится информация о

борьбе с вредителями данной культуры. В программе реализовать следующие операции:

- а) корректировку информации во всех трёх файлах;
- б) вывод по запросу требуемой информации.

12.5. Ведение и учет коммерческих точек. Создать файл, в котором хранятся сведения о всех точках:

- а) ассортимент товаров в каждой точке;
- б) стоимость и количество по каждому виду продаваемого товара;
- в) наличие разрешения на торговлю: время выдачи (дата), на какой срок, разрешение санитарно — эпидемиологической станции и др.

Предусмотреть возможности регистрации новых точек, удаления информации о закрытых точках, выдачи справочной информации, контроля срока действия разрешения.

12.6. Ведение и учет наличия хозяйственных товаров в магазине: учёт количества; список всех товаров; добавление количества нового типа; изъятие товаров; выдача справки о конкретном товаре.

12.7. В записях файла хранятся наименование магазина, его номер, адрес, а также имя файла, в котором хранится информация об имеющихся в нём книгах: количество, стоимость, издательство. Программа реализует следующие операции: выдачу справок, продажи, поступление новых книг, а также поиск требуемых книг.

12.8. Из текста программы на языке Си удалить все комментарии. Обратить внимание на особые случаи со строками в кавычках и символьными константами.

12.9. Дано бинарное дерево, в записях которого имеются два указателя на строки, являющиеся именами файлов (файлы корневого узла не обрабатываются). Имя первого файла записи используется в качестве ключа бинарного дерева. В первом файле записаны строки знаков (длина не более двухсот знаков), оканчивающиеся точкой. Слова в строке разделены одним или несколькими пробелами. Длина слова не более восьми знаков. В файлах не более трёхсот разных слов.

В функции выполнить кодировку каждого файла: 0 — соответствует точке, 1 — наиболее часто встречаемому слову, 2 — слову с меньшей частотой встречаемости и т.д. по убыванию частоты встречаемости. Кодировка слов для всех файлов должна быть одинакова. Закодированный файл поместить в файл, задаваемый вторым именем в узлах данного дерева. Общий словарь файлов поместить в первый файл корневого узла, числовые коды данных слов — во второй файл этого же узла. Глобальные переменные и библиотечные функции (fscanf, sscanf) не использовать.

12.10. В командной строке заданы имена файлов. В записях файла в виде текста хранится либо непосредственно арифметическое выражение, либо арифметическое выражение, представленное в виде обратной польской записи. Идентификаторы — один символ.

Если строка есть обратная польская запись, то в записи файла хранится указатель на стек, в котором записаны арифметические значения идентификаторов (вещественные числа).

Если строка является обычным арифметическим выражением, то в записи хранятся:

а) указатель на функцию, используя которую данное выражение преобразуется в обратную польскую запись;

б) указатель на стек, в который записаны значения переменных.

Независимо от вида текста необходимо вычислить значения арифметических выражений и поместить их в те же записи и в тот же файл.

12.11. С клавиатуры задаются имена двух файлов, имя — не более двенадцати знаков. В первом файле записаны имена файлов, подлежащих обработке. Их не более пятидесяти, имя — не более семи знаков (без расширения). В файлах, подлежащих обработке, хранятся записи, содержащие строки знаков. Длина строки не более ста знаков, в файле не более трёхсот строк.

В первой функции рассортировать исходные файлы в порядке возрастания длин строк, в них содержащихся, и записать в новые файлы, имена которых получаются из старых имён путём удвоения последнего знака. Сортировку можно выполнить в памяти.

Во второй функции выполнить слияние рассортированных файлов в общий файл, имя этого файла задается с клавиатуры. Исходные файлы сохранить, в результирующий файл информация только записывается, промежуточные файлы (за исключением рассортированных файлов) не создавать. Глобальные переменные не использовать. При слиянии файлов (объединении) в памяти разрешается хранить не более чем по одной записи из каждого файла.

12.12. В командной строке задано имя файла, в котором записаны имена файлов, подлежащих обработке. Их не более восьми. В этих файлах записаны строки знаков, длина строки не более тридцати знаков. Строки упорядочены в порядке возрастания или убывания. Признак упорядочения («у» / «в») задан первым символом, записанным в файл.

Создать два результирующих упорядоченных файла. В первый файл поместить строки из файлов, упорядоченных по убыванию, во второй — строки из файлов, упорядоченных по возрастанию. Имена результирующих файлов в исходном файле (в командной строке) записаны в конце файла. Слияние файлов в результирующий файл выполнить в отдельных функциях, к которым обращаться по указателю. Глобальные переменные

не использовать и дополнительные файлы не создавать. В памяти можно хранить только по одной строке из каждого файла.

12.13. В файлах записаны последовательности целых чисел. Каждая последовательность начинается с числа, задающего длину последовательности. Признаком конца всей совокупности последовательностей данного файла является число нуль. В функцию передаются количество элементов в последовательности и указатель на нее. Если длина последовательности четная, то во второй функции (используя указатель на нее) определить максимальную по длине повторяющуюся ее часть, а если длина последовательности нечетная — то минимальную. Имена файлов задаются в командной строке. Вывести на экран имя файла и найденные максимальные и минимальные повторяющиеся части последовательности.

12.14. С клавиатуры задается «n» (n — не менее четырёх) имен файлов. Записи файла включают в себя фамилию, имя, отчество. Четвёртым элементом записи может быть:

а) либо строка знаков (не более двухсот знаков), заканчивающаяся точкой;

б) либо массив символов (не более десяти знаков), являющийся записью целого длинного числа со знаком. Последний символ массива «*»;

в) либо целое и вещественное число.

Другие элементы записи разрешается вводить самим. В первые три файла помещаются результаты обработки всех последующих файлов.

Обработка записей файла заключается в следующем. В записях, содержащих строки знаков, поменять местами длинное и короткое слова. Слова разделены одним или несколькими пробелами. Реализовать одну ситуацию: короткое слово встречается раньше длинного слова. В функции разрешается использовать переменную типа строки, символьную переменную и данные целого типа. Библиотечные функции для работы со строками не использовать. Записи с переставленными словами поместить в первый файл.

12.15. Имеется бинарный файл. Записью файла является строка, два первых байта которой задают длину строки (строка не более двухсот символов). Строка содержит алфавитные знаки и цифры. Сделать конкатенацию всех символьных слов, причем слова меньше четырёх байт удалить. Сложить все числа в строке и результат записать в конец полученной строки. Строку записать в тот же файл в том же формате (первые два байта — длина строки). Новые массивы и файлы не создавать.

12.16. Имеются два файла с описанием мужчин и женщин. В записях файлов содержатся данные о человеке (цвет глаз, рост, вес и другие данные) и требования, предъявляемые данным человеком к партнеру.

Подобрать для каждого лица несколько кандидатур так, чтобы они наиболее подходили друг к другу. При каждом вводе данных необходимо записывать их в файлы.

12.17. Имеется файл, в котором находятся: номер группы, ФИО студентов и название файла, в котором содержится перечень литературы. В программе реализовать: ввод, вывод списков литературы и студентов; списка студентов, получивших данную книгу. Обеспечить вывод ФИО студентов и литературы, которые просрочили срок сдачи литературы. Под список литературы использовать один файл. В записях студентов, получивших книгу, указывается код, соответствующий каждой книге (для уменьшения объёма хранимой информации).

12.18. В командной строке задано имя файла, в котором содержатся имена не более десяти других файлов. В этих файлах хранятся целые числа. Ввести числа в память, рассортировать их по возрастанию и затем записать их в файл. Имя файла для записи чисел используется то же, что дано в командной строке, но расширение заменить на DAT.

12.19. Имя текстового файла задано в командной строке. Файл содержит не более десяти строк, каждая строка не превышает восьмидесяти символов. В тексте встречаются числа, все они записаны в восьмеричной системе счисления. Преобразовать эти числа в десятичную систему счисления и записать отредактированный текст обратно в файл.

12.20. Дан массив указателей на стеки — их не более тридцати. Одним из элементов записей стека является указатель на имя файла. Элементами файла являются указатели на строки знаков (длиной не более пятидесяти). В первой функции найти первые по алфавиту двадцать строк, в которых младшая шестнадцатеричная цифра последнего знака — пять. Во второй функции рассортировать полученный массив строк в порядке возрастания кода первого знака строки. В третьей функции записать этот массив в файл, задаваемый в командной строке.

12.21. Дан массив указателей на кольца. Элементами записей кольца являются размеры матриц целых чисел и указатель на первый элемент матрицы. Матрица записана по столбцам. В каждой строке матрицы найти число, знаковое представление которого по алфавиту первое. Знаковое представление найденных чисел записать в файл, имя файла задается в командной строке. Созданный файл рассортировать на том же месте в алфавитном порядке записанных чисел-строк. В памяти место выделить только под две строки. Глобальные переменные, библиотечные функции, кроме функций работы с файлами, не использовать.

12.22. В командной строке даны имена двух символьных файлов. Первый файл содержит произвольный текст, слова в тексте разделены пробелами и знаками препинания. Второй файл содержит не более сорока пар слов, разделённых запятыми. Эти слова образуют пары: каждое первое слово считается заменяемым, а второе — заменяющим. Найти в первом файле все заменяемые слова и заменить их на соответствующие заменяющие слова. Результат поместить во второй файл.

12.23. Запись кольца содержит указатель на имя файла. В записи файла хранятся: фамилия, образование (среднее специальное, высшее), пол, семейное положение, а также девичья фамилия или для военнообязанного указывается военная специальность, а для невоеннообязанного — причина и год, с которого стал невоеннообязанным. Кроме того, хранится код (один байт), каждый бит которого определяет наличие соответствующей информации в записи файла. Записи файла рассортированы в порядке убывания фамилий.

В новый файл поместить в алфавитном порядке записи, относящиеся к лицам, которые изменили фамилию, имеют высшее образование и невоеннообязанные. Сортировки не использовать. Имя нового файла задается в командной строке. В кольце не более десяти элементов.

12.24. Дан массив указателей на бинарные деревья. Элементами дерева являются ключ — строка знаков и указатель на кольцо. Элементами кольца являются указатель на строку знаков и массив чисел с плавающей точкой. Как минимум четыре последних элемента не используются. Признак конца используемых чисел массива — число сто. Числа являются коэффициентами цепной дроби. Вычислить значение цепной дроби для каждого элемента кольца и для всех узлов бинарного дерева, в которые попадаем по правому указателю. Строку из кольца, коэффициенты цепной дроби и её значение записать блоками в файл, заданный в командной строке.

12.25. Дан указатель на массив указателей на имена бинарных файлов. Их количество задается в командной строке. Записи файла содержат или указатель на текст, или являются числами типа `int` либо `float`. Для каждого типа данных создать своё односвязное кольцо, элементы которого рассортированы в порядке возрастания чисел. В записи файла можно включать свои элементы. Отдельно сортировку кольца не выполнять.

12.26. В командной строке заданы имена бинарных файлов. В файле хранится текст — фамилия и массив чисел типа `double`. Для каждого файла создать свою очередь и из каждой очереди вывести на экран без повторения фамилии, которые в данной очереди встречаются многократно.

12.27. Дан стек. Элемент стека — указатель на бинарное дерево. Элементы дерева: указатель на текст, массив чисел целого типа и число, являющееся ключом. В командной строке заданы имена файлов. В один файл поместить информацию самой длинной, а во второй — самой короткой ветвей дерева. Файлы вывести на экран.

12.28. Дан указатель на массив указателей на бинарные деревья, последний элемент массива NULL. В записи дерева имеется или указатель на строку знаков, или указатель на числа типа float либо double. В командной строке задаются три файла. В каждый файл записать данные из бинарного дерева только одного типа. В записи дерева можно включать свои данные.

12.29. С клавиатуры задается имя файла, в котором хранится текст. В функции рассортировать в строках слова по величине их длины и записать в тот же файл на исходное место.

13. Прямой доступ к файлам

13.1. В файле содержится информация об авиарейсах: откуда и куда направляется, по каким дням, количество мест. Написать программу, которая бы:

- а) выдавала справочную информацию о нужном рейсе;
- б) добавляла новую информацию;
- в) резервировала билеты.

13.2. С клавиатуры вводятся два файла. В первом файле хранятся адреса и фамилии девушек, а во втором — тексты посланий. Сделать программу, чтобы любой выбранной девушке можно было послать любое послание — выдавался скоординированный текст. Должна быть возможность добавления, удаления и просмотра первого и второго файла.

13.3. С клавиатуры вводится файл. В нем содержится информация о гостинице, ФИО жильцов, количество номеров, площадь номера, класс номеров. Написать программу, которая

- а) выдаёт справки, кто и где проживает;
- б) ведёт учёт используемых номеров;
- в) обновляет информацию в файле.

13.4. Создать файл. В записи файла хранятся фамилия студента и список взятых им книг (не более двадцати). Студентов не более тридцати. Вывести список книг, взятых указанным студентом.

13.5. Справочная магазина автомобилей. Создать файл, в котором хранятся список всех типов автомобилей и имена двух других файлов: в одном — характеристики машин, в другом — список автостанций, где их обслуживают. Написать программу, которая выдаёт справки:

- а) есть ли в магазине машина требуемого типа;
- б) характеристику машины заданного типа;
- в) список автостанций, обслуживающих данный тип автомашин.

13.6. Создать файл, в котором хранится информация о видеофильмах (название, тип фильма, год выпуска). Программа реализует следующие операции:

- а) добавляет в файл новую информацию;
- б) исключает информацию о запрещённых фильмах;
- в) выводит список фильмов соответствующего типа.

13.7. Программа выдаёт справки по железнодорожному вокзалу:

- а) направление движения;
- б) дата, время отправления;
- в) тип билета, его стоимость;
- г) наличие свободных мест;
- д) наличие рейсов в определенный день.

13.8. Программа ведёт учёт наличия транспортных средств в автопарке и хранит их описание:

а) грузовая машина — в ремонте, списана, свободна, занята, её грузоподъёмность, стоимость одного часа эксплуатации;

б) автобус — количество мест, мягкий или жесткий, стоимость билета.

Реализовать выдачу справок.

13.9. Программа ведёт учёт лекарств в аптеке:

а) поступление новых лекарств;

б) выдача лекарств, если они имеются;

в) дата изготовления и контроль срока хранения лекарств;

г) вывод на экран и удаление лекарств, срок хранения которых уже истёк.

13.10. Создать файл с адресами районных отделов внутренних дел (РОВД) и для каждого РОВД указать второй файл. Во вторых файлах по каждому РОВД хранится информация о лицах, имевших правонарушения: фамилия, адрес, код правонарушения, его текстовая формулировка, применённое наказание. Реализовать выдачу справок и учёт информации.

13.11. В командной строке указано имя текстового файла. В файл записаны строки, размер строки не превышает восьмидесяти символов. Найти строку символов, не содержащую ни одного символа, входящего в другие строки.

13.12. Вычислить значения арифметических выражений, записанных в файл. Имя переменной — одна буква. Значения переменных в виде $A=3,1415$, $B=-37,2$ и т.д. записаны во второй файл. Имена файлов задаются в командной строке. В головном модуле вывести само выражение и его вычисленное значение.

13.13. В файле, задаваемом с клавиатуры, хранятся записи с вариантами. Элемент, определяющий вариантную часть, принимает значение «S» или «N»: «S» — определяет строку цифр (длина не более пяти знаков), «N» — определяет число (его значение не более 65550). Общей частью записи является строка знаков (её длина не более ста).

В функции преобразовать числа в строки знаков (системные функции не использовать). Записи, содержащие строки — числа, встречающиеся в файле более одного раза, без повторения записать в результирующий файл, задаваемый с клавиатуры. Исходный файл сохранить. Результирующих значений будет не более двухсот.

13.14. Создать файл следующего вида: первые 512 байт — область ключей, в ней неупорядоченно записаны ключи (числа типа `int`), далее в файле располагается область данных, содержащая записи. В записях хранятся ключ и описание устройства, соответствующего этому ключу. Информация в записи вводится с клавиатуры. Конец записи — символ 0.

13.15. В файле по указанному ключу найти требуемую запись. Файл имеет структуру, описанную в задании 13.14.

13.16. Замена записи по ключу в файле, созданном в задании 13.14:

- а) с уплотнением данных;
- б) без уплотнения области данных.

13.17. Подсчитать служебные слова языка Си в тексте программы (не учитывать слова, заключенные в кавычки). Имя файла с программой задается в командной строке.

13.18. Программа формирует таблицу перекрестных ссылок, т.е. выводит список идентификаторов, используемых в программе, и для каждого идентификатора выводит список номеров строк, в которых они встречаются.

13.19. С клавиатуры задается «n» квадратных матриц, содержащих числа с плавающей точкой. Записать элементы матрицы в файл, обходя ее по часовой стрелке или против часовой стрелки (по спирали), используя указатели. Имена файлов, вид обхода (0, 1), размеры матрицы (не более десяти) задаются в командной строке. Обращение к функции, обеспечивающей обход матрицы, выполнить по указателю на функцию. Глобальные переменные не использовать.

13.20. В командной строке задано имя файла со строками. Количество строк и размер строк заранее не известны. Прочитать строки из файла и сохранить в оперативной памяти.

13.21. В файле по формату «% d» записаны массивы чисел. В каждом массиве: первое число задает систему счисления, второе — длину массива, затем следуют элементы массива. Длина и элементы массива записаны в заданной системе. Таких групп данных может быть несколько. Не используя рекурсии, для всего файла создать бинарное дерево. В качестве ключа использовать элементы массива, но в десятичной системе счисления. Найти сумму ключей левой ветви дерева.

13.22. В командной строке заданы имена текстовых файлов. Первые два байта строки задают ее длину. Создать для каждого файла упорядоченное кольцо, в качестве признака упорядочения использовать первое слово строки: up — по алфавиту; dn — в обратном порядке.

13.23. В двоичном файле указан размер квадратной матрицы, а затем построчно записаны элементы матрицы целых чисел. Таких групп данных в файле несколько. Вычислить сумму элементов, расположенных ниже главной диагонали. Результат записать в конец файла. Имя файла задается в командной строке.

13.24. В файле, имя которого задано в командной строке, хранится текст. Длина строки в тексте не превышает восьмидесяти символов. Если в тексте встречаются числа, то их необходимо вывести на экран, а файл уплотнить.

13.25. В текстовом файле записаны строки. В каждой строке определить длинное и короткое слово и вывести их на экран.

13.26. В файлах записаны последовательности целых чисел. Каждая последовательность начинается с числа, задающего ее длину. Признаком конца всей совокупности последовательностей является число 0.

В функцию передаются количество элементов в последовательности и указатель на нее. Если длина последовательности — число четное, то во

второй функции (используя указатель на неё) определить максимальную по длине повторяющуюся ее часть; если — нечетное, то — минимальную. Имена файлов задаются в командной строке. Вывести на экран имя файла и найденные минимальную и максимальную повторяющиеся части последовательности.

13.27. Дано бинарное дерево. В качестве ключа используется строка знаков (не более десяти символов). В записях дерева хранятся указатели на кольца, в записях кольца есть указатели на строки знаков. Слова в строке разделены произвольным количеством пробелов. Определить частоту встречаемости слов во всех строках кольца. Вывести в файл слова и соответствующую им частоту встречаемости. Имя файла для каждого кольца задается с клавиатуры. Глобальные переменные и системные функции (кроме функций ввода/вывода и работы с файлами) не использовать. Исходные строки сохранить.

13.28. В программе обработать бинарное дерево. В узлах дерева содержатся имя файла и дата его создания. Написать программу, удаляющую из дерева все записи с именами файлов, которые были созданы до даты, заданной с клавиатуры.

13.29. В командной строке заданы имя файла и последовательность знаков (до десяти байт). Найти в файле эту последовательность знаков и заменить её словом «text».

13.30. Дан массив указателей на кольца. Элементами записи кольца являются размеры матрицы целых чисел и указатель на первый элемент матрицы. Матрица записана по столбцам.

В каждой строке матрицы найти число, символьное представление которого минимально. Символьное представление найденных чисел записать в файл. Имя файла задаётся в командной строке. Строки в файле рассортировать в алфавитном порядке. В памяти определить место только для двух строк. Глобальные переменные, библиотечные функции, кроме функций работы с файлами, не использовать.

13.31. Дан массив указателей на стеки — их не более тридцати. Одним из элементов записи стека является указатель на имя файла. Элементами файла являются указатели на строки знаков. В одной функции найти первые по алфавиту двадцать пять строк, в которых младшая шестнадцатеричная цифра последнего знака — пять. Во второй функции рассортировать полученный массив строк в порядке возрастания кода первого знака строки. В третьей функции записать этот массив в файл, задаваемый в командной строке.

13.32. Дано кольцо с двумя связями. В записи кольца имеется указатель на бинарное дерево. Элементы бинарного дерева: указатель на целое или длинное вещественное число, записанное в виде символьной строки, а также целое число — ключ дерева. В командной строке задан файл, в который записаны имена двух файлов.

В один бинарный файл поместить числовое значение длинных вещественных чисел и ключ узла с этим числом, во второй — целые числа и ключ, соответствующий этому узлу. Рекурсии не использовать. Реализовать свою функцию перевода строки в число.

13.33. Дан массив указателей на кольца, их не более пятидесяти, последний элемент — указатель NULL. Один элемент кольца — указатель на число, задающее систему счисления (от 2 до 36). Другой элемент — массив указателей на числа в десятичной системе счисления. Чисел не более ста, последнее число ноль.

В одной функции реализовать обход кольца, во второй — перевод чисел в указанную систему счисления. Результат перевода разместить в оперативной памяти, указатель на этот массив записать в кольцо, а изображение числа в новой системе поместить в файл: в один — результаты перевода чисел в систему счисления с основанием три, во второй — с основанием семь, в третий — с основанием шестнадцать. Другие результаты в файлы не записывать.

13.34. В командной строке задано имя файла. В файл записаны строки. Длина строки не превышает восьмидесяти символов. Если в строке встречаются числа, то эти числа вывести на экран.

13.35. Дан указатель на массив указателей на имена бинарных файлов. Их количество задается в командной строке. Записи файла содержат указатель или на текст, или на число типа `int` либо типа `float`. Для каждого типа данных создать свою очередь с указателями на соответствующие узлы бинарного дерева. Данные из каждой очереди поместить в отдельный файл. Имена файлов задаются в командной строке. В записи дерева можно включать свои элементы. Отдельно сортировку кольца не выполнять.

13.36. В командной строке заданы имена файлов. Элементами записи файла являются:

- а) число целого типа, записанное в виде строки знаков;
- б) указатель на массив чисел, являющихся номерами функций, которые необходимо выполнить. Признак конца массива — число ноль. Выполнить соответствующие функции, используя в качестве аргумента целое число.

13.37. Элементом узла бинарного дерева является массив указателей на функции, которые надо выполнить. Последний указатель массива — NULL. Функции в качестве параметра принимают и возвращают результат в виде числа удвоенной точности. Необходимо выполнить соответствующие функции. Результат выполнения функции, её аргумент и номер функции поместить в узел кольца, подсоединяемого к соответствующему узлу бинарного дерева. Функцию перевода строки в число написать свою.

13.38. Дан массив указателей на структуры, последний элемент массива — указатель NULL. Элемент структуры — указатель на стек и знак «U», «D» — признак сортировки стека. Элемент стека — число типа double. На месте исходных стеков получить один стек, рассортированный в порядке возрастания элементов, второй — в порядке убывания. Сортировки не использовать.

Библиотека БГУИР

Учебное издание

Демидович Евгений Михайлович

Сборник заданий

по курсу

**ОСНОВЫ АЛГОРИТМИЗАЦИИ
И ПРОГРАММИРОВАНИЯ**

для студентов специальности
«Вычислительные машины, системы и сети»
дневной и вечерней форм обучения

Редактор Н.А. Бебель
Корректор Е.Н. Батурчик

Подписано в печать 17.01.2003.

Бумага офсетная.

Усл. печ. л. 3,14.

Печать ризографическая.

Уч.-изд. л. 2,9.

Тираж 100 экз.

Формат 60x84 1/16.

Гарнитура «Ариал».

Заказ 682.

Издатель и полиграфическое исполнение:

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Лицензия ЛП № 156 от 30.12.2002.

Лицензия ЛВ № 509 от 03.08.2001.

220013, г. Минск, П. Бровки, 6