

**Полякова Ирина Николаевна**

Сборник заданий по Си для практических занятий в компьютерном классе.  
Методическое пособие для студентов бакалавриата по направлению  
“Прикладная математика и информатика”, 2010.

В данном пособии приведены темы и примеры задач к ним, которые предлагаются на практических занятиях в компьютерном классе студентам-бакалаврам 2 курса по направлению “Прикладная математика и информатика” на семинарах по курсу “Операционные системы” при изучении раздела “Языки системного программирования. Язык программирования СИ”.

## Работа с числовыми данными

1. Подсчитать количество цифр в десятичной записи целого неотрицательного числа  $n$ .
2. Определить, верно ли, что куб суммы цифр натурального числа  $n$  равен  $n^2$ .
3. Определить, является ли натуральное число  $n$  степенью числа 3.
4. Для данного вещественного числа  $a$  среди чисел  $1, 1 + (1/2), 1 + (1/2) + (1/3), \dots$  найти первое, большее  $a$ .
5. Для данного вещественного положительного числа  $a$  найти наименьшее целое положительное  $n$  такое, что  $1 + 1/2 + 1/3 + \dots + 1/n > a$ .
6. Дано натуральное число  $n$ . Найти значение числа, полученного следующим образом: из записи числа  $n$  выбросить цифры 0 и 5, оставив прежним порядок остальных цифр.
7. Дано натуральное число  $n$ . Получить все такие натуральные  $q$ , что  $n$  делится на  $q^2$  и не делится на  $q^3$ .
8. Распечатать первые  $n$  чисел Фибоначчи ( $f_0 = 1; f_1 = 1; f_{k+1} = f_{k-1} + f_k; k = 1, 2, 3, \dots$ )
9. Распечатать числа Фибоначчи (см. задачу 8), являющиеся простыми числами со значениями меньше  $n$ .
- 10.. Целой переменной  $k$  присвоить значение, равное третьей от конца цифре в записи целого положительного числа  $x$ .
- 11.. Целой переменной  $k$  присвоить значение, равное сумме цифр в записи целого положительного трехзначного числа  $x$ .
- 12.. Целой переменной  $k$  присвоить значение, равное первой цифре дробной части в записи вещественного положительного числа  $x$ .
- 13.. Определить число, полученное выписыванием в обратном порядке цифр заданного целого трехзначного числа.
- 14.. Идет  $n$ -ая секунда суток. Определить, сколько полных часов и полных минут прошло к этому моменту.

## Работа с символьными данными

1. Заменить во вводимой последовательности все малые буквы на большие, отстоящие от конца алфавита настолько, насколько исходная буква отстоит от начала.
2. Напечатать каждую букву столько раз, насколько она отстоит от начала алфавита, из каждой группы '+' оставить только один '+'.  
"abc123+++&" -> "abbccc123+&"
3. Пусть во входном потоке находится последовательность литер. Вывести в выходной поток последовательность литер, измененную следующим образом: оставить только те группы цифр, которые составлены из подряд идущих цифр с возрастающими значениями; все остальные цифры и группы цифр удалить (другие символы копировать в выходной поток без изменения).
4. Пусть во входном потоке находится последовательность литер. Вывести в выходной поток последовательность литер, измененную следующим образом: удалить из каждой группы цифр, которой не предшествует точка, все начальные нули (кроме последнего, если за ним идет точка либо в этой группе нет других цифр, кроме нулей ; например, a-000123+bc+0000.0008-0000+0001.07 преобразуется в a-123+bc+0.0008-+1.07).
5. Пусть во входном потоке находится последовательность литер, заканчивающаяся точкой (кодировка ASCII). Вывести в выходной поток последовательность литер, измененную следующим образом:
  - a) заменить все символы '?' на '!';
  - b) удалить все символы '-' и удвоить все символы '&';
  - c) удалить все символы, не являющиеся строчными латинскими буква-ми;
  - d) заменить все прописные латинские буквы строчными (другие символы копировать в выходной поток без изменения);
  - e) заменить все строчные латинские буквы прописными (другие символы копировать в выходной поток без изменения);
  - f) каждую группу рядом стоящих символов '+' заменить одним таким символом;
  - g) каждую группу из n рядом стоящих символов '\*' заменить группой из n/2 рядом стоящих символов '+' ( n >= 2 ); одиночные '\*' копировать в выходной поток без изменения;
  - h) удалить из каждой группы подряд идущих цифр все начальные незначащие нули (если группа состоит только из нулей, то заменить эту группу одним нулем);
  - i) удалить все комбинации символов the;
  - j) оставить только те группы цифр, которые составлены из подряд идущих цифр с возрастающими значениями; все остальные цифры и группы цифр удалить ( другие символы копировать в выходной поток без изменения);

к) удалить группы символов, расположенные между фигурными скобками { и }. Скобки тоже должны быть удалены. Предполагается, что скобки сбалансированы, и внутри каждой пары скобок других фигурных скобок нет.

6. Реализовать интерпретатор упрощенной команды like языка SQL. На вход системы итеративно подаются анализируемые строки и шаблоны в следующем виде:

*'строка' like 'шаблон'*

Необходимо вывести ответ YES или NO, в зависимости от того, сопоставился шаблон с входной строкой или нет. Шаблон может содержать следующие команды:

% - допускает совпадение с любым количеством любых символов

\_ - допускает совпадение с любым одиночным символом

[c1-c2] - допускает совпадение с любым одиночным символом из диапазона c1-c2

[c1c2c3...cN] - допускает совпадение с любым одиночным символом из

последовательности c1,c2,c3,...,cN

[^c1-c2] - допускает совпадение с любым одиночным символом не из диапазона c1-c2

[^c1c2c3...cN] - допускает совпадение с любым одиночным символом не из

последовательности c1,c2,c3,...,cN

### Пример

исходные данные	Результат
'abcde' like 'a'	NO
'abcde' like 'a%'	YES
'abcde' like '%a'	NO
'abcde' like 'b'	NO
'abcde' like 'b%'	NO
'abcde' like '%b'	NO
'25%' like '_5[%]'	YES
'_52' like '[_]5%'	YES
'ab' like 'a[a-cdf]'	YES
'ad' like 'a[a-cdf]'	YES
'ab' like 'a[-acdf]'	NO
'a-' like 'a[-acdf]'	YES
'[]' like ' [[]]'	YES
'''' like '_'''	YES
'U' like '[^a-zA-Z0-9]'	NO

## Указатели. Массивы. Динамическая память

1. Написать функцию от трёх параметров, определяющую, чередуются ли положительные и отрицательные элементы в целочисленном массиве  $a$  из  $n$  элементов и вычисляющую целочисленное значение  $p$ . Если элементы чередуются, то  $p$  - это сумма положительных элементов, иначе  $p$  - это произведение отрицательных элементов. Размерность массива и его элементы должны быть считаны в функции `main()` со стандартного ввода и переданы в функцию как параметры. Возврат результата  $p$  также должен быть осуществлен через параметр функции.
2. Написать функцию от трёх параметров, определяющую, упорядочены ли строго по возрастанию элементы в целочисленном массиве  $a$  из  $n$  элементов, и вычисляющую целочисленное значение  $p$ . Если элементы упорядочены, то  $p$  - это произведение разностей рядом стоящих элементов, иначе  $p$  - это количество нарушений порядка в массиве  $a$ . Размерность массива и его элементы должны быть считаны в функции `main()` со стандартного ввода и переданы в функцию как параметры. Возврат результата  $p$  также должен быть осуществлен через параметр функции.
3. Написать программу, обнуляющую каждую четную двоичную единицу в коде, размещенном в переменной типа `int`. Вывести исходные данные и полученный результат в виде, удобном для анализа проведенных преобразований.
4. Написать программу, обнуляющую каждую нечетную двоичную единицу в коде, размещенном в переменной типа `int`. Вывести исходные данные и полученный результат в виде, удобном для анализа проведенных преобразований.
5. Описать функцию, вычисляющую значение  $x_0 + x_0*x_1 + x_0*x_1*x_2 + \dots + x_0*x_1*x_2*\dots*x_m$ , где  $x_i$  - элементы вещественного массива  $x$  из  $n$  элементов,  $m$  - индекс первого отрицательного элемента этого массива либо число  $n-1$ , если такого элемента в массиве нет. Продемонстрировать работу этой функции.
6. Описать функцию, вычисляющую значение  $x_0*y_0+x_1*y_1+ \dots + x_k*y_k$ , где  $x_i$  - отрицательные элементы вещественного массива  $a$  из  $n$  элементов, взятые в порядке их следования;  $y_i$  - положительные элементы этого массива, взятые в обратном порядке;  $k = \min(p,q)$ , где  $p$  - количество положительных элементов массива  $a$ ,  $q$  - количество отрицательных элементов этого массива. Продемонстрировать работу этой функции.
7. Описать функцию, вычисляющую значение  $\max(x_0 + x_{n-1}, x_1 + x_{n-2}, x_2 + x_{n-3}, \dots, x_{(n-1)/2} + x_{n/2})$ , где  $x_i$  - элементы вещественного массива  $x$  из  $n$  элементов. Продемонстрировать работу этой функции.

8. Описать функцию, вычисляющую значение  $\min(x_0 * x_1, x_1 * x_2, x_2 * x_3, \dots, x_{n-3} * x_{n-2}, x_{n-2} * x_{n-1})$ , где  $x_i$  - элементы вещественного массива  $x$  из  $n$  элементов. Продемонстрировать работу этой функции. (2балла)

9. На вход подается массив вещественных чисел (неизвестной длины). Отсортировать массив «пузырьком». Подсчитать количество операций, выполненных при сортировке (сравнений, присваиваний).

10. На стандартный вход (в main) подается массив целых чисел произвольной (заранее неизвестной) длины. Описать функцию, вычисляющую значение  $x_0*y_0+x_1*y_1+ \dots + x_k*y_k$ , где  $x_i$  – отрицательные элементы массива  $a$  из  $n$  элементов, взятые в порядке их следования;  $y_i$  – положительные элементы этого массива, взятые в обратном порядке;  $k = \min(p,q)$ , где  $p$  – количество положительных элементов массива  $a$ ,  $q$  – количество отрицательных элементов этого массива. Продемонстрировать работу этой функции.

11. Программа. В файле1 (имя файла задается в командной строке) - непустая последовательность слов, разделенных пробелами. Ограничения на длину слова и всей последовательности вводить не разрешается. Распечатать в стандартный выходной поток заданную последовательность, если в ней не совпадают вторая и предпоследняя буквы.

12. Программа. В файле1 (имя файла задается в командной строке) - непустая последовательность слов, разделенных пробелами. Ограничения на длину слова и всей последовательности вводить не разрешается. Распечатать в стандартный выходной поток все слова, у которых не совпадают первая и последняя буквы.

13. Дан граф, реализовать поиск кратчайшего пути в графе из данной вершины  $A$  в вершину  $B$ . Топология графа задается во входном файле в произвольном формате. Выбор вершин  $A, B$  осуществляется с клавиатуры в диалоговом режиме. В результате работы программа должна распечатать на экране маршрут и его длину. Предусмотреть удобный пользовательский интерфейс. Имя входного файла задается с клавиатуры. Программа должна корректно обрабатывать несвязные графы. Вершины графа помечены строковыми идентификаторами. Граф ориентированный, дуги единичной длины

## Структуры

Пусть результаты анализа некоторого текста, состоящего из английских слов, содержатся в следующем частотном словаре dictionary:

```
#define MAXSIZE 1000
#define LENGHT 20 /* максимальная длина слова */
struct elem { char word [20]; struct info data; };
struct info { int count; /* количество повторений слова в данном тексте */
char alias [20]; /* синоним данного слова */ };
struct { struct elem tabl [ MAXSIZE];
int number; /* количество слов в словаре */ } dictionary;
```

Каждое слово (word) встречается в словаре только один раз; синонимы (alias) могут быть одинаковыми у разных слов.

Написать программу, которая:

- создаёт словарь заданного вида (определяется вариантом);
- выводит созданный словарь на экран;
- производит выборку и/или изменения элемента (определяется вариантом);
- выводит результат и/или измененный словарь на экран.

Наполнение словаря (не менее 10 слов), вывод словаря на экран, выборку и/или изменения элемента – оформить в виде отдельных функций. Обеспечить возможность работы программы в интерактивном режиме (словарь вводится пользователем с клавиатуры).

Пример варианта:

Описать функцию, определяющую, встречается ли данное слово в этом словаре.

Результат – номер элемента массива либо 0. Слова неупорядочены по алфавиту.

## Работа с файлами

1. В файле записана непустая последовательность целых чисел. Имя файла задается в командной строке. Определить, сколько четных чисел содержится в файле.
2. Написать программу, определяющую, сколько строк, состоящих из одного, двух, трех и т.д. символов, содержится в данном файле. Считать, что длина каждой строки - не более 80 символов. Имя файла задается в командной строке.
- 3..Программа. В файле записана непустая последовательность целых чисел. Имя файла задается в командной строке. Изменить файл следующим образом: поменять местами максимальный элемент последовательности и первый по счёту элемент последовательности. Использовать дополнительные временные файлы запрещается.
4. Программа. В файле записана непустая последовательность целых чисел. Имя файла задается в командной строке. Определить, составляют ли эти числа арифметическую прогрессию.
- 5 ..Программа. В файле записана непустая последовательность целых чисел. Имя файла задается в командной строке. Изменить файл следующим образом: поменять местами минимальный и максимальный элементы последовательности. Использовать дополнительные временные файлы запрещается.
6. Программа. В файле записана непустая последовательность целых чисел. Имя файла задается в командной строке. Определить, образуют ли эти числа возрастающую последовательность.
7. Написать программу, определяющую, какой символ чаще других встречается в данном файле. Имя файла задается в командной строке.
8. Напишите программу, которая выдает на экран  $n$  строк файла, начиная со строки номер  $m$  (нумерация строк с единицы). Имя файла,  $n$  и  $m$  задаются в командной строке. Предусмотреть обработку возможных ошибок.
9. Написать программу, которая удаляет из файла все строки длиной более десяти символов. Длина строк не ограничена. Имя файла задается в командной строке.
10. Написать программу, центрирующую строки файла относительно середины экрана, т.е. добавляющую в начало строки такое количество пробелов, чтобы середина строки печаталась в 40-ой позиции (считаем, что обычный экран имеет ширину 80 символов и длина строк в файле не больше 80). Имя файла задаётся в командной строке.



11. Написать программу, которая изменяет заданный файл следующим образом : усекает все строки до 10 символов. Длина строк не ограничена. Имя файла задается в командной строке.

12. Программа. Даны два файла, строки в которых упорядочены по алфавиту. Написать программу, осуществляющую слияние этих двух файлов в третий, строки которого тоже упорядочены по алфавиту. Имена всех трех файлов задаются в командной строке.

13. Программа. Дан файл и две строки. Все вхождения первой строки в файл заменить второй строкой (вхождения первой строки в качестве подстроки не рассматривать). Имя файла и строки задаются в командной строке.

14.. Программа. Дан файл и две строки. Все вхождения первой строки в файл (в том числе и в качестве подстроки) заменить второй строкой. Имя файла и строки задаются в командной строке.