

3.2. Кортежи элементов строк (подстроки) и действия с ними

1. Заданная строка из 60 знаков содержит утверждение Фортран-программы. Выделить содержимое полей метки, предложения и обозначения.

2. Проверить, содержится ли в заданной строке подстрока ЦИИ.

3. Определить номер позиции, начиная с которой в заданную строку входит в качестве подстроки слово ПРОГРАММА.

4. Задано две строки знаков: одна длиной 60, а другая - не более 60. Определить количество вхождений второй строки в первую в качестве подстроки.

5. В заданной строке, содержащей синтаксически корректное утверждение на C++, в поле предложения содержатся пробелы. Исходя из конкретного вида утверждения выполнить одно из следующих действий:

1) в явном объявлении размерности только одного массива выделить имя массива и определить его размерность;

2) в явном объявлении двух массивов выделить имена массивов и определить, являются ли они одномерными;

3) в условном логическом операторе выделить текст, задающий условие, и определить его длину в знаках.

6. Задана строка, содержащая оператор передачи информации Фортрана, использующий логический номер устройства. Определить, является заданный оператор оператором ввода или оператором вывода, и в соответствии с полученным результатом занести в поле обозначения слово ВВОД или слово ВЫВОД. Предполагается, что пробелы в исходной строке использованы как до, так и после оператора, а сам оператор записан без пробелов. Длина строки - 60 знаков.

3.3. Редактирование строк

1. Удалить пробелы в заданной строке.

2. Удалить в строке все знаки, входящие в заданный набор.

3. Заменить заданную подстроку в исходной строке, содержащей не более 60 знаков, на другую подстроку. Заменяемая и заменяющая подстроки задаются строками, которые могут иметь разную длину.

4. Задана строка из текста программы на Фортране, содержащая

60 знаков. Сдвинуть на заданное число позиций вправо часть строки, начиная с указанной позиции. Освободившееся поле заполнить пробелами. Содержимое поля обозначения и длина строки должны остаться неизменными.

5. Строка содержит синтаксически корректное утверждение на C++. В поле предложения пробелы используются только для отделения ключевых слов. Обеспечить замену отдельных фрагментов утверждения на заданные:

1) в операторе форматного ввода заменить номер устройства и метку формата;

2) в операторе цикла заменить фрагменты, задающие начальное и конечное значения переменной цикла, а шаг изменения сохранить из исходной строки;

3) в операторе безусловного перехода заменить метку;

4) в объявлении типа заменить ключевое слово, определяющее тип.

6. Заменить в строке цифры 0, 1, 2, ..., 9 на слова, соответственно, НОЛЬ, ОДИН, ДВА, ..., ДЕВЯТЬ.

3.4. Обработка текста, представленного массивом строк

1. Разместить содержимое строки в нескольких строках меньшей длины, сохранив порядок следования знаков и полностью заполнив все новые строки (кроме, может быть, последней).

2. В тексте из 30 строк по 66 знаков подсчитать количество строк, начинающихся с одного или нескольких пробелов, за которыми следует знак "-" (дефис).

В задачах 3 - II предполагается, что задан текст, представленный массивом строк. Каждая строка содержит не более заданного количества знаков. Текст состоит из предложений, каждое из которых может либо находиться полностью в одной строке, либо начинаться в одной строке и продолжаться в последующих строках. Каждое предложение заканчивается одним из знаков препинания: точкой, восклицательным или вопросительным знаком. Некоторые предложения могут размещаться в строке с отступом на пять позиций, т.е. такая строка содержит пробелы в первых пяти позициях.

3. Определить количество предложений в тексте, если каждое предложение заканчивается точкой.

4. Определить количество предложений, начинающихся с отступа.

5. Подсчитать количество предложений в тексте, если каждое из них может быть закончено любым из допустимых знаков препинания.

6. В каждой строке текста содержится не более трех предложений, причем ни одно предложение не имеет переносов на другую строку. Сформировать новый массив с текстом, в котором каждое предложение было бы размещено в отдельной строке с отступом на пять пробелов.

7. В каждой строке текста содержится не более одного законченного предложения. Определить количество предложений в тексте и для каждого предложения - его длину (в знаках).

8. При записи текста использованы отступы. Определить количество предложений, начинающихся с отступа и длины (в строках) текстов между двумя соседними отступами.

9. Для заданного текста сформировать массивы, каждый из которых содержал бы только те предложения, которые в исходном тексте заканчиваются одним из возможных знаков препинания. В массивах предложения должны быть расположены в порядке их следования в исходном тексте и каждое предложение должно начинаться с:

1) порядкового номера предложения в массиве, законченного точкой;

2) порядкового номера предложения в исходном тексте, законченного закрывающей круглой скобкой.

10. Некоторые предложения в тексте могут начинаться с отступов. Составить перечень номеров строк исходного текста, содержащих такие предложения.

11. Определить количество и составить перечень номеров строк исходного текста, начинающихся с цифры и непосредственно следующей за ней закрывающей круглой скобки.

В последующих задачах предполагается, что задан текст синтаксически корректной программы на языке C++. Текст представлен массивом строк.

12. Из текста программы длиной 40 строк сформировать новый текст, не содержащий комментариев. Определить длину нового текста (в строках) и разместить его в новом массиве, сохранив порядок следования в исходном массиве размещаемых утверждений. Для каждого такого утверждения поместить в поле обозначений его порядковый номер в новом массиве.

13. Преобразовать текст программы, находящийся в массиве строк, так, чтобы он не содержал комментариев. Определить длину нового текста (в строках).

14. В тексте программы, не содержащей комментариев, определить количество утверждений, не имеющих строк-продолжений.

15. В тексте программы без комментариев одно утверждение имеет строки-продолжения. Подсчитать для этого утверждения, сколько строк оно занимает.

16. Текст программы не содержит комментариев. Определить количество утверждений, имеющих строки-продолжения, и для каждого такого утверждения подсчитать, сколько строк оно занимает.

17. Для заданной программы определить количество комментариев, количество утверждений и для каждого утверждения указать, сколько строк оно занимает.

3.5. Обработка текста, разделяемого на слова

1. Выделить в строке отдельные слова, разделенные пробелами, определить длину каждого слова, подсчитать общее количество слов.

2. Из строки, содержащей несколько слов, разделенных не менее чем одним пробелом, сформировать новую строку, в которой те же слова разделены только одним пробелом.

3. Задана строка, в которой пробелы разделяют слова. Разместить содержимое строки в нескольких строках меньшей длины, стремясь полностью заполнить эти строки (кроме, возможно, последней), разделяя слова только одним пробелом и не допуская переноса слов.

4. Задан массив строк длиной не более 30 знаков каждая.

Строки не содержат пробелов. Сформировать массив строк, длина ко-

торых определяется значением дополнительно задаваемого параметра. (Задание длины формируемых строк согласовать с максимальной длиной строки на устройстве вывода.) Каждая строка в результирующем массиве должна содержать максимально возможное количество очередных строк исходного массива, разделенных по крайней мере одним пробелом. Если результирующая строка получается короче заданной длины, в нее следует по возможности равномерно вставить дополнительные пробелы между составляющими ее исходными строками.

5. Задан текст, содержащий произвольное количество строк, в которых отдельные слова могут разделяться одним или несколькими пробелами и знаками пунктуации (переносы слов с одной строки на другую не используются). Сформировать новый текст, который является результатом одного из следующих преобразований исходного текста:

1) заменить заданное слово на другое слово столько раз, сколько оно встречается в тексте;

2) каждое слово, встречающееся более одного раза в тексте, заменить начиная со второго вхождения на обозначение вида: $S(NN)$, где S - первый знак слова; NN - порядковый номер в тексте первого вхождения слова.

6. Преобразовать исходный текст таким образом, чтобы каждое его слово целиком находилось в одной и той же строке (т.е. ликвидировать переносы, которые обозначаются в тексте знаком "-" в конце строки).

7. Задан произвольный текст. Слова в тексте могут разделяться одним или несколькими пробелами, знаками пунктуации и переходами на следующую строку. Каждое слово размещается в одной и той же строке (т.е. переносы слов не используются). Выполнить следующие действия:

1) выделить каждое отдельное слово, входящее в текст;

2) для каждого такого слова определить количество вхождений в исходном тексте;

3) упорядочить слова в порядке убывания количества вхождений;

4) распечатать упорядоченные слова в виде таблицы, содержащей три колонки: номера по порядку, обнаруженные слова, количества вхождений.

8. Произвольный заданный текст вывести в виде абзаца с заданной длиной строки. При этом обеспечить:

- отступ от левого края в первой строке абзаца на заданное число позиций;

- размещение каждого слова целиком в одной и той же строке (ликвидировать переносы, которые обозначены в тексте знаком "-" в конце строки);

- выравнивание правого края абзаца за счет добавления или удаления пробелов между словами.

В конец текущей строки можно добавлять слова из начала следующей строки или, наоборот, непоместившиеся слова в конце строки переносить в начало следующей. Последнюю строку абзаца выравнивать по правому краю не требуется.