

```
1 Program c4_v3_12;  
2 {
```

Задача.

Пользователю хостинговой компании доступны не более 13 доменов, у каждого домена – уникальное имя. Сервер хостинговой компании регистрирует обращение к каждому домену, сохраняя имя домена и количество переданных ему пакетов. Вам предлагается написать эффективную, в том числе по используемой памяти, программу, которая будет составлять отчет по пришедшим на сервер запросам и определять количество доменов на которые было переслано минимальное количество пакетов за истекший период. Следует учитывать, что количество запросов в списке может быть очень велико. Перед тестом программы кратко опишите используемый вам и алгоритм решения задачи.

На вход программе в первой строке подается количество пришедших запросов N. В каждой из последующих N строк записаны имя домена и количество переданных ему пакетов, разделенные двоеточием.

Длина строки не превосходит 100 символов, имя домена может содержать только буквы, цифры и символ "." (точка).

Пример входных данных:

```
5  
Mydomen.homedom.ru:480  
123.11.12:1200  
123.ru:560  
Mydomen.homedom.ru:80  
123.11.12:140
```

Программа должна вывести имена всех доменов, встречающихся в списке, в порядке возрастания (не убывания) числа переданных на них пакетов, с указанием общего количества полученных доменом пакетов и количества доменов с минимальным количеством полученных пакетов. При этом каждый домен должен быть выведен только один раз.

Пример выходных данных для приведенного выше примера входных данных:

```
Mydomen.homedom.ru 560  
123.ru 560  
123.11.12 1340  
2
```

Решение.

Степан Рамбовский,
11 "А", SGLS,
24.03.12, 16:27

```

}

45 {
    Compiler: Borland Pascal v.7.0

    КРАТКИЙ АЛГОРИТМ
    Занесем входные данные в массив, состоящий из записей.
    Заметим, что максимальное количество доменов не более 13,
    поэтому создадим одномерный массив состоящий из 13 элементов.
    В каждой записи данного массива будем хранить имя домена и
    количество обращений к нему. Динамически будем вносить в массив
    поступающие данные, учитывая возможность появления нового домена
    и увеличения количества обращений к уже существующим доменам.

    Отсортируем массив с записями по количеству запросов по
    возрастанию.

    Выведем каждую запись массива на экран с новой строки.

    Определим количество записей имеющих наименьшее количество
    обращений и выведем его с новой строки.
}

61 Uses
62     Crt;

63 Type
64     zap = record
65         dom_name : String [99]; { имя домена }
66         hosts : Longint; { количество обращений к домену }
67     end;

68     om = array [1..13] of zap;

69 Var
70     i, j, { индексы }
71     t, { количество различных доменов }
72     f { флаг нового домена }
73     : Byte;

74     N, { общее количество обращений }
75     host { текущее количество обращений }
76     : Longint;

77     s : Char; { символ для текущей обработки }

78     A : om; { массив с записями }

79     buf : zap; { темповая }

80     name : String [98]; { для формирования текущего имени-домена }

```

```

81     Begin
82         ClrScr;

83         for i := 1 to 13 do
84             begin
85                 A [i].dom_name := '';
86                 A [i].hosts := 0;
87             end;

88         ReadLn (N);

89         t := 0;
90         repeat
91             name := '';
92             repeat
93                 Read (s);
94                 name := name + s;
95             until s = ':';
96             Delete (name, Length (name), 1);

97             ReadLn (host);

98             f := 0;
99             for j := 1 to t do
100                 if name = A [j].dom_name
101                     then
102                         begin
103                             A [j].hosts := A [j].hosts + host;
104                             f := 1;

105                             Break;
106                         end;

107                 if f = 0
108                     then
109                         begin
110                             Inc (t);
111                             A [t].dom_name := name;
112                             A [t].hosts := host;
113                         end;

114                 Dec (N);
115             until N = 0;

116         for i := 1 to t - 1 do
117             for j := i + 1 to t do
118                 if A [j].hosts < A [i].hosts
119                     then
120                         begin

```

```

121         buf := A [i];
122         A [i] := A [j];
123         A [j] := buf;
124     end;

125     WriteLn;

126     for i := 1 to t do
127         WriteLn (A [i].dom_name , ' ', A [i].hosts);

128         for i := 1 to t do
129             if A [i].hosts <> A [i + 1].hosts
130                 then
131                     Break;
132             WriteLn (i);

133         ReadLn;
134     End.

```

Listing данной задачи опубликован в сети Internet по адресу
<http://www.Best-Listing.ru/color-9-task-477.html>

Sergey Mitrofanov, 22.04.14, 12:02

E-mail: infostar@mail.ru

© <http://www.Best-Listing.ru/>, 2006–2014