

```

1      /*
      TITLE: Удалить все повторяющиеся слова

      File: g5_str.cpp

      Задача. Исключить из данного текста все слова, которые в нем
      встречаются более одного раза.

      Решение. Сергей Митрофанов,
              Номе,
              16.10.08, 20:38;
              17.10.08, 20:36.

      History. Это самый первый листинг, который я пишу на
      системе LIST 5.0, в которой только сегодня появился
      модуль серого блока, его создания, удаления и
      восстановления.

      */

15     # include <stdio.h>
16     # include <conio.h>
17     # include <string.h>

18     int main ()
19     {
20         clrscr ();

21         char
22             * slovo [64], // адреса слов в тексте
23             R [128], // текст-результат
24             T [128]; // данный текст

25         printf ("введите текст: ");
26         gets (T);

27         // найдем адреса всех слов
28         slovo [0] = strtok (T, " ");
29         int K = 1; // число слов в предложении
30         while ( (slovo [K ++] = strtok (NULL, " ")) != NULL )
31             ;
32         // последний адрес - нулевой, ему нет соответствующего слова

33         int i, j; // индексы массива слов

34         for (i = 0; i < K - 1; i ++)
35             printf ("slovo [%i] = %s\n", i, slovo [i]);
36         printf ("\n");

37         // все адреса одинаковых слов сделаем нулевыми
38         char flag;

```

```

39     for (i = 0; i < K - 2; i ++)
40     {
41         flag = 0;
42         for (j = i + 1; j < K - 1; j ++)
43             if (strcmp (slovo [i], slovo [j]) == 0)
44             {
45                 flag = 1;
46                 slovo [j] = NULL;
47             }
48         if (flag == 1)
49             slovo [i] = NULL;
50     }

51     for (i = 0; i < K - 1; i ++)
52         printf ("slovo [%i] = %s\n", i, slovo [i]);

53     // склеим в текст-результат все слова, адрес у которых не ноль
54     strcpy (R, "");
55     for (i = 0; i < K - 1; i ++)
56         if (slovo [i] != NULL)
57         {
58             strcat (R, slovo [i]);
59             strcat (R, " ");
60         }

61     // затрем последний пробел
62     R [strlen (R) - 1] = '\0';
63     printf ("\nR = %s", R);

64     getch ();

65     return 0;
66 }

```

Sergey Mitrofanov, 21.09.13

E-mail: infostar@mail.ru

© <http://www.Best-Listing.ru>, 2006–2013