

```

1      /*
      MSP, 30.10.07, 14:43

      Вопрос.
      Что такое стрелка: -> ?

      Объяснение.
      Part 2.

      Ниже читаем ответ.
      */

9      # include <stdio.h>
10     # include <conio.h>

11     class spisok
12     {
13     public:
14         spisok * L; // указатель на элемент списка слева
15         int data; // данные элемента списка
16         spisok * R; // указатель на элемент списка справа
17     }; // это правильно, здесь НУЖНО ставить ;

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

21         spisok element; // создана новая переменная element

22         // двусвязный список-элемент - дек - кольцо состоит из элементов.

23         // Пока порожден ВСЕГО 1 элемент списка, и на сей момент он
24         // весь "мусорный" из-за неопределенной информации, случайно
25         // имеющейся в памяти машины, программист еще ничего сам не
26         // записывал в element.

27         // Теперь разберемся в том, как же добраться до ПОЛЕЙ СЛОЖНОЙ
28         // переменной element, и именно до тех маленьких переменных,
29         // из которых она сама состоит?

30         // Как обратиться к L, data, R?
31         // Ответ: через ТОЧКУ.

32         // Вот так:
33         // element.L - обратились к указателю на предыдущий элемент
34         // списка
35         // element.data - обратились к данному элемента списка
36         // element.R - обратились к указателю на следующий элемент
37         // списка

```

```

38 // Проинициализируем element
39 // Выделим память для L и R, зададим значение data

40 element.L = new spisok; // сама OS даст адресное пространство
41 // A2..(A2 + 9)

42 printf ("Введите число: ");
43 scanf ("%i", & element.data); // введем 20
44 fflush (stdin);

45 element.R = new spisok; // сама OS даст адресное пространство
46 // A3..(A3 + 9)

47 // Итак, с адреса A1 располагается в памяти PC переменной
48 // element
49 // теперь оно строго определено самой OS по требованию
50 // программиста:
51 // A2, 20, A3

52 // element == A2 + 123 + A3 - единое целое, записанное по адресу
53 // A1

54 // element имеет АДРЕС A1
55 // В element.L записан АДРЕС A2
56 // в element.data записано, например, число 20
57 // в element.R записан АДРЕС A3
58 // Все адреса предоставлены OS

59 // element.L == адрес, он имеет тип spisok *
60 // В списке дека сейчас всего 1 элемент == element
61 // Получим элемент списка СЛЕВА, то есть предыдущий
62 // Создадим его.
63 // Память под него выделена OS - 10 байт, но они все "мусорные",
64 // проинициализируем предыдущий элемент

65 // | element.L - адрес A2
66 // |
67 // |
68 // v
69 // (L, data, R) ===== (* element.L) - то, что лежит по адресу A2

70 (* element.L).L = new spisok; // получили адрес A4, для
71 // элемента, который находится
72 // левее, чем A2

73 printf ("Введите число: ");
74 scanf ("%i", & (* element.L).data); // введем 10
75 fflush (stdin);

76 (* element.L).R = new spisok; // получили адрес A5, для

```

```

77                                     // элемента, который находится
78                                     // правее, чем A2

79     // заполним элемент, правее element
80     (* element.R).L = new spisok;

81     printf ("Введите число: ");
82     scanf ("%i", & (* element.R).data); // введем 30
83     fflush (stdin);

84     (* element.R).R = new spisok;

85     printf ("element.data = %i\n", element.data);
86     printf ("(* element.L).data = %i\n", (* element.L).data);
87     printf ("(* element.R).data = %i\n", (* element.R).data);

88     // теперь смотрите дальнейшее объяснение, part 3.

89     getch ();

90     return 0;
91 }

```

Listing данной задачи опубликован в сети Internet по адресу
<http://www.Best-Listing.ru/color-3-task-19.html>

Sergey Mitrofanov, 30.10.13, 21:25

E-mail: infostar@mail.ru

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