

# 利用间接指针和指针数组进行排序的程序的分析

2022 年 8 月 16 日

1. 阅读程序，回答相应问题。

---

```
1  #include <stdio.h>
2
3  /*利用间接指针和指针数组进行排序*/
4
5  int main()
6  {
7      // 数据数组data_array
8      int data_array[] = {5, 2, 4, 6, 1, 3};
9
10     // 分配与data_array等长的地址数组addr_array
11     int *addr_array[sizeof(data_array) / sizeof(int)]; // 索引数组
12
13     int len = sizeof(data_array) / sizeof(int); // 两数组的长度
14
15     // 建立索引关系
16     for(int i = 0; i < len; i++)
17     {
18         addr_array[i] = data_array + i;
19     }
20
21     // 输出数据数组data_array的内容
22     printf("Before sorting, the data array:\n");
23
24     for(int i = 0; i < len; i++)
25     {
26         printf("%d\t", data_array[i]);
27     }
```

```

28
29 printf("\n");
30
31 // 输出依次被addr_array中的元素索引的元素
32 printf("The elements indexed by pointers:\n");
33
34 for(int i = 0; i < len; i++)
35 {
36     printf("%d\t", *(*(addr_array + i)));
37 }
38
39 printf("\n");
40
41 printf("About to sort...\n");
42
43 // 注意，以下操作不改变数据数组data_array
44 // 只改变索引数组addr_array
45
46 // 从下标head开始进行排序调整索引数组addr_array
47 for(int head = 0; head < len - 1; head++)
48 {
49     for(int i = head + 1; i < len; i++)
50     {
51         if(*addr_array[i] < *addr_array[head]) // 比较被索引的元素
52         {
53             // 如果索引目标元素较小，那么相应的索引将提到前面
54             int *temp_addr = addr_array[i];
55             addr_array[i] = addr_array[head];
56             addr_array[head] = temp_addr;
57         }
58     }
59
60     printf("%d pass(es) completed.\n", head + 1);
61
62     // 按索引输出data_array中的元素
63     printf("The elements indexed by pointers:\n");
64
65     for(int i = 0; i < len; i++)
66     {

```

```

67         printf("%d\t", *((addr_array + i)));
68     }
69
70     printf("\n");
71 }
72
73
74 printf("After sorting, the data array:\n");
75
76 for(int i = 0; i < len; i++)
77 {
78     printf("%d\t", data_array[i]);
79 }
80
81 printf("\n");
82
83 return 0;
84 }

```

---

- (a) 按顺序写出第 65-68 行先后输出的内容。
- (b) 按顺序写出第 76-79 行输出的内容。
- (c) 请给出第 51 行被判断的次数，并简要说明理由。
- (d) 请给出第 54 行被执行的次数，并简要说明理由。