

1. 给你  $n$  个同学的成绩，请你统计并给出成绩大于等于平均分的同学的平均分。  
输入共 2 行，第一行，一个整数  $n$ ， $1 \leq n \leq 1000$ ，第二行， $n$  个整数，表示  $n$  个同学的成绩。输出为一个小数，表示所求平均分，请保留两位小数。

输入样例：

10

89 76 20 66 59 92 100 70 85 88

输出样例：

88.33

```
Language Run Stop Run Settings...
c语言第七次作业1.c
1  #include<stdio.h>
2
3  int main()
4  {
5      int n,k=0,i,a[1000];
6      double aver=0,upperaver=0;
7
8      scanf("%d",&n); //输入一个整数 n, 1≤n≤1000
9
10     for(i=0; i<n; i++)
11     {
12         scanf("%d",&a[i]); //输入成绩
13         aver=aver+1.0*a[i];
14     }
15     aver=aver/n; //平均分的计算
16     for (i=0;i<n;i++)
17     {
18         if(a[i]>=aver) //大于等于平均分的同学的平均分
19         {
20             upperaver=upperaver+a[i];
21             k++;
22         }
23     }
24
25     printf("%.2f",upperaver/k); //成绩除了那个n数，保留两位小数
26
27     return 0;
28 }
29
30
31
32
33 }
```

10  
89 76 20 66 59 92 100 70 85 88  
88.33

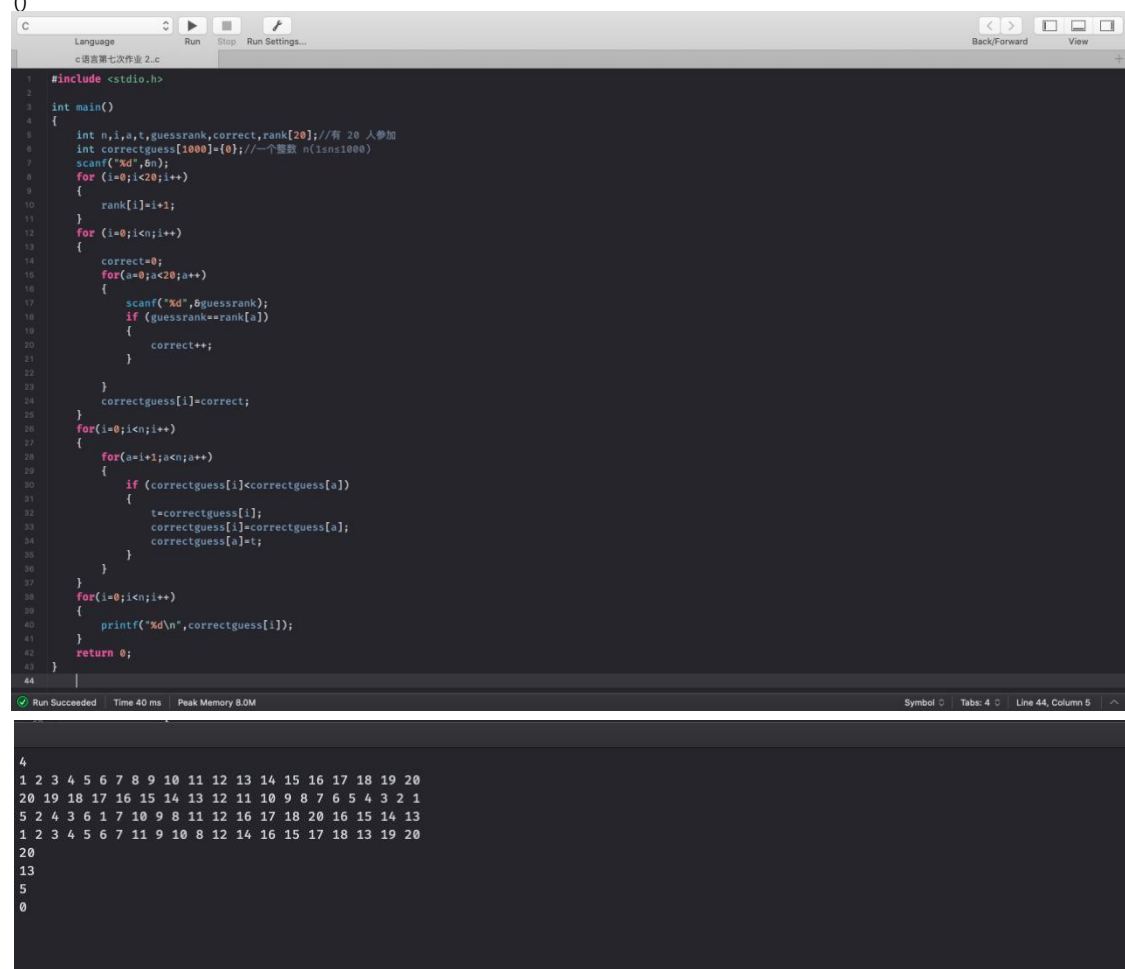
2. 有一场比赛，共有 20 人参加，比赛结果为这 20 人依次获得第一名到第 20 名。在赛前，开展了比赛结果竞猜活动，共有  $n$  名参与者提前给出他们对此场比赛结果的预测名次。现在请你进行统计，按照猜对人数从大到小进行排序输出。输入：共有  $n+1$  行，第一行，一个整数  $n(1 \leq n \leq 1000)$ ，接下来共  $n$  行，每行给出 20 个用空格分隔的整数，依次是对 20 名选手预测的名次。输出共  $n$  行，代表从大到小排好序的猜对人数。

输入样例：

```
4
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
5 2 4 3 6 1 7 10 9 8 11 12 16 17 18 20 16 15 14 13
1 2 3 4 5 6 7 11 9 10 8 12 14 16 15 17 18 13 19 20
```

输出样例：

```
20
13
5
0
```



```
1 #include <stdio.h>
2
3 int main()
4 {
5     int n,i,a,t,guessrank,correct,rank[20]; //有 20 人参加
6     int correctguess[1000]={0}; //一个整数 n(1<=n<=1000)
7     scanf("%d",&n);
8     for (i=0;i<20;i++)
9     {
10         rank[i]=i+1;
11     }
12     for (i=0;i<n;i++)
13     {
14         correct=0;
15         for(a=0;a<20;a++)
16         {
17             scanf("%d",&guessrank);
18             if (guessrank==rank[a])
19             {
20                 correct++;
21             }
22         }
23         correctguess[i]=correct;
24     }
25     for(i=0;i<n;i++)
26     {
27         for(a=i+1;a<n;a++)
28         {
29             if (correctguess[i]<correctguess[a])
30             {
31                 t=correctguess[i];
32                 correctguess[i]=correctguess[a];
33                 correctguess[a]=t;
34             }
35         }
36     }
37     for(i=0;i<n;i++)
38     {
39         printf("%d\n",correctguess[i]);
40     }
41     return 0;
42 }
43
44
```

```
4
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
5 2 4 3 6 1 7 10 9 8 11 12 16 17 18 20 16 15 14 13
1 2 3 4 5 6 7 11 9 10 8 12 14 16 15 17 18 13 19 20
20
13
5
0
```