

1 有 5 个学生，每个学生的数据包括学号、姓名、3 门课的成绩，从键盘输入每个学生的数据，打印出每个学生的总分及三门课的平均成绩。注意：使用结构体，自定义两个函数（非 main 函数），一个读入数据，一个实现输出。

输入样例：

202001 wang 60 80 78

202002 tang 75 88 91

202003 sang 83 92

88 输出样例：

num:202001

name:wang

total :218.00

avg:72.67

num:202002

name:tang

total :254.00

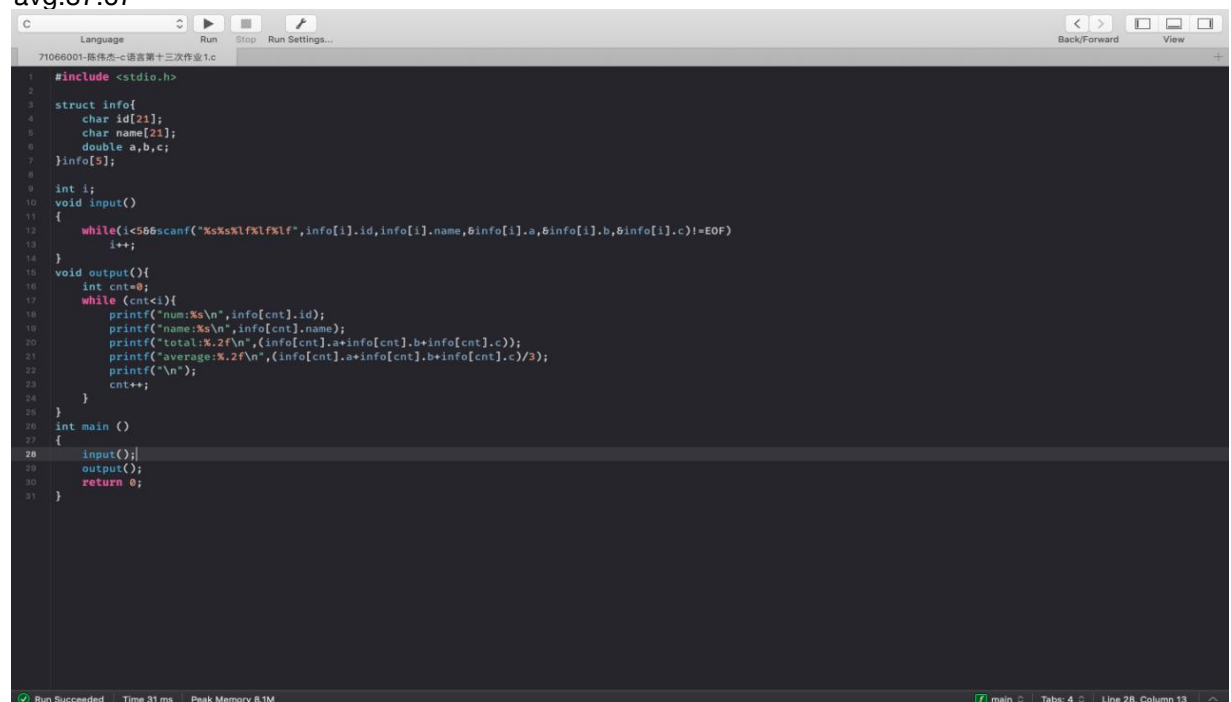
avg:84.67

num:202003

name:sang

total :263.00

avg:87.67



```
1 #include <stdio.h>
2
3 struct info{
4     char id[21];
5     char name[21];
6     double a,b,c;
7 }info[5];
8
9 int i;
10 void input()
11 {
12     while(i<5&&scanf("%s%lf%lf%lf",info[i].id,info[i].name,&info[i].a,&info[i].b,&info[i].c)!=EOF)
13         i++;
14 }
15 void output(){
16     int cnt=0;
17     while (cnt<i){
18         printf("num:%s\n",info[cnt].id);
19         printf("name:%s\n",info[cnt].name);
20         printf("total:%.2f\n",info[cnt].a+info[cnt].b+info[cnt].c);
21         printf("average:%.2f\n",info[cnt].a+info[cnt].b+info[cnt].c)/3);
22         printf("\n");
23         cnt++;
24     }
25 }
26 int main ()
27 {
28     input();
29     output();
30     return 0;
31 }
```

Run Succeeded Time 31 ms Peak Memory 8.1M

C

LanguageRunStopRun Settings...

71066001-陈伟杰-C语言第十三次作业1.c

Back/ForwardView

```
1  #include <stdio.h>
2
3  struct info{
4      char id[21];
5      char name[21];
6      double a,b,c;
7  }info[5];
8
9  int i;
10 void input()
11 {
12     while(i<500scanf("%s%s%lf%lf%lf",info[i].id,info[i].name,&info[i].a,&info[i].b,&info[i].c)!=EOF)
13         i++;
14 }
```

FilterAll Output

202001 wang 60 80 78
202002 tang 75 88 91
202003 sang 83 92 88
num:202001
name:wang
total:218.00
average:72.67
num:202002
name:tang
total:254.00
average:84.67
num:202003
name:sang
total:263.00
average:87.67

Run SucceededTime 42 msPeak Memory 8.3MmainTab: 4Line 28, Column 13

2. 现有 $N(0 < N \leq 100)$ 名同学参加了期末考试，并且获得了每名同学的信息：姓名（不超过 8 个字符的字符串，没有空格）、语文、数学、英语成绩（均为不超过 150 的自然数）。总分最高的学生就是最厉害的，请输出最厉害的学生各项信息（姓名、各科成绩）。如果有多个总分相同的学生，输出靠前的那位。

输入样例：

3

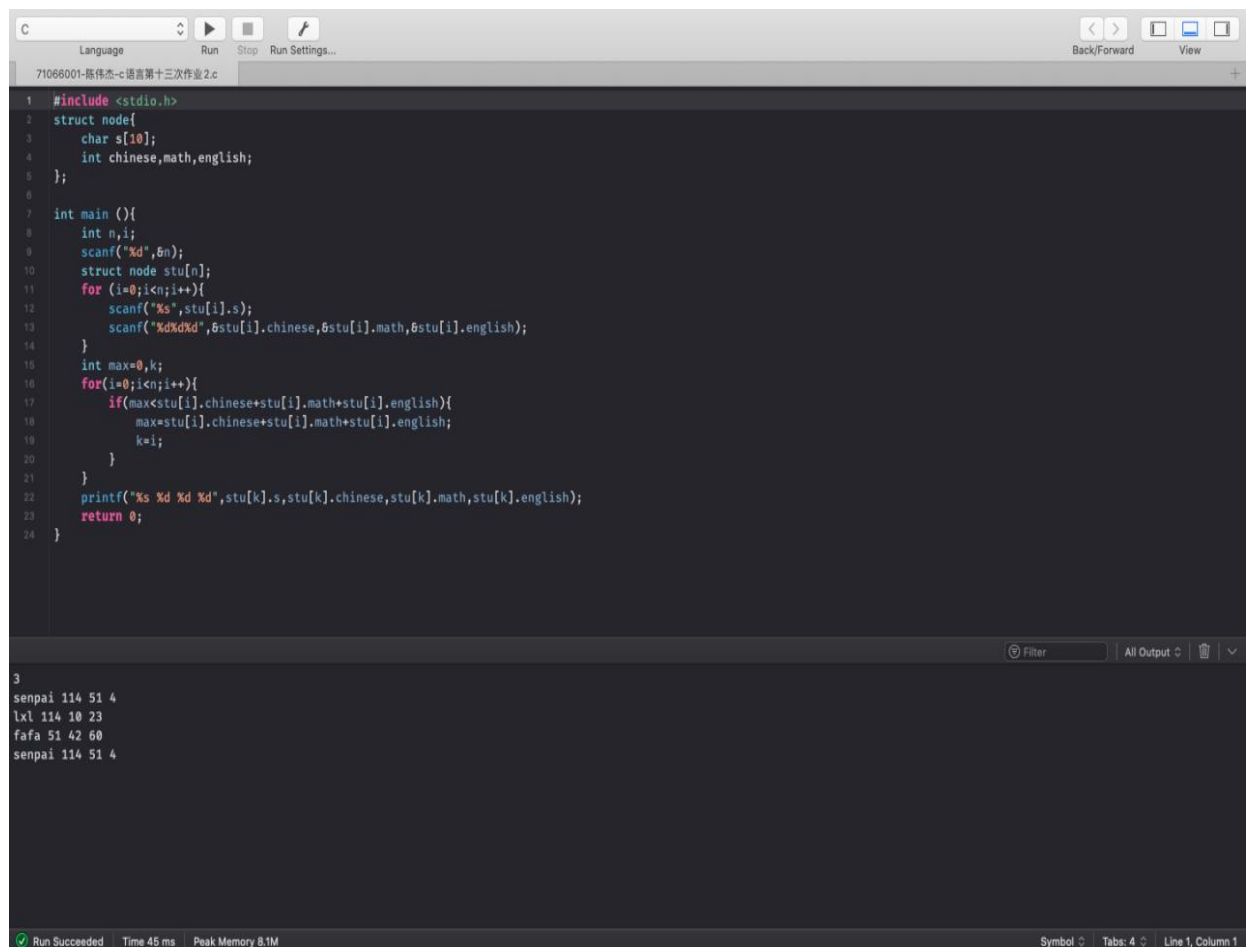
senpai 114 51 4

lxl 114 10 23

fafa 51 42 60

输出样例：

senpai 114 51 4



```
1 #include <stdio.h>
2 struct node{
3     char s[10];
4     int chinese, math, english;
5 };
6
7 int main(){
8     int n, i;
9     scanf("%d", &n);
10    struct node stu[n];
11    for(i=0; i<n; i++){
12        scanf("%s", stu[i].s);
13        scanf("%d%d%d", &stu[i].chinese, &stu[i].math, &stu[i].english);
14    }
15    int max=0, k;
16    for(i=0; i<n; i++){
17        if(max<stu[i].chinese+stu[i].math+stu[i].english){
18            max=stu[i].chinese+stu[i].math+stu[i].english;
19            k=i;
20        }
21    }
22    printf("%s %d %d %d", stu[k].s, stu[k].chinese, stu[k].math, stu[k].english);
23    return 0;
24 }
```

3
senpai 114 51 4
lxl 114 10 23
fafa 51 42 60
senpai 114 51 4

Run Succeeded Time 45 ms Peak Memory 8.1M

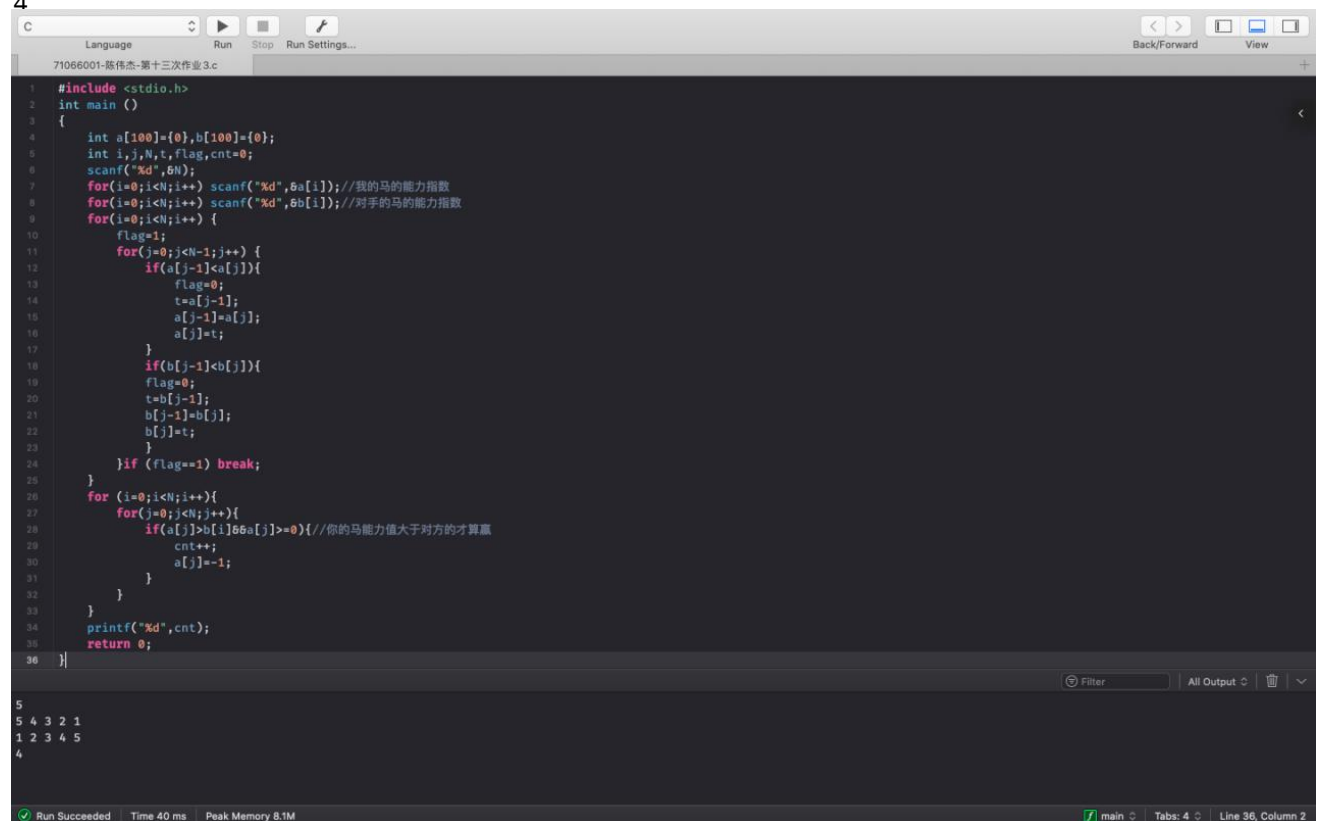
3 你和对手赛马,双方都有 n 匹马,每匹马的能力数值 $a_i(0 \leq a_i \leq 1000)$, 由你来规定双方马匹出场的顺序,请问你最多能赢多少场(要求对局时候你的马能力值大于对方的才算赢,等于算平手)。输入三行: 第一行, 一个整数 $n(1 \leq n \leq 100)$ 。接下来的 2 行, 每行 n 个整数 a_1, a_2, \dots, a_n , 用空格分割, 代表每匹马的能力。第二行的指数是你的马的能力指数, 第三行则是对手的。输出为一个整数, 表示你最多胜利的场次。

输入样例:

```
5
5 4 3 2 1
1 2 3 4 5
```

输出样例:

```
4
```



```
1 #include <stdio.h>
2 int main ()
3 {
4     int a[100]={0},b[100]={0};
5     int i,j,N,t,flag,cnt=0;
6     scanf("%d",&N);
7     for(i=0;i<N;i++) scanf("%d",&a[i]); //我的马的能力指数
8     for(i=0;i<N;i++) scanf("%d",&b[i]); //对手的马的能力指数
9     for(i=0;i<N;i++) {
10         flag=1;
11         for(j=0;j<N-1;j++) {
12             if(a[j-1]<a[j]){
13                 flag=0;
14                 t=a[j-1];
15                 a[j-1]=a[j];
16                 a[j]=t;
17             }
18             if(b[j-1]<b[j]){
19                 flag=0;
20                 t=b[j-1];
21                 b[j-1]=b[j];
22                 b[j]=t;
23             }
24             if (flag==1) break;
25         }
26         for (i=0;i<N;i++){
27             for(j=0;j<N;j++){
28                 if(a[j]>b[i]66a[j]>=0){ //你的马能力值大于对方的才算赢
29                     cnt++;
30                     a[j]=-1;
31                 }
32             }
33         }
34         printf("%d",cnt);
35         return 0;
36     }
```

```
5
5 4 3 2 1
1 2 3 4 5
4
```

Run Succeeded Time 40 ms Peak Memory 8.1M