1. 编写一个程序，用户输入5个整数，该程序通过调用函数，对5个整数按照从小到大的顺序排列并输出。

源代码:

|  |
| --- |
| #include <stdlib.h>  #include<stdio.h>  void print\_sort(int \*arr, int n);  int main()  {  int a[5];  int c;  for(int i=0;i<5;i++){  scanf("%d",&a[i]);  }  print\_sort(a,5);  return 0;  }  void print\_sort(int \*arr,int n){  int b;  int c=0;  int i=0,j=0;  for(i=0;i<n;i++){  for(j=0;j<n-i;j++){  if(\*(arr+j)>\*(arr+j+1)){  b=\*(arr+j+1);  \*(arr+j+1)=\*(arr+j);  \*(arr+j)=b;  }  }  }  for(c;c<5;c++){  printf("%d ",\*(arr+c));  }  } |

输出:

|  |
| --- |
| 5 4 1 2 3  1 2 3 4 5 |

1. 编写一个程序，用户输入一个日期和一个整数n，该程序通过调用函数，输出该日期向后n天的日期，如果n<0即为向前的日期。需要考虑闰年。

源代码:

|  |
| --- |
| #include <stdio.h>  void find\_date(int \*p1,int \*p2,int \*p3,int x);  int main(void){  int year,month,day;  int n=0;  scanf("%d %d %d %d",&year,&month,&day,&n);  find\_date(&year,&month,&day,n);  printf("%02d-%02d-%02d\n",year,month,day);  return 0;  }  void find\_date(int \*p1, int \*p2, int \*p3, int x){  int a[12]={31,28,31,30,31,30,31,31,30,31,30,31};  \*p3=\*p3+x;  if(\*p1%4==0){  if (\*p1 % 100 == 0){  if (\*p1 % 400 == 0){  a[2] = 29;  }  a[2] = 28;  }  a[2] = 29;  }  else{  a[2] = 28;  }  if (\*p3 > a[\*p2]){  for (; \*p3 > a[\*p2];){  \*p3 = \*p3 - a[\*p2];  \*p2++;  if (\*p2 == 13){  \*p1++;  \*p2 = 1;  }  }  }  if (\*p3 < 0){  for (; \*p3 < 0;){  \*p2 = \*p2 - 1;  \*p3 = a[\*p2] + \*p3;  if (\*p2 == 0){  \*p2 = 12;  \*p1--;  }  }  }  } |

输出:

|  |
| --- |
| 2000 2 28 3  2000-02-02 |

1. 编写一个程序，用户输入4个学生的学号和他们在5门课程的成绩，该程序通过调用不同的函数，分别实现三个功能：（1）计算第n门课程的成绩；（2）找出有n门或n门以上课程不及格的学生，输出他们的学号和全部课程成绩及平均成绩；（3）找出平均成绩大于等于avg\_score或全部课程成绩大于等于min\_score的学生，输出他们的学号和全部课程成绩及平均成绩。

源代码:

|  |
| --- |
| #include <stdio.h>  #define NUM\_STUDENTS 4  #define NUM\_CLASSES 5  #define ID\_LENGTH 15  void avg\_class(int \*scores, int n);  void find\_fail(char (\*ids)[ID\_LENGTH], int \*scores, int n);  void find\_better(char \*ids, int (\*scores)[NUM\_CLASSES], int avg\_score, int min\_score);  int main(void){  char ids[NUM\_STUDENTS][ID\_LENGTH];  int scores[NUM\_STUDENTS][NUM\_CLASSES];  int n=0;  int avg\_score=0,min\_score=0;  for(int i=0;i<NUM\_STUDENTS;i++){  scanf("%s",ids[i]);  for (int j=0;j<NUM\_CLASSES;j++){  scanf("%d",&scores[i][j]);  }  }  printf("\n要计算平均成绩的课程序号：[0--%d]",NUM\_CLASSES-1);  scanf("%d",&n);  avg\_class(\*scores,n);  printf("\n要统计不及格课程的数量：[1--%d]",NUM\_CLASSES);  scanf("%d",&n);  find\_fail(ids,\*scores,n);  printf("\n不低于的平均成绩：");  scanf("%d",&avg\_score);  printf("不低于的全部成绩：");  scanf("%d",&min\_score);  find\_better(\*ids,scores,avg\_score,min\_score);  return 0;  }  void avg\_class(int \*scores,int n){  int i;  float sum=0,avg;  scores+=n;  for(i=0;i<4;i++){  sum=sum+\*scores;  scores+=5;  }  avg=sum/4;  printf("序号%d的课程平均成绩为%.2f\n",n,avg);  }  void find\_fail(char (\*ids)[ID\_LENGTH],int \*scores,int n){  int i,z=0,j,e;  float sum=0,avg;  for(j=0;j<4;j++){  for(i=0;i<5;i++){  if(\*scores<60){  z++;  }  scores++;  }  if(z>=n){  ids+=j;  printf("%s",\*ids);  scores-=5;  printf("全部成绩:");  for(e=0;e<5;e++){  printf(" %d",\*scores);  sum+=\*scores;  scores++;  }  avg=sum/5;  printf(" 平均成绩:%.2f",avg);  }  ids-=j;  z=0;  }  }  void find\_better(char \*ids,int (\*scores)[NUM\_CLASSES],int avg\_score,int min\_score){  int i,j,z=0,e,t,a;  float sum=0,avg;  for(i=0;i<4;i++){  for(j=0;j<5;j++){  if(\*(\*(scores+i)+j)>=min\_score){  z++;  }  sum+=\*(\*(scores+i)+j);  }  avg=sum/5;  if((z==5)||(avg>=avg\_score)){  for(t=0;t<15;t++){  printf("%c",\*ids);  ids++;  }  printf(" 全部成绩：");  for(e=0;e<5;e++){  printf(" %d",\*(\*(scores+i)+e));  }  printf(" 平均成绩:%.2f\n",avg);  }  else{  for(a=0;a<15;a++){  ids++;  }  }  sum=0;  z=0;  }  } |

输出:

|  |
| --- |
| 2019010501 61 63 65 67 69  2019010502 71 45 83 28 74  2019010503 83 92 95 78 89  2019010504 56 65 77 74 79  要计算平均成绩的课程序号：[0--4]3  序号3的课程平均成绩为61.75  要统计不及格课程的数量：[1--5]2  2019010502全部成绩: 71 45 83 28 74 平均成绩:60.20 |