

Data Structure first lab

First Lab in Data Structure

Student Name: Mohamed Ibrahim Mohamed El-Saeed Ibrahim , ID: 18011342

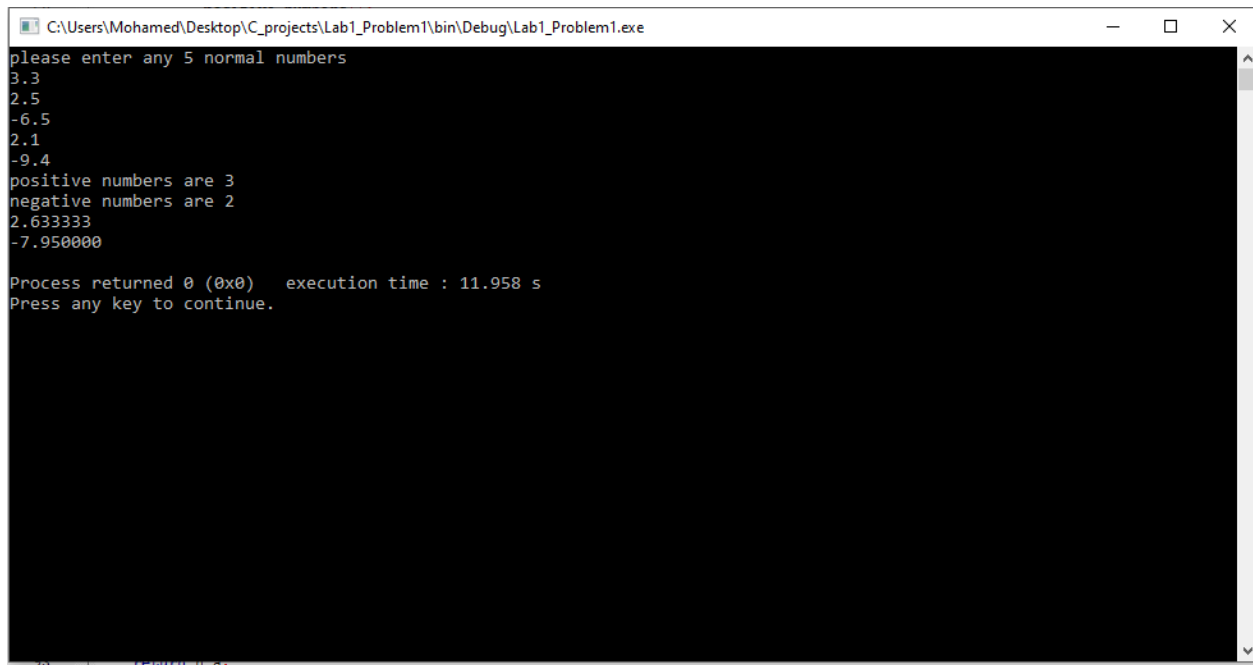
Mohamed Ibrahim
[Pick the date]

1st problem:-

What I did in this project:

- 1- I defined some variables and specially I defined array of 5 elements
- 2- I made a for loop to scan 5 values to the array
- 3- I made 2 functions which I pass the array with it's 5 elements to them to count the +ve and the –ve numbers in it
- 4- I made another 2 functions and passed to them the array with it's 5 elements and also the number of negative or positive values in the array to calculate the average of the negative and positive values
- 5- each function return what I wanted from it then the values got printed.

Attempt:



```
C:\Users\Mohamed\Desktop\C_projects\Lab1_Problem1\bin\Debug\Lab1_Problem1.exe
please enter any 5 normal numbers
3.3
2.5
-6.5
2.1
-9.4
positive numbers are 3
negative numbers are 2
2.633333
-7.950000
Process returned 0 (0x0)   execution time : 11.958 s
Press any key to continue.
```

Code images :

```
1  #include<stdio.h>
2  #include<stdlib.h>
3  int count_positive(float number[5]);
4  int count_negative(float number[5]);
5
6  float average_of_positives (float number[5],int s);
7  float average_of_negatives (float number[5],int s);
8
9  int main(){
10     int i,positive_number,negative_number;
11     float p_n_a,n_n_a;
12     float numbers [5];
13
14     printf("please enter any 5 normal numbers\n");
15
16     for (i=0;i<5;i++){
17         scanf("%f",&numbers[i]);
18     }
19
20     positive_number = count_positive(&numbers[0]);
21     negative_number = count_negative(&numbers[0]);
22
23     printf("positive numbers are %d \n",positive_number);
24     printf("negative numbers are %d \n",negative_number);
25
26     p_n_a = average_of_positives(numbers,positive_number);
27
28     printf("%f\n",p_n_a);
29
30     n_n_a = average_of_negatives(numbers,negative_number);
31     printf("%f\n",n_n_a);
32
33
34     return 0;
35 }
36
37 int count_negative(float number[5]){
38     int negative_numbers =0;
39     int i;
40     for (i=0; i<5;i++){
41         if( number[i]<0 ){
42             negative_numbers++;
43         }
```

```
40     int i;
41     for (i=0; i<5;i++){
42         if( number[i]<0 ){
43             negative_numbers++;
44         }
45         else
46         {
47             negative_numbers=negative_numbers;
48         }
49     }
50     return negative_numbers;
51 }
52
53 int count_positive(float number[5]){
54     int positive_numbers =0;
55     int i;
56     for (i=0; i<5;i++){
57         if( number[i]>0 ){
58             positive_numbers++;
59         }
60         else
61         {
62             positive_numbers=positive_numbers;
63         }
64     }
65     return positive_numbers;
66 }
67
68 float average_of_positives (float number[5],int s)
69 {
70     int i;
71     float p_a=0;
72
73     for (i=0;i<5;i++){
74         if (*(number+i)>0)
75             p_a = p_a + number[i];
76     }
77     p_a = p_a / s;
78     return p_a;
79 }
80
81
82
```

```
56     int i;
57     for (i=0; i<5;i++){
58         if( number[i]>0 ){
59             positive_numbers++;
60         }
61         else
62         {
63             positive_numbers=positive_numbers;
64         }
65     }
66     return positive_numbers;
67 }
68
69 float average_of_positives (float number[5],int s)
70 {
71     int i;
72     float p_a=0;
73
74     for (i=0;i<5;i++){
75         if (*(number+i)>0){
76             p_a = p_a + number[i];
77         }
78     }
79     p_a = p_a / s;
80     return p_a;
81 }
82
83 float average_of_negatives (float number[5],int s)
84 {
85     int i;
86     float n_a=0;
87
88     for (i=0;i<5;i++){
89         if (number[i]<0){
90             n_a = n_a + number[i];
91         }
92     }
93     n_a = n_a / s;
94     return n_a;
95 }
```

code:

```
#include<stdio.h>

#include<stdlib.h>

int count_positive(float number[5]);

int count_negative(float number[5]);

float average_of_positives (float number[5],int s);

float average_of_negatives (float number[5],int s);

int main(){

    int i,positive_number,negative_number;

    float p_n_a,n_n_a;

    float numbers [5];

    printf("please enter any 5 normal numbers\n");

    for (i=0;i<5;i++){

        scanf("%f", &numbers[i]);

    }

    positive_number = count_positive(&numbers[0]);

    negative_number = count_negative(&numbers[0]);

    printf("positive numbers are %d \n",positive_number);
```

```
printf("negative numbers are %d \n",negative_number);
```

```
p_n_a = average_of_positives(numbers,positive_number);
```

```
printf("%f\n",p_n_a);
```

```
n_n_a = average_of_negatives(numbers,negative_number);
```

```
printf("%f\n",n_n_a);
```

```
return 0;
```

```
}
```

```
int count_negative(float number[5]){
```

```
int negative_numbers =0;
```

```
int i;
```

```
for (i=0; i<5;i++){
```

```
    if( number[i]<0 ){
```

```
        negative_numbers++;
```

```
    }
```

```
    else
```

```
    {
```

```
        negative_numbers=negative_numbers;
```

```
    } }
```

```
return negative_numbers;
}
```

```
int count_positive(float number[5]){
int positive_numbers =0;
int i;
    for (i=0; i<5;i++){
        if( number[i]>0 ){
            positive_numbers++;
        }
        else
        {
            positive_numbers=positive_numbers;
        }
    }
return positive_numbers;
}
```

```
float average_of_positives (float number[5],int s)
{
    int i;
    float p_a=0;

    for (i=0;i<5;i++){
```

```
        if (*(number+i)>0)
            p_a = p_a + number[i];
    }
    p_a = p_a / s;
    return p_a;
}
```

```
float average_of_negatives (float number[5],int s)
{
    int i;
    float n_a=0;

    for (i=0;i<5;i++){
        if (number[i]<0)
            n_a = n_a + number[i];
    }
    n_a = n_a / s;
    return n_a;
}
```

2nd problem:-

What I did in this project:

- 1- I defined some variables and specially I defined 2d array of 9*3 elements
- 2- I made a for loop set the 3rd column values to zero to make It easy to increment it.
- 3- I made a while loop to continuously scan values that we need to enter until we entered -1,
- 4- And the values that we scanned we use them to know the over-all salary of the employer.
- 5- I made a temp variable and set its value to 200 and used it to know this employer's salary in any range from the 9 ranges by using a for loop and checking the salary that we calculated with the temp and temp+99 if it doesn't match then temp will be incremented by 100 so I will search in the next range between 300 and 399 and etc...
- 6- for example if the salary was 450 it will know that it is between 400 and 499 so the 3rd column of this row in the main 2d array will get incremented by 1
- 7- after this it resets the temp that we use to 200 again and waits for another input to enter from the user.
- 8- if we entered -1 the while loop will end and a for loop will begin to finally set the final shape of the array and print it.

Attempt:

```
Enter Employee gross sale (-1 to end) : 1000
Employee salary is 290
commission_base[0][2]=1
Enter Employee gross sale (-1 to end) : 2000
Employee salary is 300
commission_base[1][2]=1
Enter Employee gross sale (-1 to end) : 5000
Employee salary is 650
commission_base[4][2]=1
Enter Employee gross sale (-1 to end) : 6000
Employee salary is 740
commission_base[5][2]=1
Enter Employee gross sale (-1 to end) : 70000
Employee salary is 6500
commission_base[8][2]=1
Enter Employee gross sale (-1 to end) : 20000
Employee salary is 2000
commission_base[8][2]=2
Enter Employee gross sale (-1 to end) : 500
Employee salary is 245
commission_base[0][2]=2
Enter Employee gross sale (-1 to end) : 600000
Employee salary is 54200
commission_base[8][2]=3
Enter Employee gross sale (-1 to end) : 20000
Employee salary is 2000
commission_base[8][2]=4
Enter Employee gross sale (-1 to end) : 1000
Employee salary is 290
commission_base[0][2]=3
Enter Employee gross sale (-1 to end) : -1
200    299    3
300    399    1
400    499    0
500    599    0
600    699    1
700    799    1
800    899    0
900    999    0
1000 and more 4

Process returned 0 (0x0)   execution time : 25.019 s
Press any key to continue.
```

Code:-

```
#include<stdio.h>

#include<stdlib.h>


int main(){

    int commission_base[9][3];

    int cond=0;

    int salary;

    // setting my main array with filling it's 3rd column with zeros;


    int i,temp,temp2,temp3;

    temp =200;

    for (i=0;i<9;i++)

    {

        commission_base[i][2] = 0;

    }

    //

    temp3=200;

    while (cond!=-1){

        printf("Enter Employee gross sale (-1 to end) : ");

        scanf("%d",&temp2);

        if(temp2==-1){

            cond=-1;
```

```

        break;
    }

    salary = 200 + (9 * temp2)/100;

    printf("Employee salary is %d\n", salary);

    for (i=0;i<9;i++){

        if( temp3+99 > salary && salary >= temp3){

            commission_base[i][2] = commission_base[i][2] +1;

            printf("commission_base[%d][2]=%d\n",i,commission_base[i][2]);

            i=9;

        }

        else if (salary >= 1000 ){

            i=8;

            commission_base[i][2] = commission_base[i][2] +1;

            printf("commission_base[%d][2]=%d\n",i,commission_base[i][2]);

        }

        temp3 = temp3+100;

    }

    temp3=200;

}

//setting the final shape and printing it;

for (i=0;i<9;i++)

{

    commission_base[i][0] = temp;

    temp = temp + 99;

    printf("%d",commission_base[i][0]);

```

```

    if(temp<1000)
    {
        commission_base[i][1] = temp;

        temp = temp + 1;

        printf("    %d ",commission_base[i][1]);

    }

    else

        printf (" and more ");

    printf(" %d \n",commission_base[i][2]);

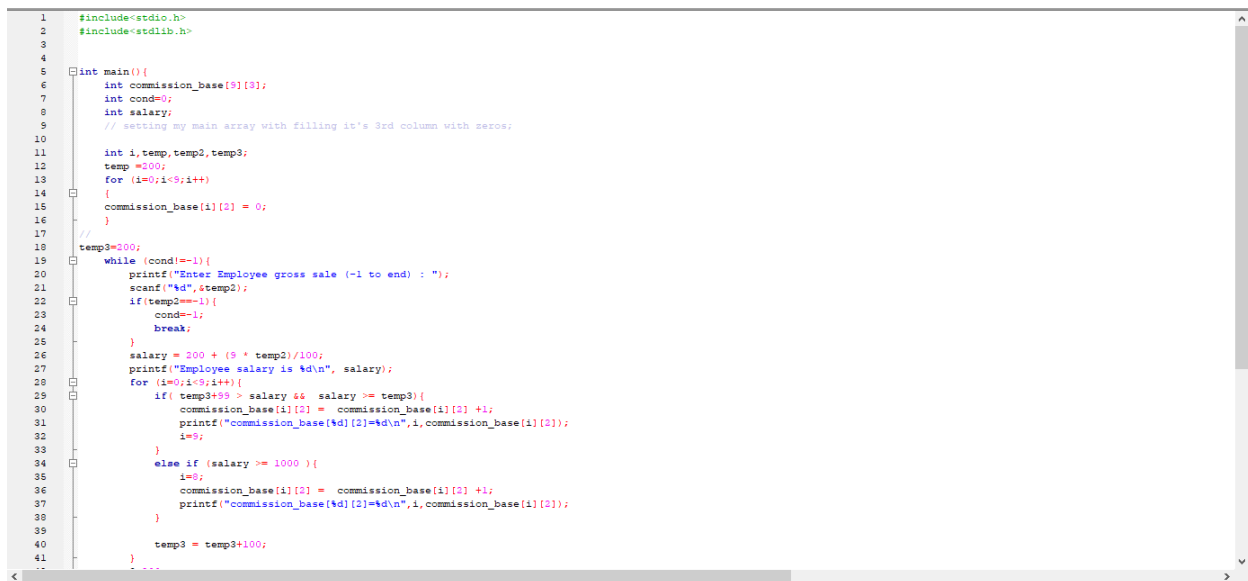
}

return 0;

}

```

Code images:-



```

1  #include<stdio.h>
2  #include<stdlib.h>
3
4
5  int main(){
6      int commission_base[9][3];
7      int cond=0;
8      int salary;
9      // setting my main array with filling it's 3rd column with zeros;
10
11      int i,temp,temp2,temp3;
12      temp =200;
13      for (i=0;i<9;i++)
14      {
15          commission_base[i][2] = 0;
16      }
17      //
18      temp3=200;
19      while (cond!=-1){
20          printf("Enter Employee gross sale (-1 to end) : ");
21          scanf("%d",&temp2);
22          if(temp2==-1){
23              cond=-1;
24              break;
25          }
26          salary = 200 + (9 * temp2)/100;
27          printf("Employee salary is %d\n", salary);
28          for (i=0;i<9;i++){
29              if( temp3+99 > salary && salary >= temp3){
30                  commission_base[i][2] = commission_base[i][2] +1;
31                  printf("commission_base[%d][2]=%d\n",i,commission_base[i][2]);
32                  i=9;
33              }
34              else if (salary >= 1000 ){
35                  i=0;
36                  commission_base[i][2] = commission_base[i][2] +1;
37                  printf("commission_base[%d][2]=%d\n",i,commission_base[i][2]);
38              }
39              temp3 = temp3+100;
40          }
41      }
42      return 0;
43  }

```

```

23     cond=-1;
24     break;
25 }
26 salary = 200 + (5 * temp2)/100;
27 printf("Employee salary is %d\n", salary);
28 for (i=0;i<9;i++){
29     if( temp3+99 > salary && salary >= temp3){
30         commission_base[i][2] = commission_base[i][2] +1;
31         printf("commission_base[%d][2]=%d\n",i,commission_base[i][2]);
32         i=8;
33     }
34     else if (salary >= 1000 ){
35         i=8;
36         commission_base[i][2] = commission_base[i][2] +1;
37         printf("commission_base[%d][2]=%d\n",i,commission_base[i][2]);
38     }
39     temp3 = temp3+100;
40 }
41 temp3=200;
42 }
43 //Setting the final shape and printing it;
44 for (i=0;i<9;i++){
45     {
46         commission_base[i][0] = temp;
47         temp = temp + 99;
48         printf("%d",commission_base[i][0]);
49         if(temp<1000)
50         {
51             commission_base[i][1] = temp;
52             temp = temp + 1;
53             printf("    %d ",commission_base[i][1]);
54         }
55         else
56             printf(" and more ");
57         printf("%d \n",commission_base[i][2]);
58     }
59 }
60 return 0;
61 }
62
63

```

3rd problem:-

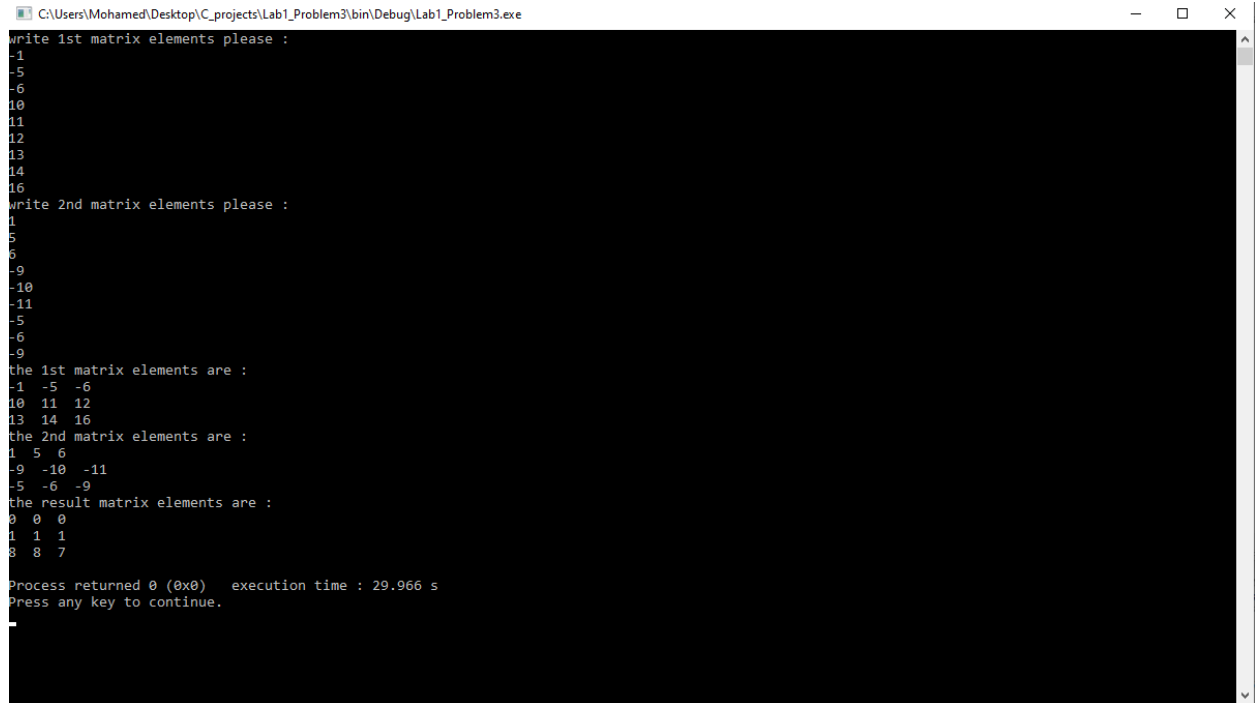
What I did in this project:

- 1- I defined some variables and specially I defined 2 2d arrays of 3*3 elements.
- 2- I made a for loop to scan the first array values by using a pointer to array of pointers.
- 3- I made another for loop to scan the 2nd array values by using another pointer to array of pointers.
- 4- Then by using those 2 pointers I summed every element with its parallel one in the other array and put the result in a 3rd array by using another pointer to array of pointers.

Attempt 1:-

```
C:\Users\Mohamed\Desktop\C_projects\Lab1_Problem3\bin\Debug\Lab1_Problem3.exe
write 1st matrix elements please :
1
2
3
4
5
6
7
8
9
write 2nd matrix elements please :
9
8
7
6
5
4
3
2
1
the 1st matrix elements are :
1 2 3
4 5 6
7 8 9
the 2nd matrix elements are :
9 8 7
6 5 4
3 2 1
the result matrix elements are :
10 10 10
10 10 10
10 10 10
Process returned 0 (0x0)   execution time : 17.513 s
Press any key to continue.
```

Attempt 2:-



```
C:\Users\Mohamed\Desktop\C_projects\Lab1_Problem3\bin\Debug\Lab1_Problem3.exe
write 1st matrix elements please :
-1
-5
-6
10
11
12
13
14
16
write 2nd matrix elements please :
1
5
6
-9
-10
-11
-5
-6
-9
the 1st matrix elements are :
-1 -5 -6
10 11 12
13 14 16
the 2nd matrix elements are :
1 5 6
-9 -10 -11
-5 -6 -9
the result matrix elements are :
0 0 0
1 1 1
8 8 7
Process returned 0 (0x0)   execution time : 29.966 s
Press any key to continue.
```

Code:-

```
#include<stdio.h>

#include<stdlib.h>

int main (){

int m1[3][3] = { {0} };

int *p11[3];

/*

// we need to make this so we made this by using the same for loop instead of making it alone at first

p11[0] = &m1[0][0];

p11[1] = &m1[1][0];

p11[2] = &m1[2][0];
```

```

*/

int m2[3][3],result[3][3];

int *p12[3];

int i,j;

printf("write 1st matrix elements please : \n");

for(i=0;i<3;i++){

    p11[i] = &m1[i][0];

    for(j=0;j<3;j++){

        scanf("%d", (*(p11+i)+j) );

    }

}

printf("write 2nd matrix elements please : \n");

for(i=0;i<3;i++){

    p12[i] = &m2[i][0];

    for(j=0;j<3;j++){

        scanf("%d", (*(p12+i)+j) );

    }

}

for(i=0;i<3;i++){

    p12[i] = &m2[i][0];

    for(j=0;j<3;j++){

        *(*(result+i)+j) = *(*(p11+i)+j) + *(*(p12+i)+j) ;

    }

}

printf("the 1st matrix elements are : \n");

```



```
for(i=0;i<3;i++){  
    for(j=0;j<3;j++){  
        printf("%d ", *(p11+i+j) );  
    }  
    printf("\n");  
}  
  
printf("the 2nd matrix elements are : \n");  
for(i=0;i<3;i++){  
    for(j=0;j<3;j++){  
        printf("%d ", *(p12+i+j) );  
    }  
    printf("\n");  
}  
  
printf("the result matrix elements are : \n");  
for(i=0;i<3;i++){  
    for(j=0;j<3;j++){  
        printf("%d ", *(result+i+j) );  
    }  
    printf("\n");  
}  
  
return 0;  
}
```

Code image:-

```
1  #include<stdio.h>
2  #include<stdlib.h>
3
4  int main () {
5
6      int m1[3][3] = { {0} } ;
7      int *p11[3] ;
8      /*
9      // We need to make this so we made this by using the same for loop instead of making it alone at first
10     p11[0] = &m1[0][0];
11     p11[1] = &m1[1][0];
12     p11[2] = &m1[2][0];
13     */
14
15     int m2[3][3], result[3][3];
16     int *p12[3];
17     int i,j;
18
19     printf("write 1st matrix elements please : \n");
20
21     for(i=0;i<3;i++){
22         p11[i] = &m1[i][0];
23         for(j=0;j<3;j++){
24             scanf("%d", (*(p11+i)+j) );
25         }
26     }
27
28     printf("write 2nd matrix elements please : \n");
29     for(i=0;i<3;i++){
30         p12[i] = &m2[i][0];
31         for(j=0;j<3;j++){
32             scanf("%d", (*(p12+i)+j) );
33         }
34     }
35
36     for(i=0;i<3;i++){
37         p12[i] = &m2[i][0];
38         for(j=0;j<3;j++){
39             (*(result+i)+j) = *(*(p11+i)+j) + *(*(p12+i)+j) ;
40         }
41     }
42 }
```

```
35
36     for(i=0;i<3;i++){
37         p12[i] = &m2[i][0];
38         for(j=0;j<3;j++){
39             (*(result+i)+j) = *(*(p11+i)+j) + *(*(p12+i)+j) ;
40         }
41     }
42
43
44     printf("the 1st matrix elements are : \n");
45     for(i=0;i<3;i++){
46         for(j=0;j<3;j++){
47             printf("%d ", *(p11+i)+j) ;
48         }
49         printf("\n");
50     }
51
52
53     printf("the 2nd matrix elements are : \n");
54     for(i=0;i<3;i++){
55         for(j=0;j<3;j++){
56             printf("%d ", *(p12+i)+j) ;
57         }
58         printf("\n");
59     }
60
61
62     printf("the result matrix elements are : \n");
63     for(i=0;i<3;i++){
64         for(j=0;j<3;j++){
65             printf("%d ", *(result+i)+j) ;
66         }
67         printf("\n");
68     }
69
70
71     return 0;
72 }
73
74
75
```