# EXPERIMENT NUMBER –10

**STUDENT'S NAME – YANA SRIVASTAVA**

**STUDENT'S UID –** 20BCS2279

**CLASS AND GROUP –** 23 B

**SEMESTER –**1st

**AIM OF THE EXPERIMENT –**

## Practical 10.1:
## WAP to store a character string in block of memory space created by malloc and then modify the same to store a large string.

## ALGORITHM

```
STEP 1:- START
STEP 2:- Print RUNTIME MEMORY ALLOCATION
STEP 4:-Enter max number of characters you want to input
STEP 3:- By using char *p = (char*)malloc(n*sizeof(char));
STEP 4:- if(p==NULL)
            Memory allocation fails..
STEP 5:-Now terminate string with NULL character
STEP 6:-Print Enter new size
STEP 7:-If p=(char*)realloc(p,n*sizeof(char));
            Enter new string please
STEP 8:-Display the new string you wrote
STEP 9:- END
```

## PROGRAM CODE

```
#include<stdio.h>

#include<stdlib.h>

int main()

{

int n,i;

printf("\tRUNTIME MEMORY ALLOCATION\n");
printf("Enter max number of characters you want to input\n");

scanf("%d",&n);



char *p = (char*)malloc(n*sizeof(char));

if(p==NULL)

{

printf("Memory allocation fails..");

exit(0);

}

puts("Enter string please");

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

{

scanf("%c",p+i);

}

//Now terminate string with NULL character

*(p+i)= '\0';

printf("You wrote %s",p);
fflush(stdin);


printf("\nEnter new size\n");
scanf("%d",&n);


p=(char*)realloc(p,n*sizeof(char));
puts("\nEnter new string please");
scanf("%d",&n);


for(i=0;i<n;i++)
{
scanf("%c",p+i);
}
//Now terminate string with NULL character

*(p+i)= '\0';
printf("You wrote %s\n",p);
free(p);
return 0;
}
```

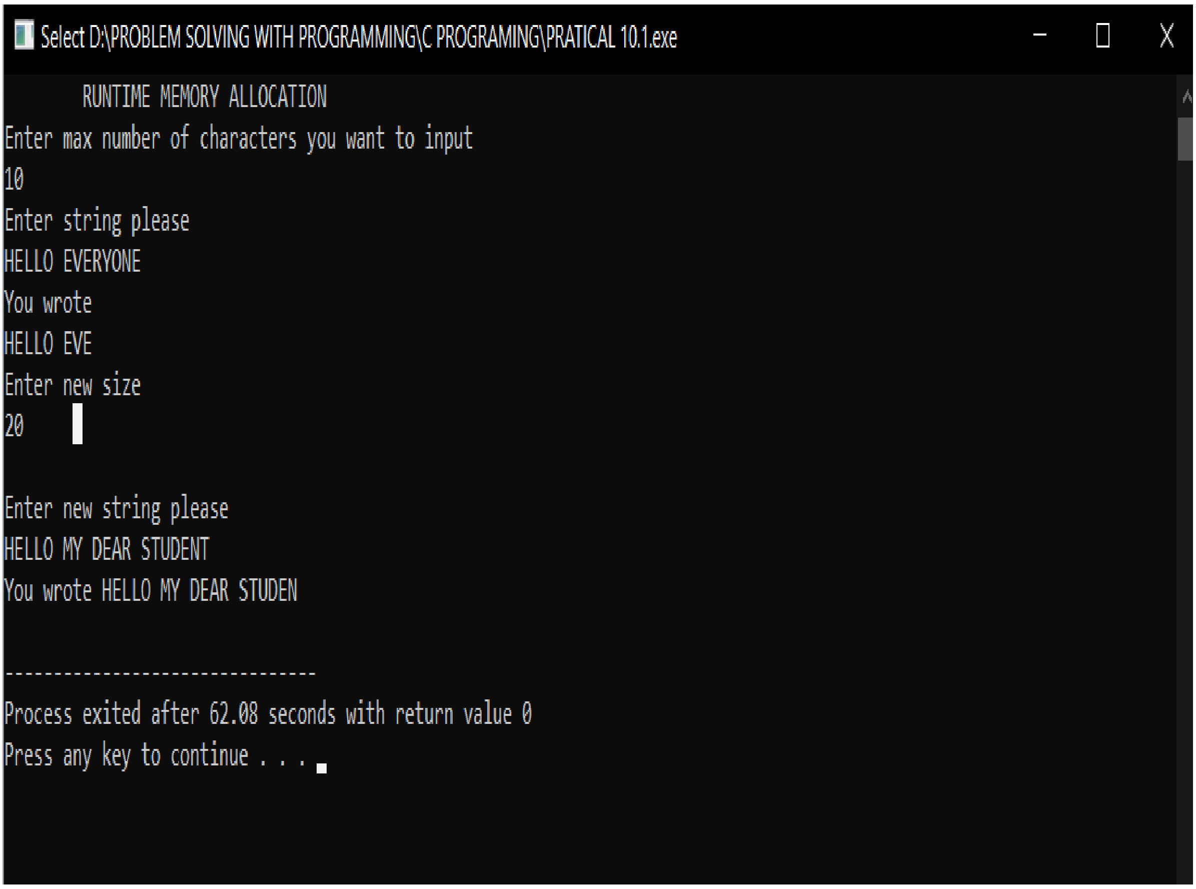
## ERRORS ENCOUNTERED DURING PROGRAM'S EXECUTION

NO ERROR

**PROGRAMS' EXPLANATION (in brief)**

**In this program to store a character string in block of memory space created by malloc and then modify the same to store a large string.**

## OUTPUT



Select D:\PROBLEM SOLVING WITH PROGRAMMING\C PROGRAMING\PRATICAL 10.1.exe

```
        RUNTIME MEMORY ALLOCATION
Enter max number of characters you want to input
10
Enter string please
HELLO EVERYONE
You wrote
HELLO EVE
Enter new size
20

Enter new string please
HELLO MY DEAR STUDENT
You wrote HELLO MY DEAR STUDEN

--------------------------------
Process exited after 62.08 seconds with return value 0
Press any key to continue . . .
```

**AIM OF THE EXPERIMENT –**

### Practical 10.2:

**At the start of your class lecture , n students were present .You declare array dynamically to store roll numbers of these students after 5 Mins m**

**more students join the class now you will reallocate memory space to store n + m roll numbers .Write a program by using functions malloc(), realloc() and free() .**

## ALGORITHM

**STEP 1:- START**

**STEP 2:- Take int *ptr; int n,m,i;**
**STEP 3:-Enter the no. of students**

**STEP 4:-Enter the roll no. ofstudents**

**STEP 5:-Enter the no. of new students entered after 5 mins.**

**STEP 6:-Enter the roll no. of new students**

**STEP 7:-Display the Roll no.of all the students are**

**STEP 8:-END**

## PROGRAM CODE

```
#include  <stdio.h>

#include <stdlib.h>

int main()

{

int *ptr; int n,m,i;

printf("Enter the no. of students :-");

scanf("%d",&n);

ptr=(int*) malloc(n*sizeof(int));


printf("Enter the roll no. of %d students :-\n",n);

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

{
scanf("%d",&ptr[i]);
}


printf("Enter the no. of new students entered after 5 mins. :- ");

scanf("%d",&m);

ptr=(int*)realloc(ptr,(n+m)*sizeof(int));

printf("Enter the roll no. of new students :-\n");

for(i=n;i<n+m;i++)

{

scanf("%d",&ptr[i]);

}


printf("Roll no.of all the students are:-");

for(i=0;i<(n+m);i++)

{

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

}

free(ptr);


return 0;

}
```

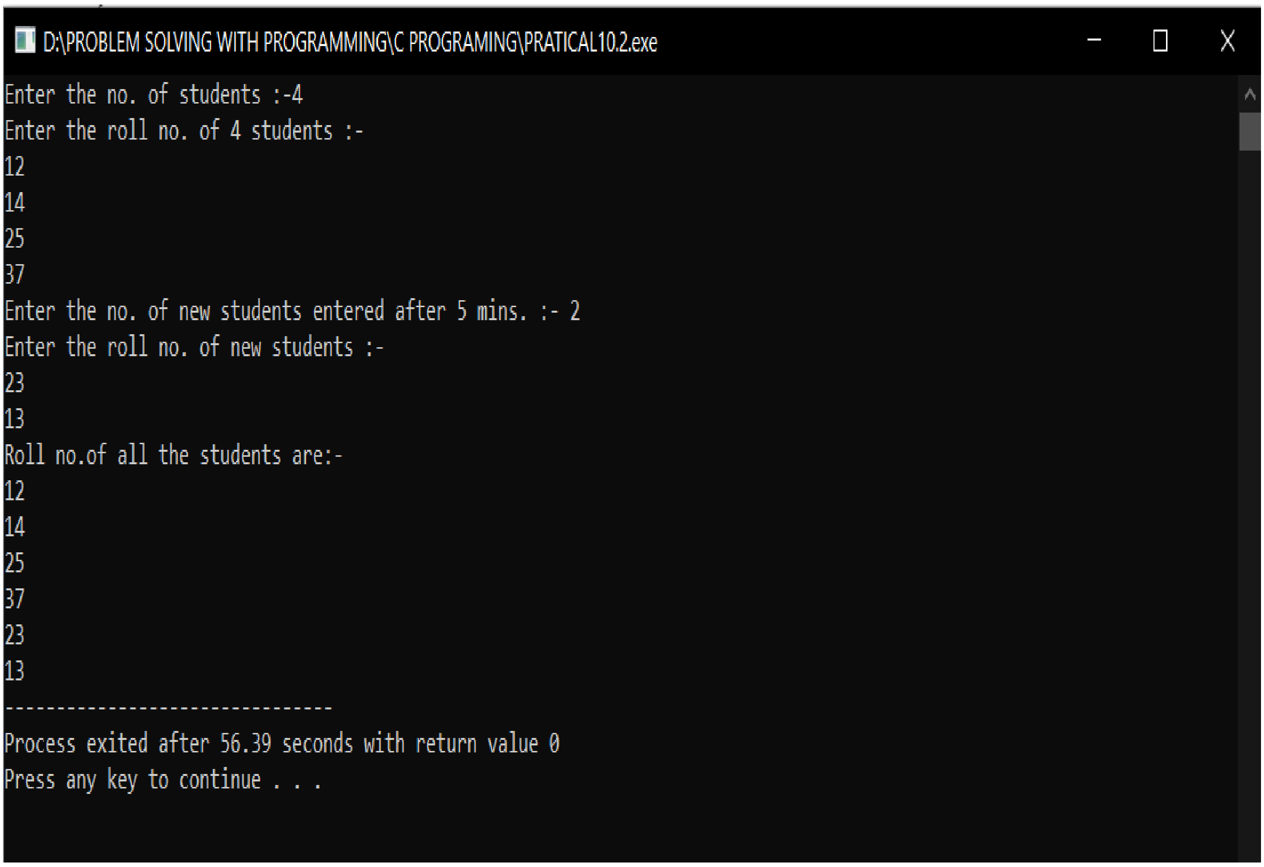
## ERRORS ENCOUNTERED DURING PROGRAM’S EXECUTION

**NO ERROR**

**PROGRAMS’ EXPLANATION (in brief)**

In this program At the start of your class lecture , n students were present .You declare array dynamically to store roll numbers of these students after 5 Mins m more students join the class now you will reallocate memory space to store n + m roll numbers a program by using functions malloc(), realloc() and free()

## OUTPUT



D:\PROBLEM SOLVING WITH PROGRAMMING\C PROGRAMING\PRATICAL10.2.exe

```
Enter the no. of students :-4
Enter the roll no. of 4 students :-
12
14
25
37
Enter the no. of new students entered after 5 mins. :- 2
Enter the roll no. of new students :-
23
13
Roll no.of all the students are:-
12
14
25
37
23
13
--------------------------------
Process exited after 56.39 seconds with return value 0
Press any key to continue . . .
```

**AIM OF THE EXPERIMENT –**

## Practical 10.3:

## Sort the list of N elements where memory is allocated dynamically using pointers.

## ALGORITHM

**STEP 1:- START**

**STEP 2:- Take int *a,n,i,j,t;**

**STEP 3:- How many numbers you want to be sorted**

**STEP 4:- a=(int *)malloc(n *sizeof(int));**

**STEP 5:-EnterNumbers**

**STEP 6:-After Sorting in Ascending Order**

**STEP 7:- END**

## PROGRAM CODE

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
  int *a,n,i,j,t;


  printf("How many numbers you want to be sorted: ");
  scanf("%d",&n);
  a=(int *)malloc(n *sizeof(int));
  printf("\nEnter %d Numbers: \n\n",n);
  for(i=0;i<=n-1;i++)
  {
    scanf("%d", (a+i));
  }
  for(i=0;i<n;i++)
  {
    for(j=0;j<=i;j++)
    {
      if(*(a+i)<*(a+j))
      {
        t=*(a+i);
        *(a+i)=*(a+j);
        *(a+j)=t;
      }
    }
  }
  printf("\nAfter Sorting in Ascending Order: \n");
  for(i=0;i<n;i++)
  printf("\n%d",*(a+i));
  return 0;
}
```

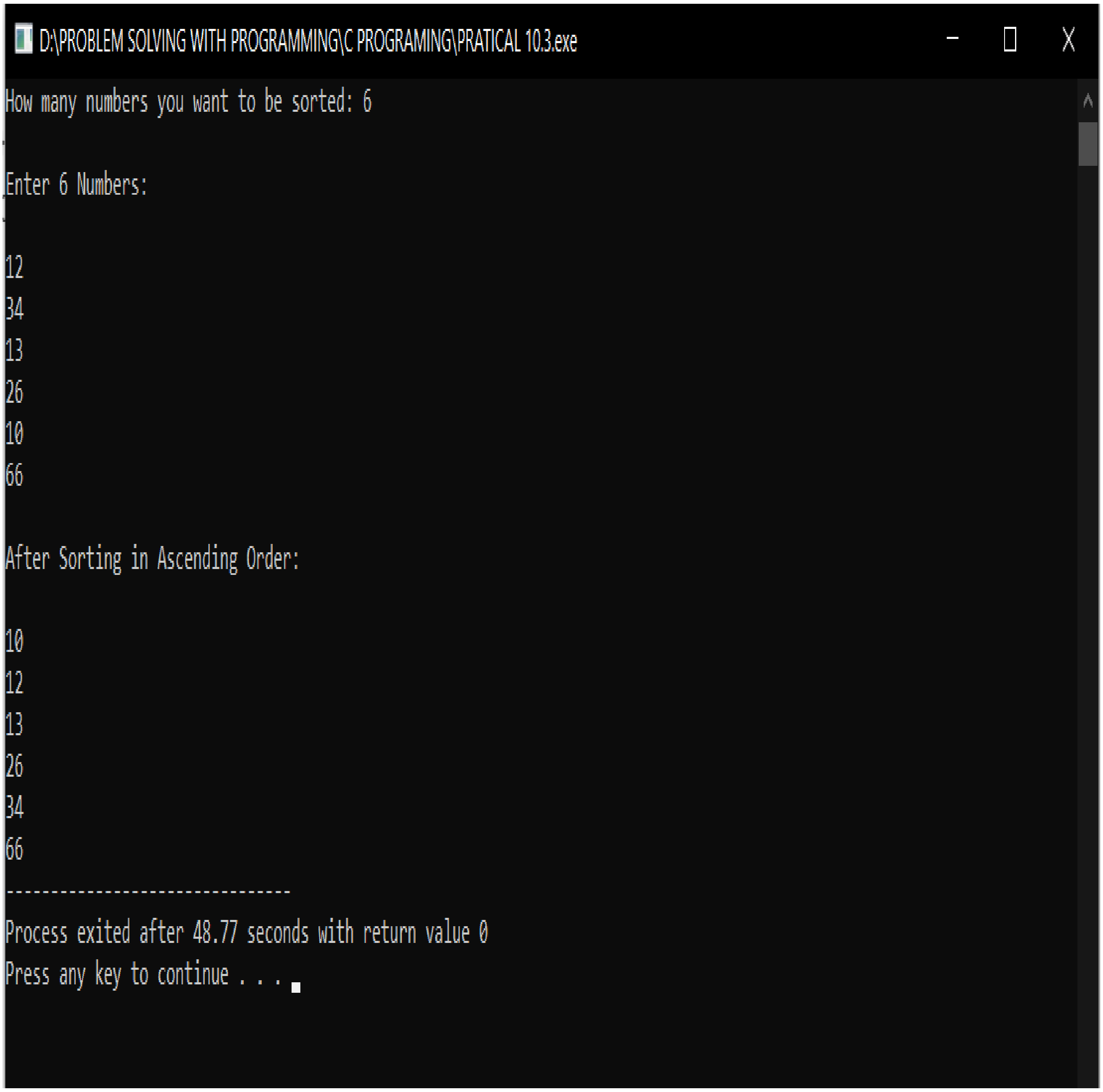
## ERRORS ENCOUNTERED DURING PROGRAM'S EXECUTION

**NO ERROR**

## PROGRAMS' EXPLANATION (in brief)

**In this program to Sort the list of N elements where memory is allocated dynamically using pointers.**

## OUTPUT



D:\PROBLEM SOLVING WITH PROGRAMMING\C PROGRAMING\PRATICAL 10.3.exe

```
How many numbers you want to be sorted: 6

Enter 6 Numbers:

12
34
13
26
10
66

After Sorting in Ascending Order:

10
12
13
26
34
66
--------------------------------
Process exited after 48.77 seconds with return value 0
Press any key to continue . . .
```

AIM OF THE EXPERIMENT –

## Practical 10.4:

## Marks of group A having m students and group B having n students have stored in two dynamically allocated arrays. Find the average aggregate marks of the whole class.

## ALGORITHM

```
STEP 1:- START
STEP 2:-Take int n,m;
STEP 3:-Enter number of students in group A
STEP 4:-Enter number of students in group B
STEP 5:-int *arr1=(int*)malloc(n*sizeof(int));
        int *arr2=(int*)malloc(m*sizeof(int));
STEP 6:-Enter marks for Group A Students
STEP 7:-Enter marks for Group B Students
STEP 8:-avg=sum/(n+m)
STEP  9:-Aggregate  marks
STEP 10:-END
```

# PROGRAM CODE

```
    #include<stdio.h>
    #include <stdlib.h>
    int main()
    {

    int n,m;

    printf("\n enter number of students in group A\n");
    scanf("%d",&n);
    printf("\n enter number of students in group B\n");
    scanf("%d",&m);
    int *arr1=(int*)malloc(n*sizeof(int));
    int *arr2=(int*)malloc(m*sizeof(int));
    printf("\nenter marks for Group A Students\n");
    for(int i=0;i<n;i++)
    {
    scanf("%d",&arr1[i]);

    }
    printf("\n enter marks for Group B Students\n");
    for(int i=0;i<m;i++)
    {
    scanf("%d",&arr2[i]);
    }
float sum=0;

for(inti=0;i<n;i++)

    {
    sum=sum+arr1[i];
    }

    for(int i=0;i<m;i++)

    {
    sum=sum+arr2[i];
    }

    float avg=sum/(n+m);
    printf("\n Aggregate marks=%f",avg);

      return 0;

    }
```

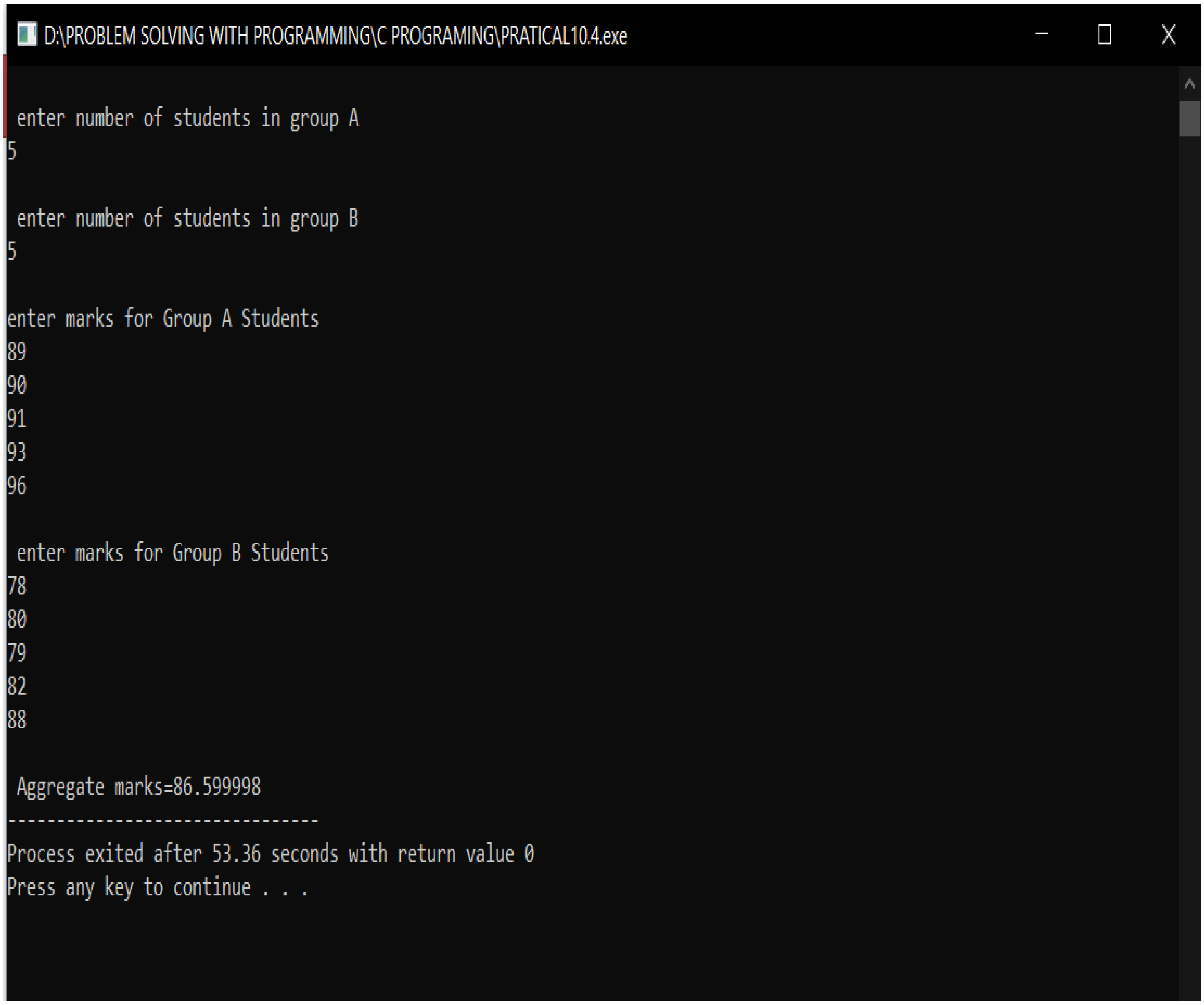
## ERRORS ENCOUNTERED DURING PROGRAM’S EXECUTION

NO ERROR

## PROGRAMS’ EXPLANATION (in brief)

**In this program Marks of group A having m students and group B having n students have stored in two dynamically allocated arrays. Find the average aggregate marks of the whole class.**

# OUTPUT



D:\PROBLEM SOLVING WITH PROGRAMMING\C PROGRAMING\PRATICAL10.4.exe

```
 enter number of students in group A
5

 enter number of students in group B
5

enter marks for Group A Students
89
90
91
93
96

 enter marks for Group B Students
78
80
79
82
88

 Aggregate marks=86.599998
--------------------------------
Process exited after 53.36 seconds with return value 0
Press any key to continue . . .
```

## LEARNING OUTCOMES

- **Identify situations where computational methods would beuseful.**
- **Approach the programming tasks using techniques learnt and writepseudo-code.**
- **Choose the right data representation formats based on the requirements of the problem.**
- **Use the comparisons and limitations of the various programming constructs and choose the right one for the task.**

**EVALUATION COLUMN (To be filled by concerned faculty only)**

| **Sr. No.** | **Parameters** | **Maximum Marks** | **Marks Obtained** |
|---|---|---|---|
| 1. | Worksheet Completion including writing learning objective/ Outcome | 10 | |
| 2. | Post Lab Quiz Result | 5 | |
| 3. | Student engagement in Simulation/ Performance/ Pre Lab Questions | 5 | |
| 4. | Total Marks | 20 | |