

TITLE : C Program to Initialize an Array Dynamically**PROGRAM :**

```
#include <stdio.h>

int main()
{
    int i,b;

    printf("Enter The Size of the Array:");

    scanf("%d",&b);

    int a[b];

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

        printf("Enter Element-%d:",i+1);

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

    }

    printf("The Initialized Array:\n{ ");

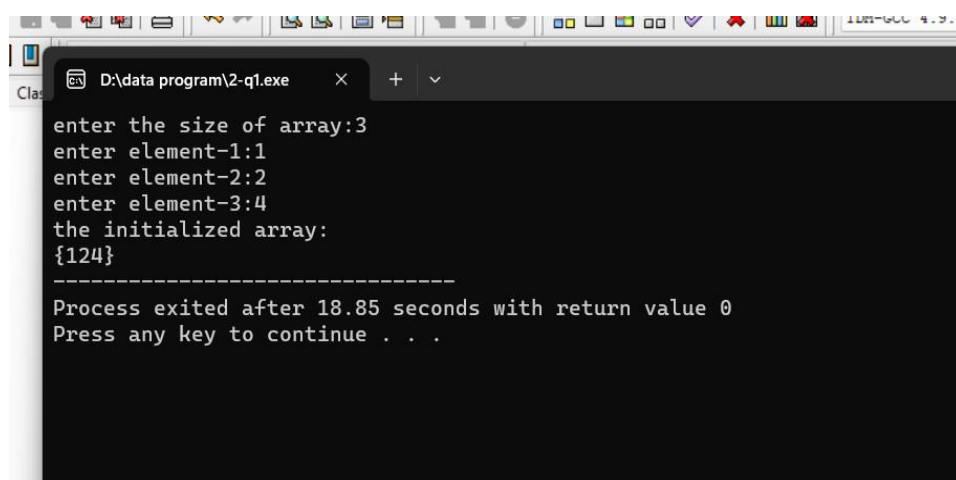
    for(i=0;i<b;i++)
    {

        printf("%d ",a[i]);

    }

    printf("}");

}
```

INPUT AND OUTPUT :

```
D:\data program\2-q1.exe
enter the size of array:3
enter element-1:1
enter element-2:2
enter element-3:4
the initialized array:
{124}
-----
Process exited after 18.85 seconds with return value 0
Press any key to continue . . .
```

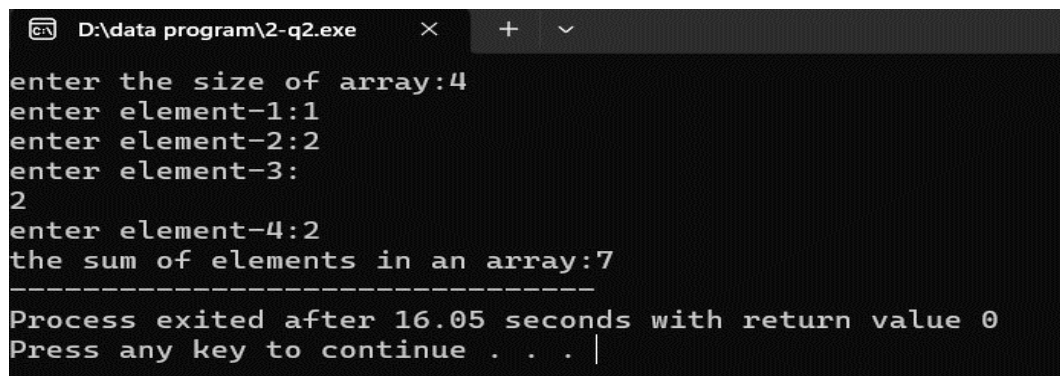
RESULT :

The C Program for Initializing an Array Dynamically is Compiled and Executed Using Dev-C++ and the Output is Verified.

TITLE : C Program to Find the Sum of Elements in the Given Array**PROGRAM :**

```
#include <stdio.h>

int main()
{
    int a[100],i,b,sum=0;
    printf("Enter The Size of the Array:");
    scanf("%d",&b);
    for(i=0;i<b;i++)
    {
        printf("Enter Element-%d:",i+1);
        scanf("%d",&a[i]);
    }
    for(i=0;i<b;i++)
    {
        sum=sum+a[i];
    }
    printf("The Sum of Elements in an Array
is:%d",sum);
}
```

INPUT AND OUTPUT :

```
D:\data program\2-q2.exe
enter the size of array:4
enter element-1:1
enter element-2:2
enter element-3:
2
enter element-4:2
the sum of elements in an array:7
-----
Process exited after 16.05 seconds with return value 0
Press any key to continue . . . |
```

RESULT :

The C Program for Finding the sum of Elements in an Array is Compiled and Executed Using Dev-C++ and the Output is Verified.

TITLE : C Program to Find the Sum of Even and Odd Elements in the Given Array.

PROGRAM :

```
#include <stdio.h>

int main()
{
    int a[100],i,b,esum=0,osum=0;

    printf("Enter The Size of the Array:");

    scanf("%d",&b);

    for(i=0;i<b;i++)
    {
        printf("Enter Element-%d:",i+1);

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

    }

    for(i=0;i<b;i++)
    {
        if(a[i]%2==0)
        {
            esum=esum+a[i];

        }

        else
        {
            osum+=a[i];

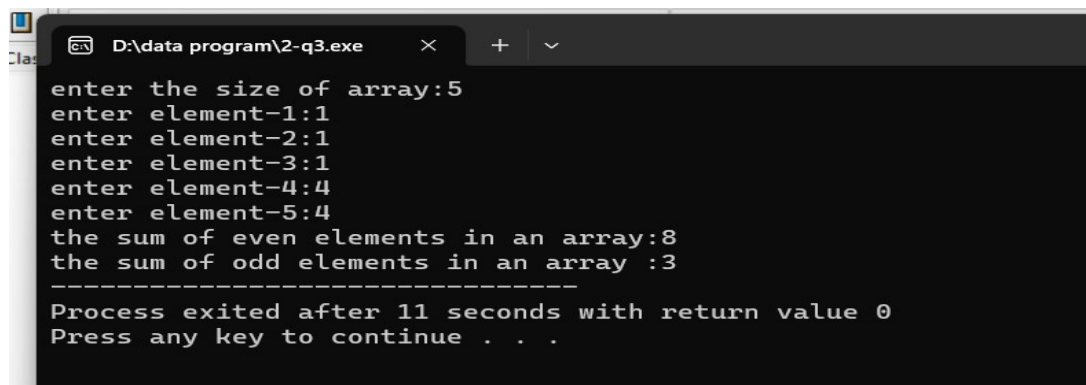
        }

    }

    printf("The Sum of Even Elements in an Array is:%d\n",esum);

    printf("The Sum of Odd Elements in an Array is:%d",osum);}
```

INPUT AND OUTPUT :



```
D:\data program\2-q3.exe
enter the size of array:5
enter element-1:1
enter element-2:1
enter element-3:1
enter element-4:4
enter element-5:4
the sum of even elements in an array:8
the sum of odd elements in an array :3
-----
Process exited after 11 seconds with return value 0
Press any key to continue . . .
```

RESULT :

The C Program for Finding the sum of Odd and Even Elements in an Array is Compiled and Executed Using Dev-C++ and the Output is Verified.

TITLE : C Program to Perform Insertion and Deletion in an Array**PROGRAM :**

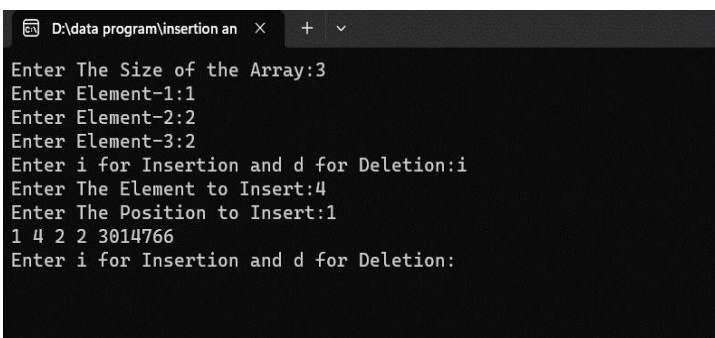
```
#include <stdio.h>

#include <string.h>

int main()
{
    int a[100],i,j,t1,t2,b,c,e;
    char o[2];
    printf("Enter The Size of the Array:");
    scanf("%d",&b);
    for(i=0;i<b;i++)
    {
        printf("Enter Element-%d:",i+1);
        scanf("%d",&a[i]);
    }
    while(true)
    {
        printf("Enter i for Insertion and d for Deletion:");
        scanf("%s",o);
        if(strcmp(o,"i")==0){
            printf("Enter The Element to Insert:");
            scanf("%d",&e);
            printf("Enter The Position to Insert:");
            scanf("%d",&c);
            t1=a[c];
            a[c]=e;
```

```
b++;

            for(i=c+1;i<b;i++)
            {
                t2=a[i];
                a[i]=t1;
                t1=t2;
            }
            for(i=0;i<b;i++)
            {
                printf("%d ",a[i]);
            }
            printf("\n");
        }
        else if(strcmp(o,"d")==0)
        {
            printf("Enter the Element to be
Deleted:");
            scanf("%d",&e);
            for(i=0;i<b;i++)
            {
                if(a[i]==e)
                {
                    a[j]=a[j+1];
                }
                b--;
            }
        }
        for(i=0;i<b;i++)
        {
            printf("%d ",a[i]);
        }
        printf("\n");
    }
}
```

INPUT AND OUTPUT :

```
D:\data program\insertion an
Enter The Size of the Array:3
Enter Element-1:1
Enter Element-2:2
Enter Element-3:2
Enter i for Insertion and d for Deletion:i
Enter The Element to Insert:4
Enter The Position to Insert:1
1 4 2 2 3014766
Enter i for Insertion and d for Deletion:
```

RESULT :

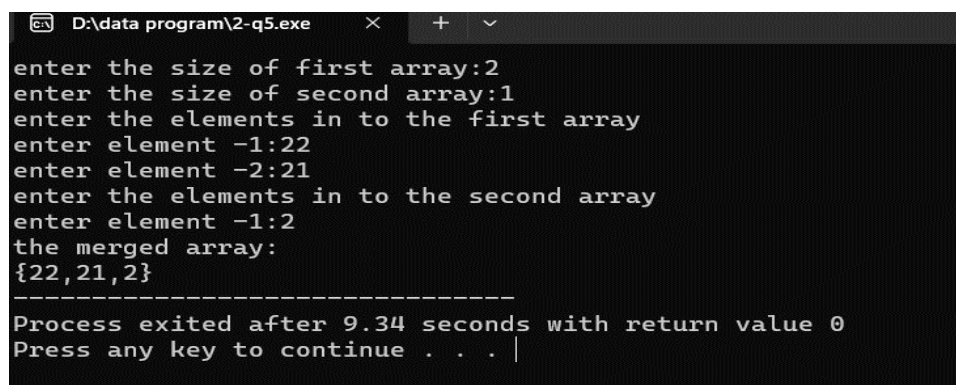
The C Program for Performing Insertion and Deletion in an Array is Compiled and Executed Using Dev-C++ and the Output is Verified.

TITLE : C Program to Merge Two Arrays.**PROGRAM :**

```
#include <stdio.h>

int main()
{
    int a[100],b[100],i,c,d,j=0;
    printf("Enter The Size of First Array:");
    scanf("%d",&c);
    printf("Enter The Size of Second Array:");
    scanf("%d",&d);
    printf("Enter The Elements into the 1st Array\n");
    for(i=0;i<c;i++)
    {
        printf("Enter Element-%d:",i+1);
        scanf("%d",&a[i]);
    }
    printf("Enter The Elements into the 2nd Array\n");
    for(i=0;i<d;i++)
    {
        printf("Enter Element-%d:",i+1);
        scanf("%d",&b[i]);
    }
}
```

```
for(i=c;i<c+d;i++)
{
    a[i]=b[j];
    j++;
}
printf("The Merged Array:\n{");
for(i=0;i<c+d;i++)
{
    if(i<c+d-1)
    {
        printf("%d,",a[i]);
    }
    else
    {
        printf("%d}",a[i]);
    }
}
}
```

INPUT AND OUTPUT :

```
D:\data program\2-q5.exe
enter the size of first array:2
enter the size of second array:1
enter the elements in to the first array
enter element -1:22
enter element -2:21
enter the elements in to the second array
enter element -1:2
the merged array:
{22,21,2}
-----
Process exited after 9.34 seconds with return value 0
Press any key to continue . . . |
```

RESULT :

The C Program for Merging Two Arrays is Compiled and Executed Using Dev-C++ and the Output is Verified.

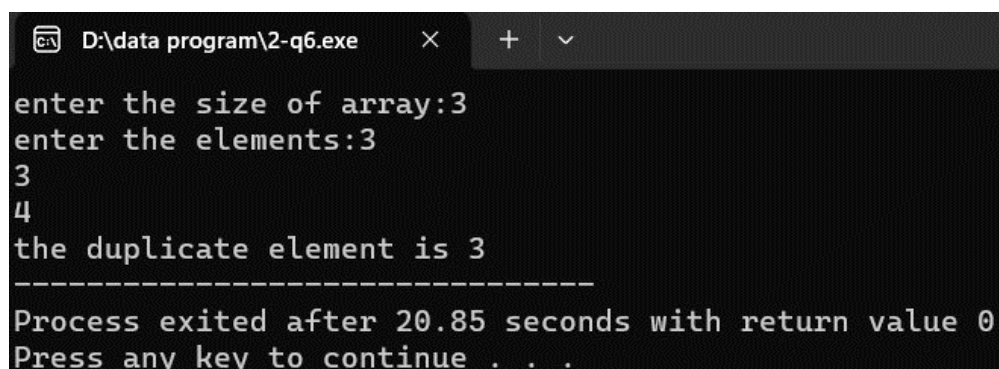
TITLE : C Program to Find Duplicate Value in an Array**PROGRAM :**

```
#include <stdio.h>

int main()
{
    int a[100],b,i,c,d=0,j[100];
    printf("Enter The Size of the Array:");
    scanf("%d",&b);
    for(i=0;i<b;i++)
    {
        printf("Enter Element-%d:",i+1);
        scanf("%d",&a[i]);
    }
    printf("Enter The Search its Duplicate:");
    scanf("%d",&c);
    for(i=0;i<b;i++)
    {
        if(c==a[i])
        {
            j[d]=i;
            d++;
        }
    }
}
```

```
        printf("The %d Duplicate(s) are Present in
Index Positions : ",d-1);

        for(i=1;i<d;i++)
        {
            printf("%d ",j[i]);
        }
    }
```

INPUT AND OUTPUT :

```
D:\data program\2-q6.exe  X  +  v
enter the size of array:3
enter the elements:3
3
4
the duplicate element is 3
-----
Process exited after 20.85 seconds with return value 0
Press any key to continue . . .
```

RESULT :

The C Program for finding the duplicate values is Compiled and Executed Using Dev-C++ and the Output is Verified.

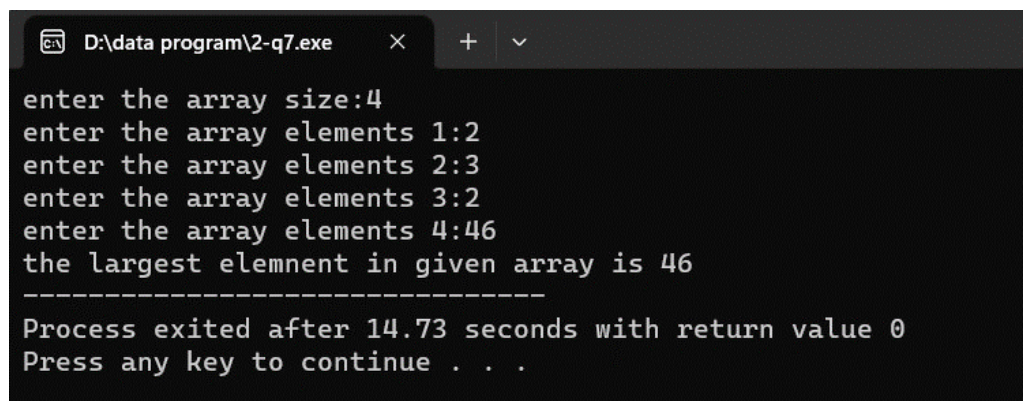
TITLE : C Program to Find Largest Element in an Array.

PROGRAM :

```
#include <stdio.h>

int main()
{
    int a[100],b,max=0,i;
    printf("Enter The Size of the Array:");
    scanf("%d",&b);
    for(i=0;i<b;i++)
    {
        printf("Enter Element-%d:",i+1);
        scanf("%d",&a[i]);
    }
    max=0;
    for(i=0;i<b;i++)
    {
        if(a[i]>max)
        {
            max=a[i];
        }
    }
    printf("The Largest Element in the array is %d",max);
}
```

INPUT AND OUTPUT :



```
D:\data program\2-q7.exe  ×  +  ∨
enter the array size:4
enter the array elements 1:2
enter the array elements 2:3
enter the array elements 3:2
enter the array elements 4:46
the largest elemnent in given array is 46
-----
Process exited after 14.73 seconds with return value 0
Press any key to continue . . .
```

RESULT :

The C Program for finding the largest Number in an Array is Compiled and Executed Using Dev-C++ and the Output is Verified.

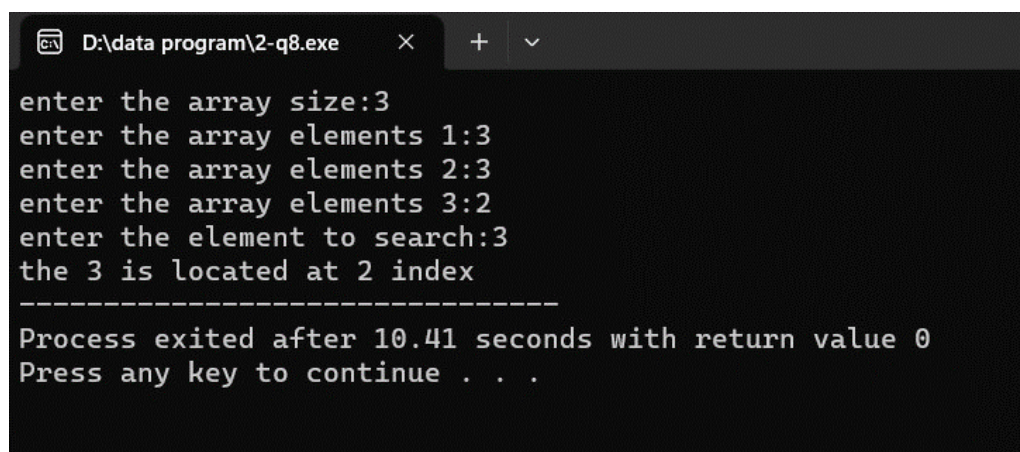
TITLE : C Program to Search an Element in an array using linear search.

PROGRAM :

```
#include <stdio.h>

int main()
{
    int a[100],b,c,j=0,i;
    printf("Enter The Size of the Array:");
    scanf("%d",&b);
    for(i=0;i<b;i++)
    {
        printf("Enter Element-%d:",i+1);
        scanf("%d",&a[i]);
    }
    printf("Enter The Element to be Searched:");
    scanf("%d",&c);
    for(i=0;i<b;i++)
    {
        if(a[i]==c)
        {
            printf("Element found at %d index",i);
            j++;
            break;
        }
    }
    if(j==0)
    {
        printf("Element Not found");
    }
}
```

INPUT AND OUTPUT :



```
D:\data program\2-q8.exe  X  +  v
enter the array size:3
enter the array elements 1:3
enter the array elements 2:3
enter the array elements 3:2
enter the element to search:3
the 3 is located at 2 index
-----
Process exited after 10.41 seconds with return value 0
Press any key to continue . . .
```

RESULT : The C Program for finding an element in an Array Using Linear Search is Compiled and Executed Using Dev-C++ and the Output is Verified.

TITLE : C Program to Search an Element in an array using Binary search.

PROGRAM :

```
#include <stdio.h>

int main()
{
    int a[100],l,m,h,b,e,f=0,i;

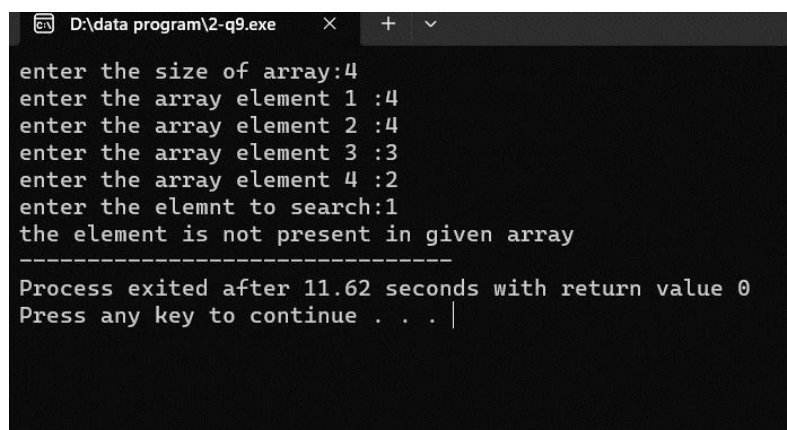
    printf("Enter The Size of the Array:");
    scanf("%d",&b);
    for(i=0;i<b;i++)
    {
        printf("Enter Element-%d:",i+1);
        scanf("%d",&a[i]);
    }

    printf("Enter The Element to be Searched For:");
    scanf("%d",&e);

    l=0;
    h=l-1;
    while(f!=1)
    {
        m=(l+h)/2;
        if(a[m]==e)
        {
            f=1;

            printf("Element Found at %d index",m);
            break;
        }
        else if(e>a[m])
        {
            l=m;
        }
        else
        {
            h=m;
        }
    }
}
```

INPUT AND OUTPUT :



```
D:\data program\2-q9.exe  X  +  v
enter the size of array:4
enter the array element 1 :4
enter the array element 2 :4
enter the array element 3 :3
enter the array element 4 :2
enter the elemnt to search:1
the element is not present in given array
-----
Process exited after 11.62 seconds with return value 0
Press any key to continue . . . |
```

RESULT :

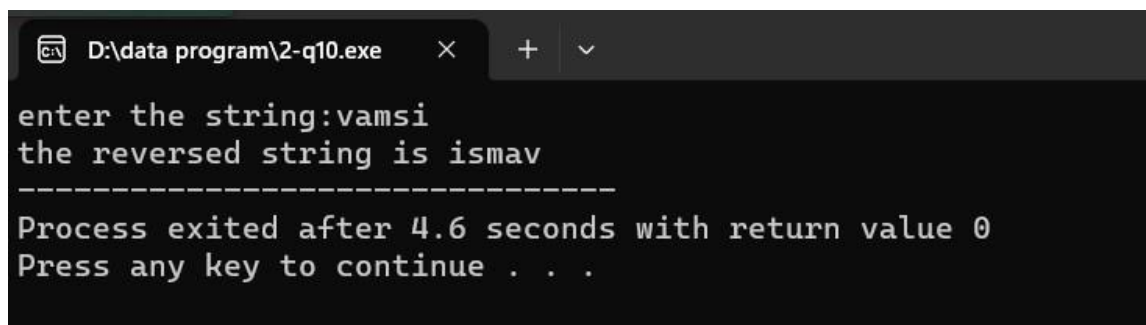
The C Program for finding an element in an Array Using Binary Search is Compiled and Executed Using Dev-C++ and the Output is Verified.

TITLE : C Program to Reverse a Given String.

PROGRAM :

```
#include <stdio.h>
#include <string.h>
int main()
{
    char a[100],b[100];
    int i,j=0;
    printf("Enter The String:");
    scanf("%s",a);
    for(i=strlen(a)-1;i>=0;i--)
    {
        b[j]=a[i];
        j++;
    }
    printf("Reversed String:%s",b);
}
```

INPUT AND OUTPUT :



```
enter the string:vamsi
the reversed string is ismav
-----
Process exited after 4.6 seconds with return value 0
Press any key to continue . . .
```

RESULT :

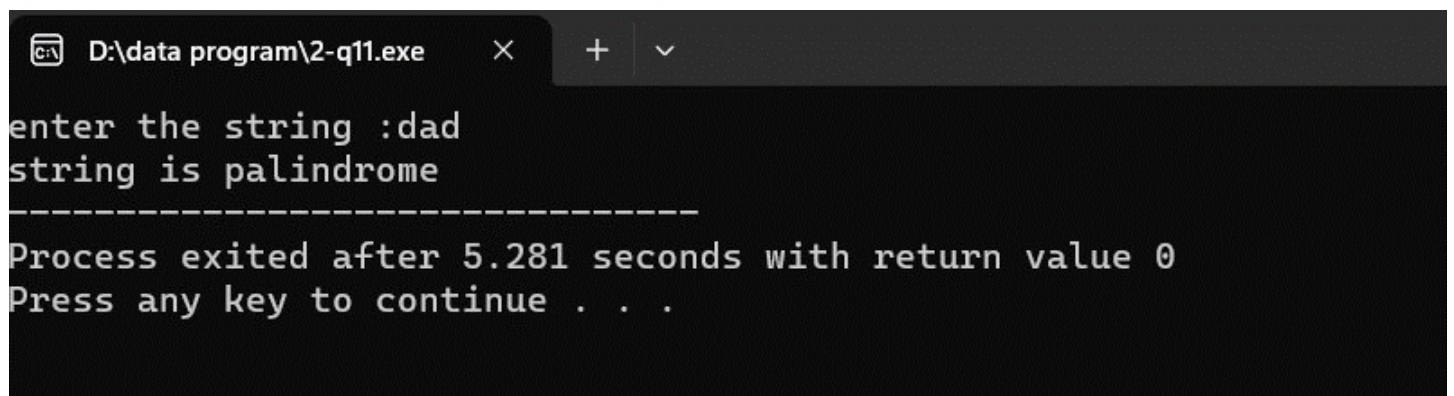
The C Program for Reversing a Given String is Compiled and Executed Using Dev-C++ and the Output is Verified.

TITLE : C Program to Check The Given String is Palindrome or Not.

PROGRAM :

```
#include <stdio.h>
#include <string.h>
int main()
{
    char a[100],b[100];
    int i,j=0;
    printf("Enter The String:");
    scanf("%s",a);
    for(i=strlen(a)-1;i>=0;i--)
    {
        b[j]=a[i];
        j++;
    }
    if(strcmp(a,b)==0)
    {
        printf("The String is a Palindrome");
    }
    else
    {
        printf("The String is not a Palindrome");
    }
}
```

INPUT AND OUTPUT :



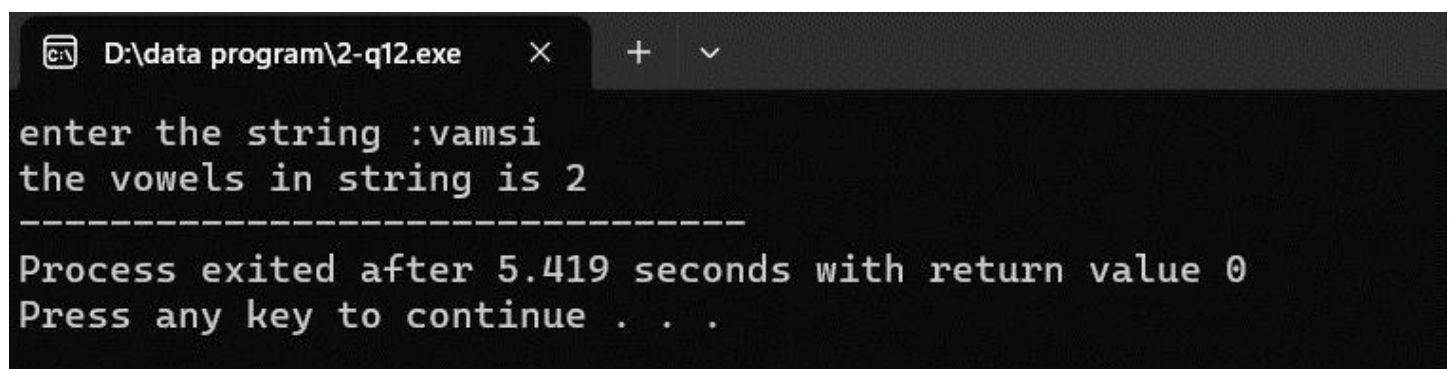
```
C:\> D:\data program\2-q11.exe
enter the string :dad
string is palindrome
-----
Process exited after 5.281 seconds with return value 0
Press any key to continue . . .
```

RESULT :

The C Program for Checking a Given String is Palindrome or not is Compiled and Executed Using Dev-C++ and the Output is Verified.

TITLE : C Program to Check and Count the Vowels in a Given String.**PROGRAM :**

```
#include <stdio.h>
#include <string.h>
int main()
{
    char a[100];
    int i,ac=0,ec=0,ic=0,oc=0,uc=0;
    printf("Enter The String:");
    scanf("%s",a);
    for(i=0;i<strlen(a);i++)
    {
        if(a[i]=='a' || a[i]=='A')
        {
            ac+=1;
        }
        else if(a[i]=='e' || a[i]=='E')
        {
            ec+=1;
        }
        else if(a[i]=='i' || a[i]=='I')
        {
            ic+=1;
        }
        else if(a[i]=='o' || a[i]=='O')
        {
            oc+=1;
        }
        else if(a[i]=='u' || a[i]=='U')
        {
            uc+=1;
        }
    }
    if(ac==0 && ec==0 && ic==0 && oc==0 && uc==0)
    {
        printf("There are No vowels in the given string");
    }
    else
    {
        printf("There are %d vowel(s) in the given string\n",ac+ec+ic+oc+uc);
        printf("A Count:%d\n",ac);
        printf("E Count:%d\n",ec);
        printf("I Count:%d\n",ic);
        printf("O Count:%d\n",oc);
        printf("U Count:%d",uc);
    }
}
```

INPUT AND OUTPUT :

```
D:\data program\2-q12.exe
enter the string :vamsi
the vowels in string is 2
-----
Process exited after 5.419 seconds with return value 0
Press any key to continue . . .
```

RESULT :

The C Program for Counting and Printing the Number of Vowels is Compiled and Executed Using Dev-C++ and the Output is Verified.

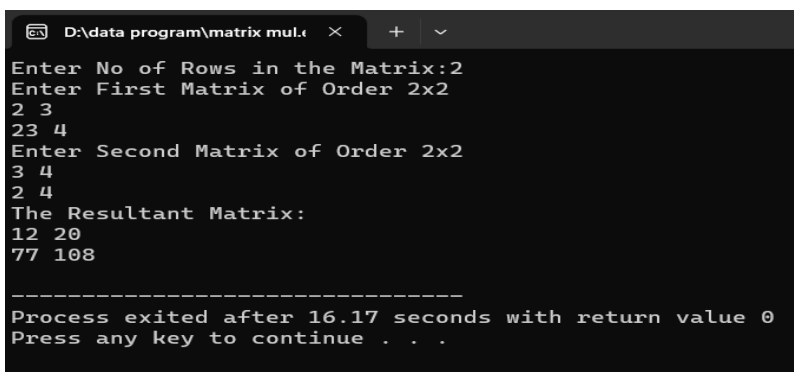
TITLE : C Program for Matrix Multiplication.**PROGRAM :**

```
#include <stdio.h>

int main()
{
    int a[10][10],b[10][10],c[10][10];
    int i,j,k,l;
    printf("Enter No of Rows in the Matrix:");
    scanf("%d",&l);
    printf("Enter First Matrix of Order %dx%d\n",l,l);
    for(i=0;i<l;i++)
    {
        for(j=0;j<l;j++)
        {
            scanf("%d",&a[i][j]);
        }
    }
    printf("Enter Second Matrix of Order %dx%d\n",l,l);
    for(i=0;i<l;i++)
    {
        for(j=0;j<l;j++)
        {
            scanf("%d",&b[i][j]);
        }
    }
}
```

```
printf("The Resultant Matrix:\n");

for(i=0;i<l;i++)
{
    for(j=0;j<l;j++)
    {
        c[i][j]=0;
        for(k=0;k<l;k++)
        {
            c[i][j]+=a[i][k]*b[k][j];
        }
    }
    for(i=0;i<l;i++)
    {
        for(j=0;j<l;j++)
        {
            printf("%d ",c[i][j]);
        }
        printf("\n");
    }
}
```

INPUT AND OUTPUT :

```
D:\data program\matrix mul.t
Enter No of Rows in the Matrix:2
Enter First Matrix of Order 2x2
2 3
23 4
Enter Second Matrix of Order 2x2
3 4
2 4
The Resultant Matrix:
12 20
77 108

-----
Process exited after 16.17 seconds with return value 0
Press any key to continue . . .
```

RESULT :

The C Program for performing matrix multiplication is Compiled and Executed Using Dev-C++ and the Output is Verified.