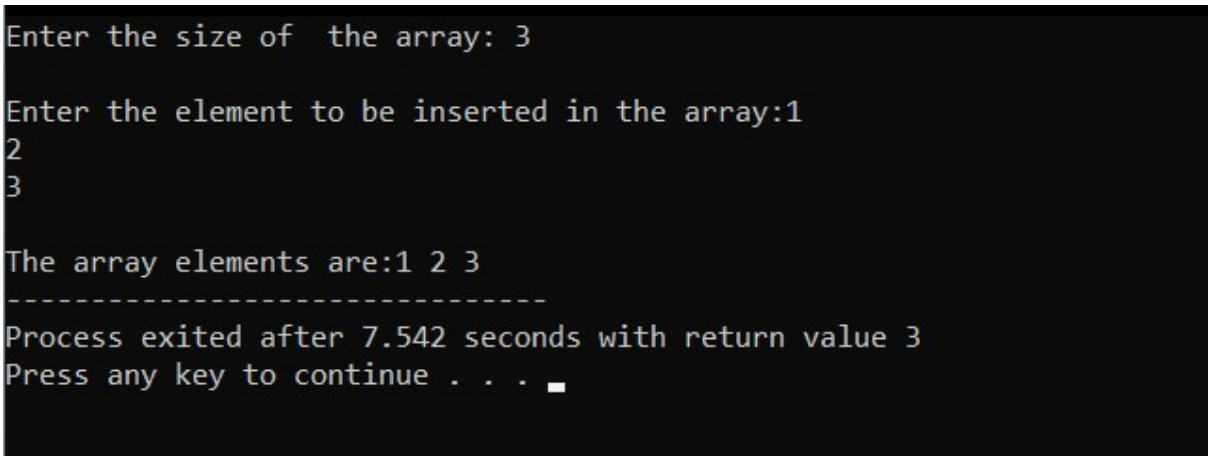
26/08/2023

PROGRAM1: To initialize array dynamically

Code:

```
#include<stdio.h>
int main()
{
	int n;
	int arr[n];
	int i;
	printf("Enter the size of array: ");
	scanf("%d",&n);
	printf("\nEnter the element to be inserted in the array:");
	for(i=0;i<n;i++){
		scanf("%d",&arr[i]);
	}
	printf("\nThe array elements are:");
	for(i=0;i<n;i++){
		printf("%d ",arr[i]);
	}
}
```

Sample input & output:



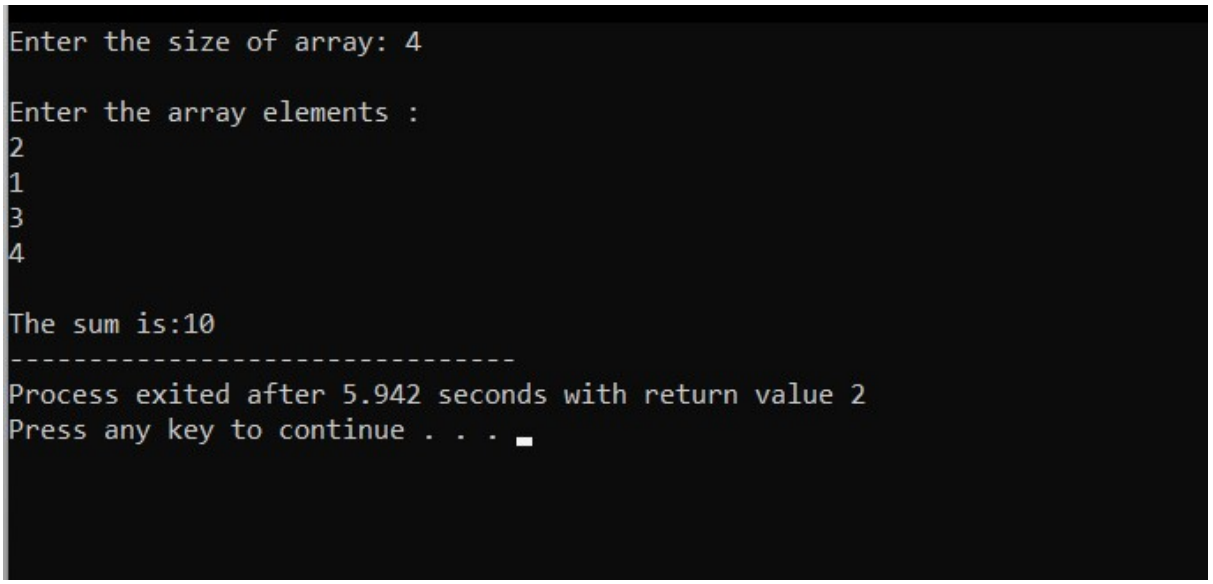
PROGRAM2: To sum of elements in array

Code:

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

```
{
	int n,sum=0;
	int arr[n];
	int i;
	printf("Enter the size of array: ");
	scanf("%d",&n);
	printf("\nEnter the element to be inserted in the array:");
	for(i=0;i<n;i++){
		scanf("%d",&arr[i]);
	}
	printf("\nThe sum of array elements is:");
	for(i=0;i<n;i++){
		sum=sum+arr[i];
	}
	printf("%d",sum);
}
```

Sample input & output:



PROGRAM3: To sum of even & odd elements in array

Code:

```
#include<stdio.h>
int main()
{
	int n,even=0,odd=0;
	int arr[n];
	int i;
	printf("Enter the size of array: ");
```

```
        scanf("%d",&n);
        printf("\nEnter the element to be inserted in the array:");
        for(i=0;i<n;i++){
                scanf("%d",&arr[i]);
        }
        for(i=0;i<n;i++){
                if (arr[i]%2==0){
                        even=even+arr[i];
                }
                else{
                        odd=odd+arr[i];
                }

        }
        printf("\nThe sum of even numbers in the array is %d",even);
        printf("\nThe sum of odd numbers in the array is %d",odd);
}
```

Sample input & output:

```
Enter the number of elements: 5
Enter 5 elements:
12
23
45
23
23
Sum of even elements: 12
Sum of odd elements: 114
```

PROGRAM4: To insert new element in array

Code:

```
#include <stdio.h>
int main()
{
   int array[50], pos, i, n, value;
   printf("Enter the size of the array :\n");
   scanf("%d", &n);
   printf("Enter %d elements:\n", n);
   for (i = 0; i < n; i++)
      scanf("%d", &array[i]);
   printf("Location to insert the new element:\n");
```

```
    scanf("%d", &pos);
    printf("Enter the new element:\n");
    scanf("%d", &value);
    for (i = n - 1; i >= pos - 1; i--)
        array[i+1] = array[i];
    array[pos-1] = value;
    printf("The new array:\n");
    for (i = 0; i <= n; i++)
        printf("%d\n", array[i]);
    return 0;
}
```

Sample input & output:

```
Enter the sixe of the array:
4
Enter 4 elements:
12
23
34
12
Location to insert the new element:
0
Enter the new element:
10
The new array:
0
12
23
34
12
```

PROGRAM5: To delete an element in array

Code:

```
#include <stdio.h>
int main()
{
    int array[100], pos, i, n;
    printf("Enter the size of the array :");
    scanf("%d", &n);
    printf("Enter %d elements:", n);
    for ( i = 0 ; i < n ; i++ )
            scanf("%d", &array[i]);
    printf("Enter the location to delete element:");
```

```
    scanf("%d", &pos);
    if ( pos >= n+1 )
    printf("Deletion not possible.\n");
    else
    {
        for ( i = pos - 1 ; i < n - 1 ; i++ )
                    array[i] = array[i+1];
        printf("The new array is:\n");

        for( i = 0 ; i < n - 1 ; i++ )
                    printf("%d ", array[i]);
    }
    return 0;
}
```

Sample input & output:

```
Enter the size of the array :5
Enter 5 elements:13
23
34
45
34
Enter the location to delete element:1
The new array is:
23 34 45 34
--------------------------------
Process exited after 14.06 seconds with return value 0
Press any key to continue . . .
```

PROGRAM 6: To merge two arrays

Code:

```
#include <stdio.h>
int main()
{
    int n1,n2,n3;
    int a[100], b[100], c[100];
    printf("Enter the size of first array: ");
    scanf("%d",&n1);
    printf("Enter the array elements: ");
    for(int i = 0; i < n1; i++)
```

```
    scanf("%d", &a[i]);
  printf("Enter the size of second array: ");
     scanf("%d",&n2);
  printf("Enter the array elements: ");
  for(int i = 0; i < n2; i++)
    scanf("%d", &b[i]);
  n3 = n1 + n2;
  for(int i = 0; i < n1; i++)
    c[i] = a[i];
  for(int i = 0; i < n2; i++)
     c[i + n1] = b[i];

  printf("The merged array: ");
  for(int i = 0; i < n3; i++)
     printf("%d ", c[i]);
}
```

Sample input & output:

```
Enter Array 1 Size: 5
Enter Array 1 Elements: 1
2
3
4
5

Enter Array 2 Size: 5
Enter Array 2 Elements: 9
8
7
6
5

The new array after merging is:
1 2 3 4 5 9 8 7 6 5
```

PROGRAM7: To find duplicate elements in array

Code:

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

```
    int n,i,j;
        int arr[n];
    printf("Enter the size of array: ");
    scanf("%d",&n);
    printf("\nEnter the elements: ");
    for(i=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
        }
    for(i=0;i<n;i++)
    {
       for(j=i;j<n-1;j++)
       {
          if(arr[i]==arr[j+1])
          {
             printf("Number %d has duplicate values\n",arr[i]);
          }
       }
    }
    return 0;
}
```

Sample input & output:

```
Enter the size of array: 5

Enter the elements: 10
2
2
10
5
Number 10 has duplicate values
Number 2 has duplicate values

--------------------------------
Process exited after 8.288 seconds with return value 0
Press any key to continue . . .
```

PROGRAM8: To search an element using linear search in array

Code:

```c
#include <stdio.h>
int main()
{
  int array[100], search, i, n;
  printf("Enter the size of the array:");
  scanf("%d", &n);
  printf("Enter the elements: ", n);
  for (i = 0; i < n; i++)
    scanf("%d", &array[i]);
  printf("Enter a number to search:");
  scanf("%d", &search);
  for (i = 0; i < n; i++)
  {
    if (array[i] == search)
    {
      printf("%d is present at location %d.\n", search, i+1);
      break;
    }
  }
  if (i == n)
    printf("%d isn't present in the array.\n", search);
  return 0;
}
```

Sample input & output:

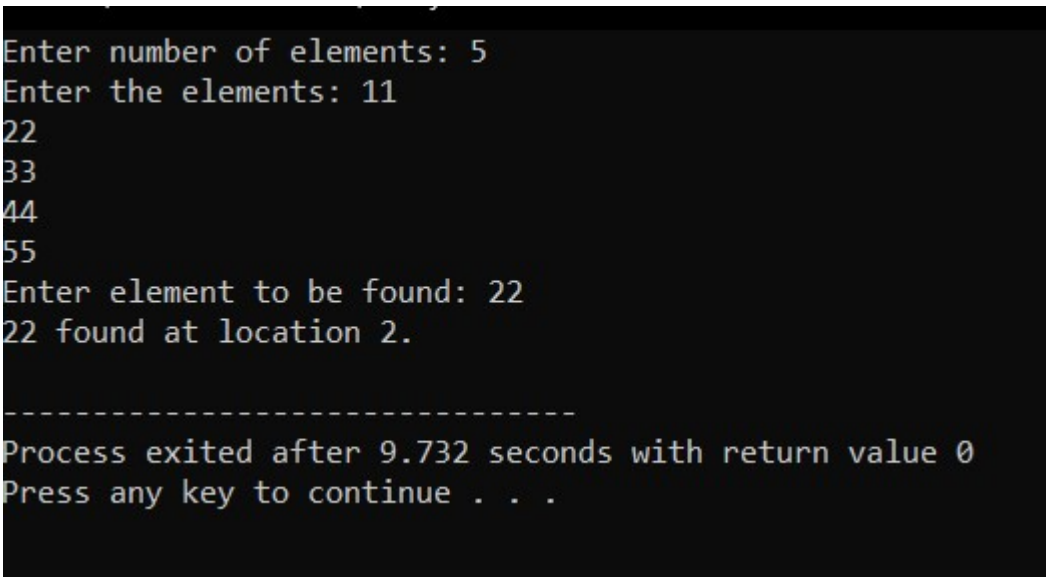
```
Enter the size of the array:5
Enter the elements: 12
23
34
45
2
Enter a number to search:34
34 is present at location 3.

--------------------------------
Process exited after 13.45 seconds with return value 0
Press any key to continue . . .
```

PROGRAM9: To search an element using binary search in array

Code:

```
#include<stdio.h>
int main()
{
   int i, first, last, middle, n, search, array[100];
   printf("Enter number of elements: ");
   scanf("%d",&n);
   printf("Enter the elements: ", n);
   for ( i = 0 ; i < n ; i++ )
      scanf("%d",&array[i]);
   printf("Enter element to be found: ");
   scanf("%d",&search);
   first = 0;
   last = n - 1;
   middle = (first+last)/2;
   while( first <= last )
   {
      if ( array[middle] < search )
         first = middle + 1;
      else if ( array[middle] == search )
      {
         printf("%d found at location %d.\n", search, middle+1);
         break;
      }
      else
         last = middle - 1;
      middle = (first + last)/2;
   }
   if ( first > last )
      printf("Not found! %d is not present in the list.\n", search);
   return 0;
}
```

Sample input & output:



PROGRAM10: To reverse a given string

Code:

```
#include <stdio.h>
#include <string.h>
int main()
{
char str[100], temp;
int i = 0, j =0;
printf (" Enter a string: ");
scanf( "%s", str);
j = strlen (str) - 1;
while ( i < j)
{
temp = str[j];
str[j] = str[i];
str[i] = temp;
i++;
j--;
}
printf (" The reversed of the string: %s", str);
return 0;
}
```

Sample input & output:

```
 Enter a string: ROSHAN
 The reversed of the string: NAHSOR
--------------------------------
Process exited after 6.848 seconds with return value 0
Press any key to continue . . .
```

PROGRAM11: To check if given string is palindrome

Code:

```
#include <stdio.h>
#include <string.h>
int main(){
   char str[100];
   int i, len;
   int flag = 0;
   printf("Enter a string: ");
   scanf("%s", str);
   len = strlen(str);
   for(i=0;i < len ;i++){
      if(str[i] != str[len-i-1]){
         flag = 1;
         break;
        }
      }
   if (flag) {
      printf("%s is not a palindrome", str);
   }
   else {
      printf("%s is a palindrome", str);
   }
   return 0;
}
```

Sample input & output:

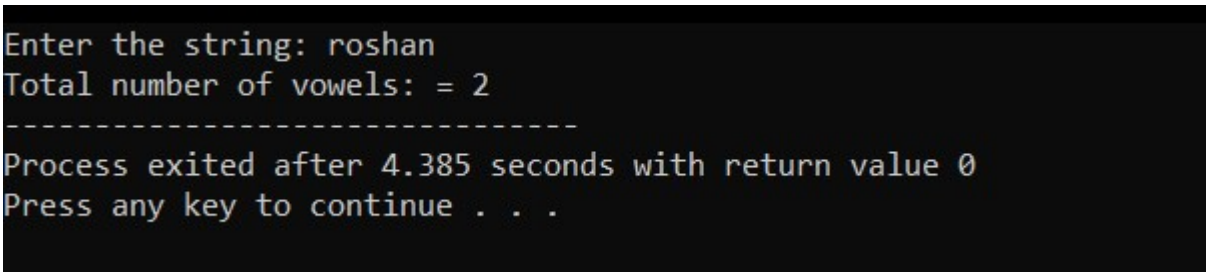
```
Enter a string: DAD
DAD is a palindrome
--------------------------------
Process exited after 14.65 seconds with return value 0
Press any key to continue . . .
```

PROGRAM12: To check & count no. of times vowels are present

Code:

```
#include <stdio.h>
#include <string.h>
int main()
{
    char str[100];
    int i, vowels = 0;
    printf("Enter the string: ");
    scanf("%s",&str);
    for(i = 0; str[i]; i++)
    {
        if(str[i]=='a'|| str[i]=='e'||str[i]=='i'||
           str[i]=='o'|| str[i]=='u'||str[i]=='A'||
           str[i]=='E'||str[i]=='I'||str[i]=='O' ||str[i]=='U')
        {
            vowels++;
        }
    }
    printf("Total number of vowels: = %d",vowels);
}
```

Sample input & output:



PROGRAM13: For matrix multiplication

Code:

```
#include <stdio.h>
int getMatrixElements(int matrix[][10], int row, int column) {
   printf("\nEnter elements: \n");
   for (int i = 0; i < row; ++i) {
      for (int j = 0; j < column; ++j) {
         printf("Enter a%d%d: ", i + 1, j + 1);
         scanf("%d", &matrix[i][j]);
```

```
    }
  }
}
int multiplyMatrices(int first[][10],
                     int second[][10],
                     int result[][10],
                     int r1, int c1, int r2, int c2) {
  for (int i = 0; i < r1; ++i) {
    for (int j = 0; j < c2; ++j) {
      result[i][j] = 0;
    }
  }
  for (int i = 0; i < r1; ++i) {
    for (int j = 0; j < c2; ++j) {
      for (int k = 0; k < c1; ++k) {
        result[i][j] += first[i][k] * second[k][j];
      }
    }
  }
}
int display(int result[][10], int row, int column) {
  printf("\nOutput Matrix:\n");
  for (int i = 0; i < row; ++i) {
    for (int j = 0; j < column; ++j) {
      printf("%d  ", result[i][j]);
      if (j == column - 1)
        printf("\n");
    }
  }
}
int main() {
  int first[10][10], second[10][10], result[10][10], r1, c1, r2, c2;
  printf("Enter rows and column for the first matrix: ");
  scanf("%d %d", &r1, &c1);
  printf("Enter rows and column for the second matrix: ");
  scanf("%d %d", &r2, &c2);
  while (c1 != r2) {
    printf("Error! Enter rows and columns again.\n");
    printf("Enter rows and columns for the first matrix: ");
    scanf("%d%d", &r1, &c1);
```

```
    printf("Enter rows and columns for the second matrix: ");
    scanf("%d%d", &r2, &c2);
  }
  getMatrixElements(first, r1, c1);
  getMatrixElements(second, r2, c2);
  multiplyMatrices(first, second, result, r1, c1, r2, c2);
  display(result, r1, c2);
}
```

Sample input & output:

```
Enter rows and column for the first matrix: 3 3
Enter rows and column for the second matrix: 3 3

Enter elements:
Enter a11: 1
Enter a12: 2
Enter a13: 3
Enter a21: 2
Enter a22: 1
Enter a23: 2
Enter a31: 3
Enter a32: 2
Enter a33: 3

Enter elements:
Enter a11: 4
Enter a12: 5
Enter a13: 6
Enter a21: 7
Enter a22: 8
Enter a23: 4
Enter a31: 3
Enter a32: 2
Enter a33: 1

Output Matrix:
27  27  17
21  22  18
35  37  29
```