

DATA STRUCTURE

NAME:D.Rohan Satyendra

REG.NO:192311359

COURSE CODE:CSA0312

MINIMUM AND MAXIMUM NUMBER IN A GIVEN ARRAY:

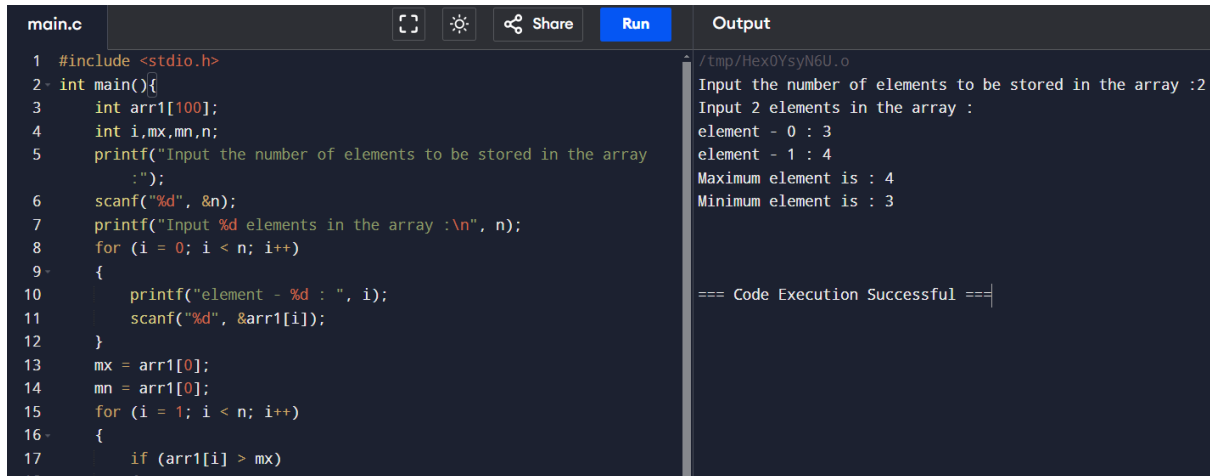
INPUT:

```
#include <stdio.h>

int main(){
    int arr1[100];
    int i,mx,mn,n;
    printf("Input the number of elements to be stored in the array :");
    scanf("%d", &n);
    printf("Input %d elements in the array :\n", n);
    for (i = 0; i < n; i++) {
        printf("element - %d : ", i);
        scanf("%d", &arr1[i]);
    }
    mx = arr1[0];
    mn = arr1[0];
    for (i = 1; i < n; i++) {
        if (arr1[i] > mx) {
            mx = arr1[i];
        }
        if (arr1[i] < mn) {
            mn = arr1[i];
        }
    }
    printf("Maximum element is : %d\n", mx);
    printf("Minimum element is : %d\n\n", mn);
    return 0;
```

```
}
```

OUTPUT:



```
main.c [Icons] [Share] [Run] Output
1 #include <stdio.h>
2 int main(){
3     int arr1[100];
4     int i,mx,mn,n;
5     printf("Input the number of elements to be stored in the array :");
6     scanf("%d", &n);
7     printf("Input %d elements in the array :\n", n);
8     for (i = 0; i < n; i++)
9     {
10         printf("element - %d : ", i);
11         scanf("%d", &arr1[i]);
12     }
13     mx = arr1[0];
14     mn = arr1[0];
15     for (i = 1; i < n; i++)
16     {
17         if (arr1[i] > mx)
18             mx = arr1[i];
19         if (arr1[i] < mn)
20             mn = arr1[i];
21     }
22     printf("Maximum element is : %d\n", mx);
23     printf("Minimum element is : %d\n", mn);
24     return 0;
25 }
```

/tmp/Hex0YsyN6U.o

Input the number of elements to be stored in the array :2

Input 2 elements in the array :

element - 0 : 3

element - 1 : 4

Maximum element is : 4

Minimum element is : 3

=== Code Execution Successful ===

SUM OF EVEN AND ODD IN AN ARRAY:

INPUT:

```
#include <stdio.h>
```

```
int main() {
```

```
int arr[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
```

```
int oddSum = 0, evenSum = 0;
```

```
int size = sizeof(arr) / sizeof(arr[0]);
```

```
for (int i = 0; i < size; i++) {
```

```
if (arr[i] % 2 == 0) {
```

```
evenSum += arr[i];
```

```
} else {
```

```
oddSum += arr[i];
```

```
}
```

```
}
```

```
printf("Sum of odd numbers: %d\n", oddSum);
```

```
printf("Sum of even numbers: %d\n", evenSum);
```

```
return 0;
```

```
}
```

OUTPUT:

main.c	Output
<pre>1 #include<stdio.h> 2 int main(){ 3 int arr[]={1,2,3,4,5,6,7,8,9,10}; 4 int oddSum=0,evenSum=0; 5 int size=sizeof(arr)/sizeof(arr[0]); 6 for (int i=0;i<size;i++){ 7 if (arr[i]%2==0){ 8 evenSum+=arr[i]; 9 }else{ 10 oddSum+=arr[i]; 11 } 12 } 13 printf("Sum of odd numbers: %d\n",oddSum); 14 printf("Sum of even numbers: %d\n",evenSum); 15 return 0; 16 }</pre>	<pre>/tmp/0NgnfvJgmH.o Sum of odd numbers: 25 Sum of even numbers: 30 === Code Execution Successful ===</pre>

MERGE NUMBERS IN AN ARRAY:

INPUT:

```
#include <stdio.h>

void mergeNumbers(int arr[], int size, int num1, int num2) {

    arr[size] = num1;

    arr[size + 1] = num2;

}

int main() {

    int arr[100] = {1, 2, 3, 4, 5};

    int size = 5;

    int num1 = 6, num2 = 7;

    mergeNumbers(arr, size, num1, num2);

    for (int i = 0; i < size + 2; i++) {

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

    }

    return 0;

}
```

OUTPUT:

main.c	Run	Output
<pre>1 #include<stdio.h> 2 void mergeNumbers(int arr[],int size,int num1,int num2){ 3 arr[size]=num1; 4 arr[size+1]=num2; 5 } 6 int main(){ 7 int arr[100]={1,2,3,4,5}; 8 int size=5; 9 int num1=6,num2=7; 10 mergeNumbers(arr,size,num1,num2); 11 for (int i=0;i<size+2;i++){ 12 printf("%d ",arr[i]); 13 } 14 return 0; 15 } 16</pre>	<div>Share</div>	<div>/tmp/zRC5Djr9MZ.o</div> <div>1 2 3 4 5 6 7</div> <div>=== Code Execution Successful ===</div>

DUPLICATE ELEMENTS IN A GIVEN ARRAY:

INPUT:

```
#include <stdio.h>
```

```
int findDuplicate(int arr[], int size) {
```

```
for (int i = 0; i < size; i++) {
```

```
for (int j = i + 1; j < size; j++) {
```

```
if (arr[i] == arr[j]) {
```

```
return arr[i];
```

```
}
```

```
}
```

```
}
```

```
return -1;
```

```
}
```

```
int main() {
```

```
int arr[] = {1, 2, 3, 4, 5, 3};
```

```
int size = sizeof(arr) / sizeof(arr[0]);
```

```
int duplicate = findDuplicate(arr, size);
```

```
if (duplicate != -1) {
```

```
printf("Duplicate element: %d\n", duplicate);
```

```
} else {
```

```
printf("No duplicate element found.\n");
```

```
}  
  
return 0;  
  
}
```

OUTPUT:

main.c	Output
<pre>1 #include<stdio.h> 2 int findDuplicate(int arr[],int size){ 3 for(int i=0;i<size;i++){ 4 for(int j=i+1;j<size;j++){ 5 if(arr[i]==arr[j]){ 6 return arr[i]; 7 } 8 } 9 } 10 return -1; 11 } 12 int main(){ 13 int arr[]={1,2,3,4,5,3}; 14 int size=sizeof(arr)/sizeof(arr[0]); 15 int duplicate=findDuplicate(arr, size); 16 if (duplicate!=-1){ 17 printf("Duplicate element: %d\n",duplicate); 18 }else{ 19 printf("No duplicate element found.\n"); 20 } 21 return 0; 22 }</pre>	<pre>/tmp/JmCVYgov0q.o Duplicate element: 3 === Code Execution Successful ===</pre>