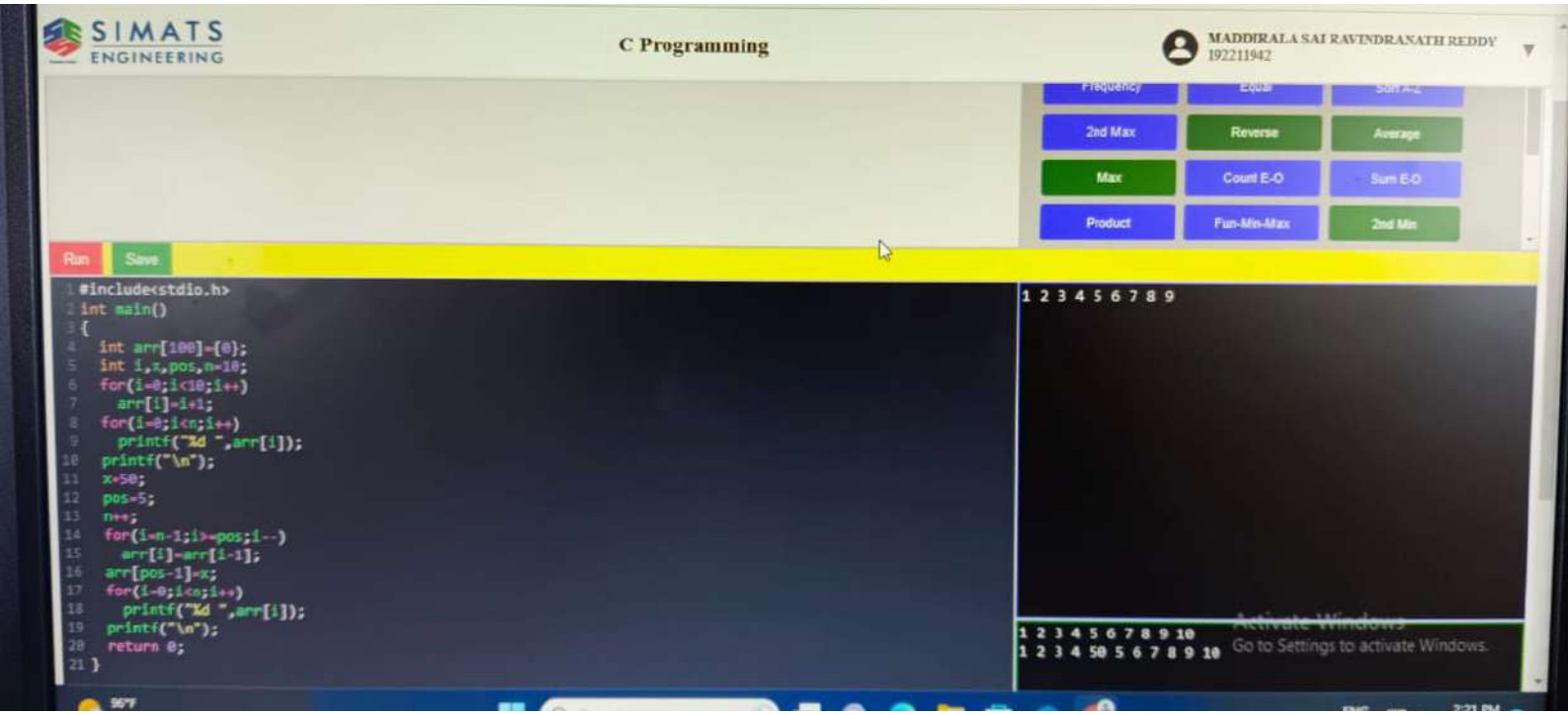
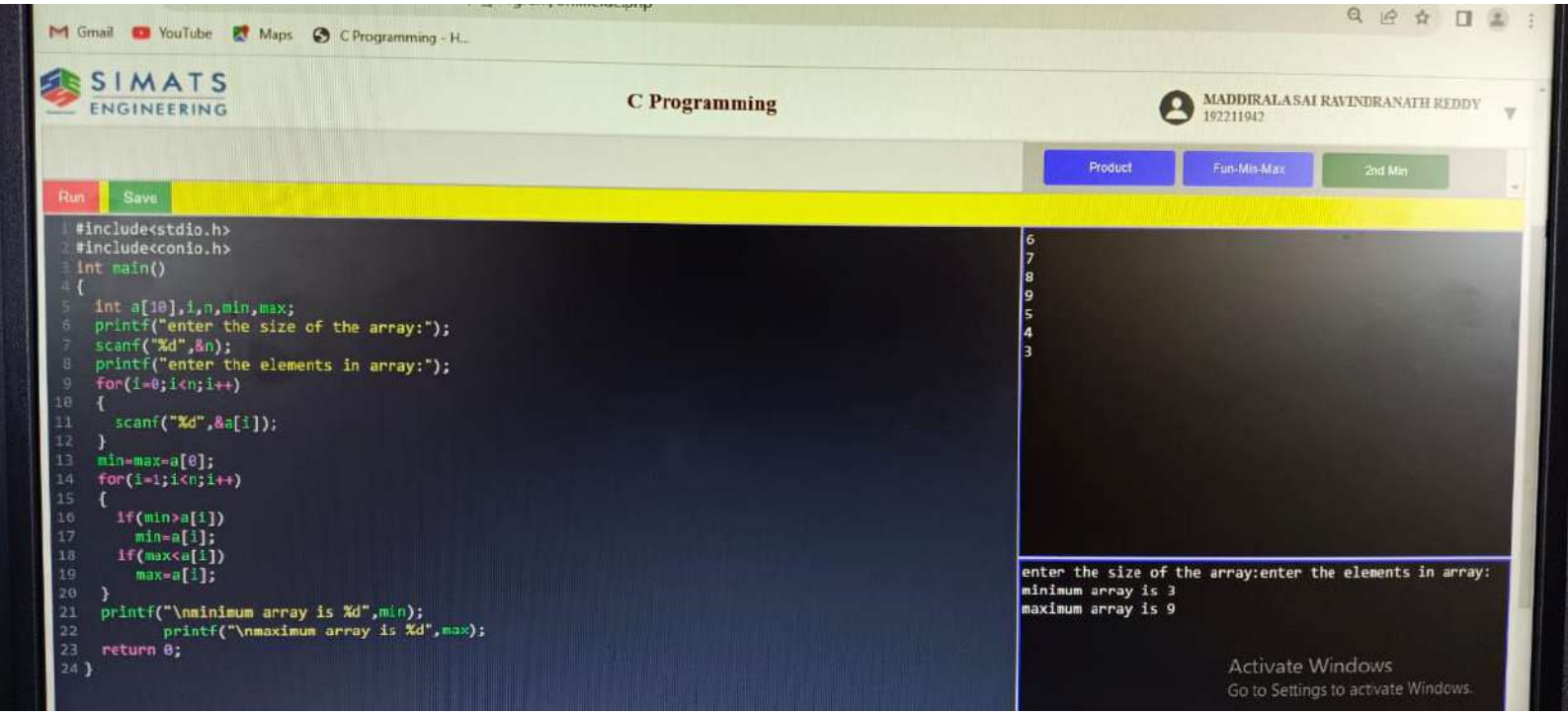
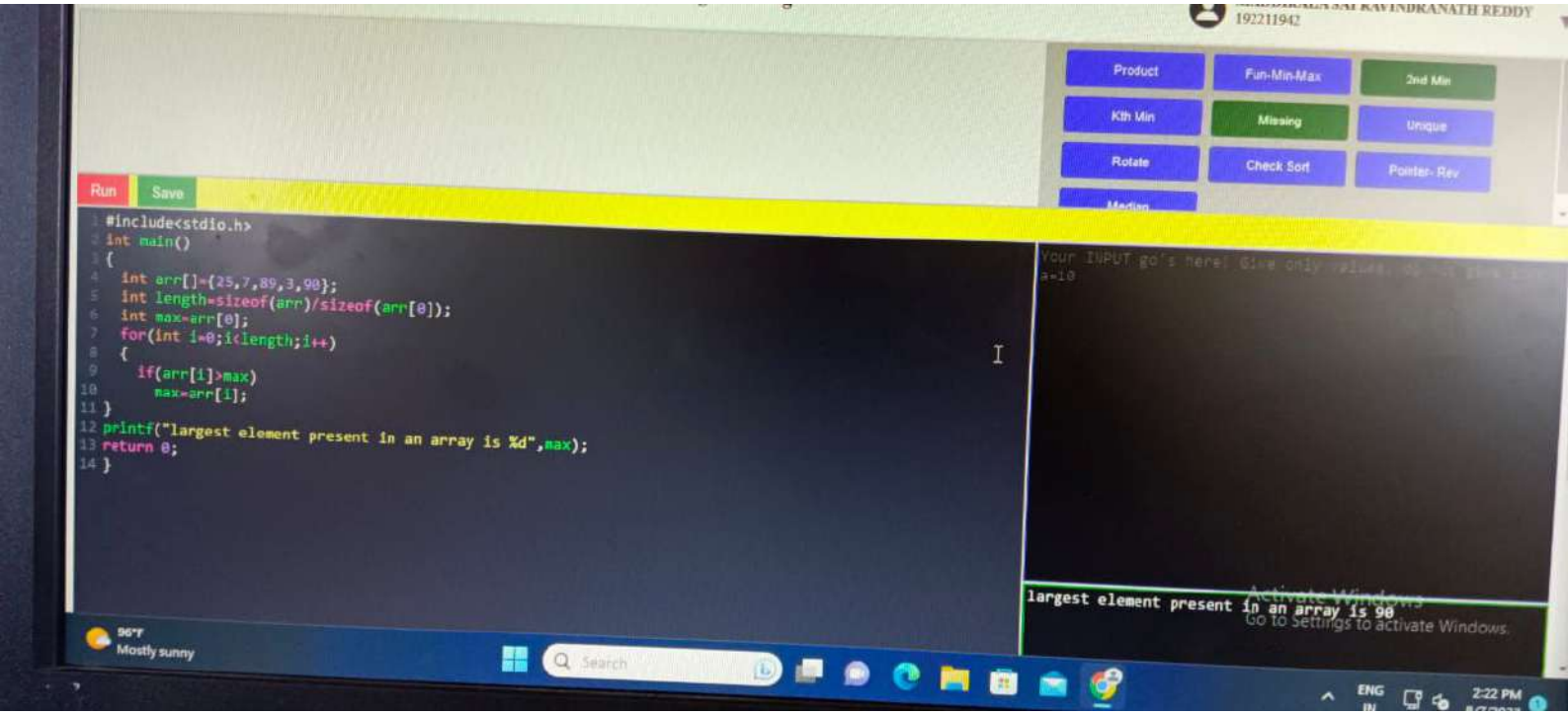
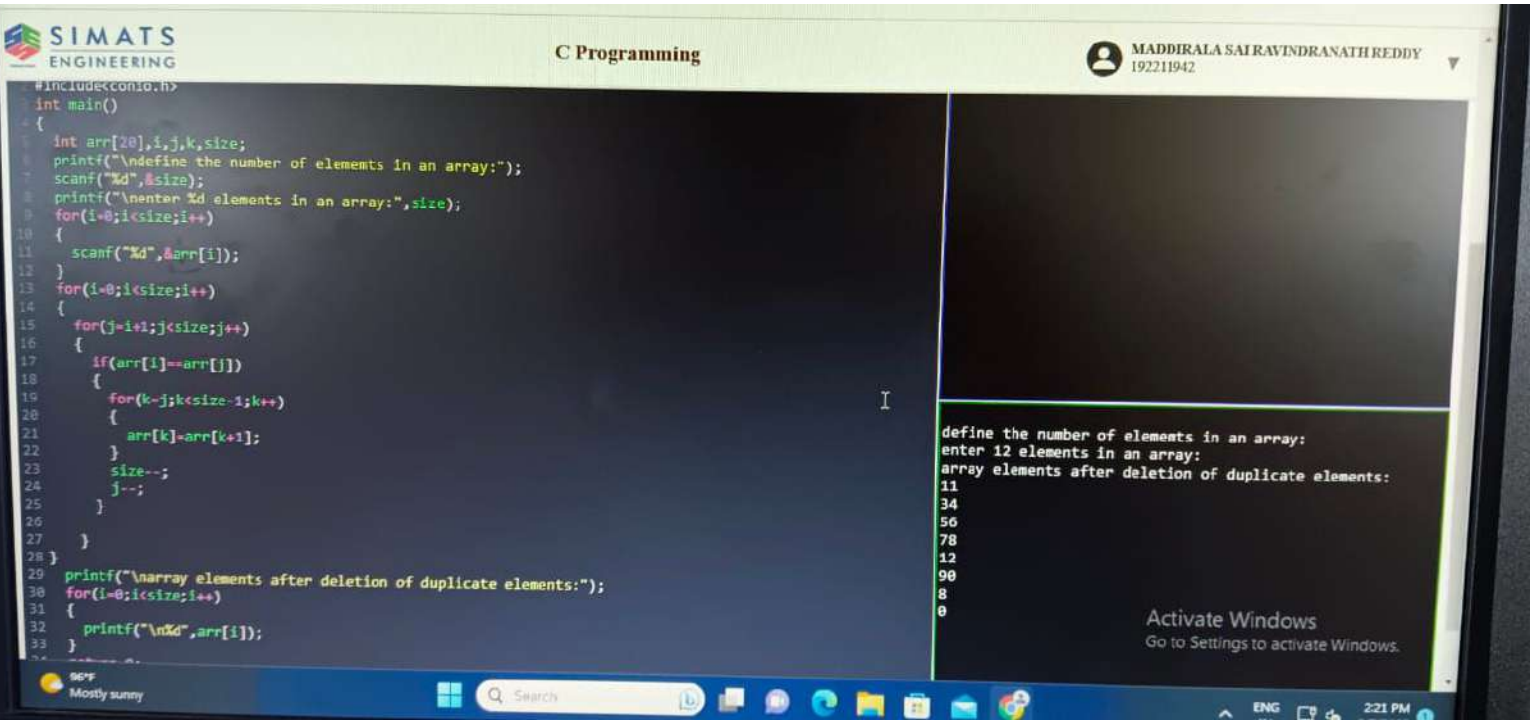
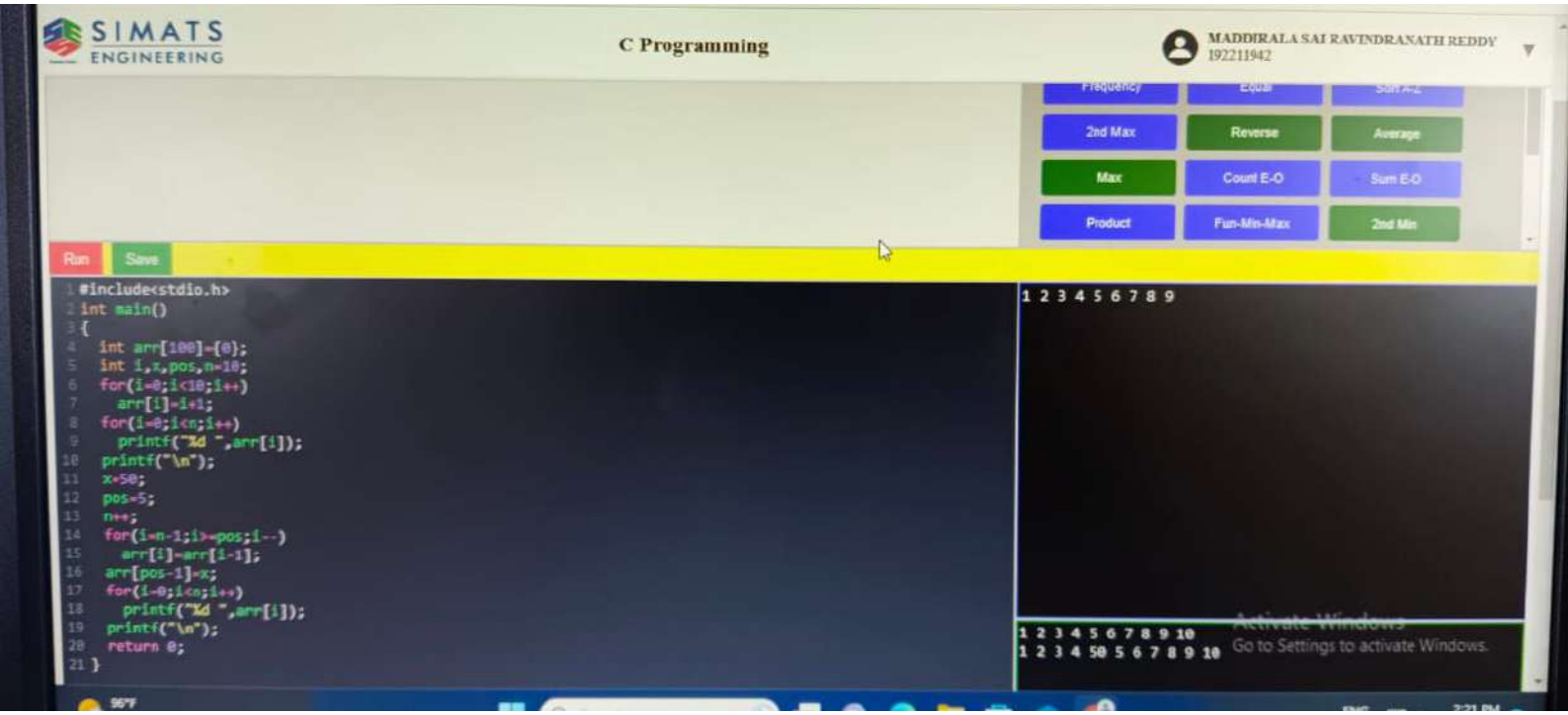
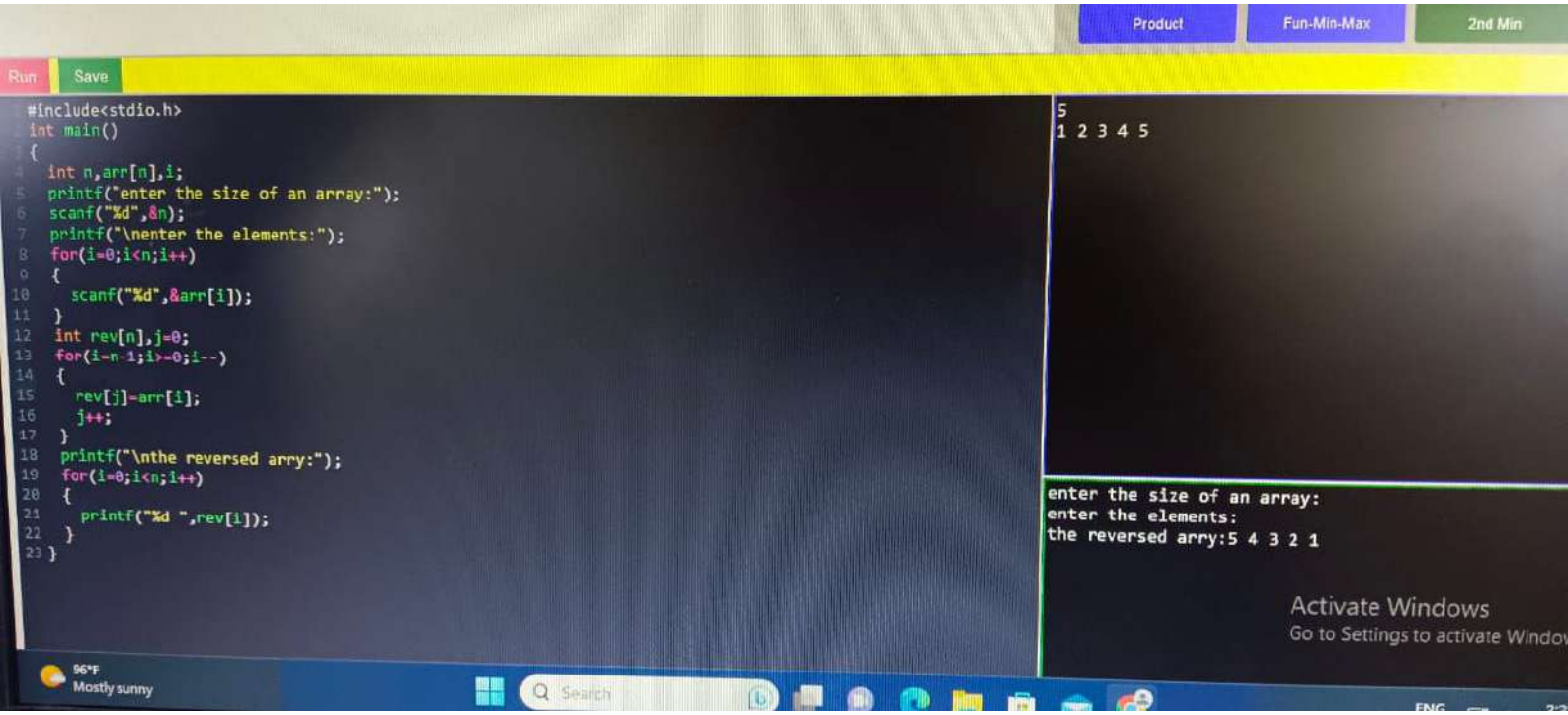

SIMATS
ENGINEERING
C Programming
MADDIRALA SAI RAVINDRANATH REDDY
192211942
2nd Max
Reverse
Average
Max
Count E-O
Sum E-O
Product
Fun-Min-Max
2nd Min
Run
Save
#include<stdio.h>
int main()
{
int arr[100]={0};
int i,x,pos,n=10;
for(i=0;i<10;i++)
arr[i]=i+1;
for(i=0;i<n;i++)
printf("%d ",arr[i]);
printf("\n");
x=50;
pos=5;
n++;
for(i=n-1;i>=pos;i--)
arr[i]=arr[i-1];
arr[pos-1]=x;
for(i=0;i<n;i++)
printf("%d ",arr[i]);
printf("\n");
return 0;
}
1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9 10
1 2 3 4 50 5 6 7 8 9 10
Activate Windows
Go to Settings to activate Windows.

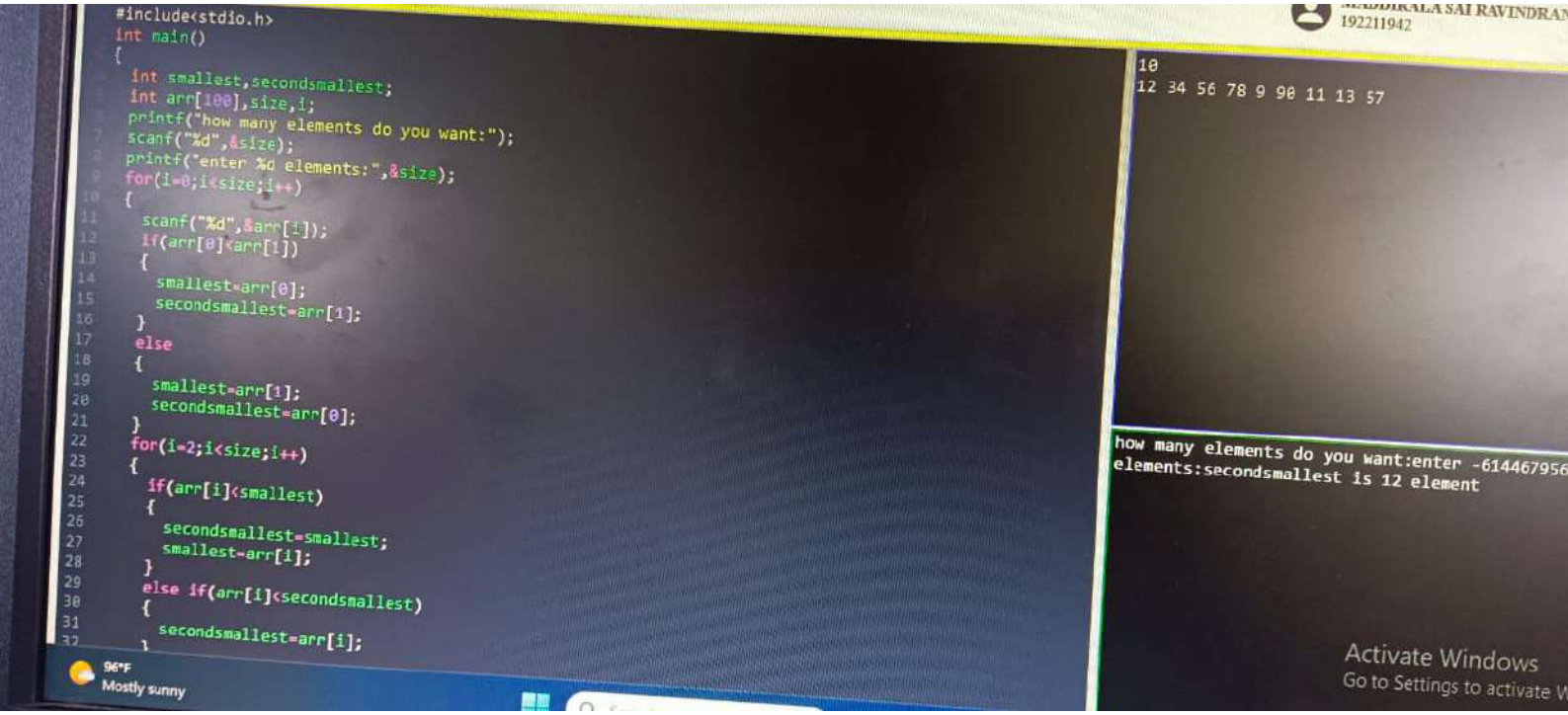

Gmail
YouTube
Maps
C Programming - H...
SIMATS ENGINEERING
C Programming
MADDIRALASAI RAVINDRANATH REDDY
192211942
Product
Fun-Min-Max
2nd Min
Run
Save
#include<stdio.h>
#include<conio.h>
int main()
{
int a[10],i,n,min,max;
printf("enter the size of the array:");
scanf("%d",&n);
printf("enter the elements in array:");
for(i=0;i<n;i++)
{
scanf("%d",&a[i]);
}
min=max=a[0];
for(i=1;i<n;i++)
{
if(min>a[i])
min=a[i];
if(max<a[i])
max=a[i];
}
printf("\nminimum array is %d",min);
printf("\nmaximum array is %d",max);
return 0;
}
6
7
8
9
5
4
3
enter the size of the array:enter the elements in array:
minimum array is 3
maximum array is 9
Activate Windows
Go to Settings to activate Windows.


192211942
Product
Fun-Min-Max
2nd Min
Kth Min
Missing
Unique
Rotate
Check Sort
Pointer- Rev
Run
Save
#include<stdio.h>
int main()
{
int arr[]={25,7,89,3,90};
int length=sizeof(arr)/sizeof(arr[0]);
int max=arr[0];
for(int i=0;i<length;i++)
{
if(arr[i]>max)
max=arr[i];
}
printf("largest element present in an array is %d",max);
return 0;
}
Your INPUT go's here! Give only values, do not give
a=10
largest element present in an array is 90
Activate Windows
Go to Settings to activate Windows.
96°F
Mostly sunny
Search
ENG
IN
2:22 PM


SIMATS
ENGINEERING
C Programming
MADDIRALA SAI RAVINDRANATH REDDY
192211942
#include<conio.h>
int main()
{
int arr[20],i,j,k,size;
printf("\ndefine the number of elememts in an array:");
scanf("%d",&size);
printf("\nenter %d elements in an array:",size);
for(i=0;i<size;i++)
{
scanf("%d",&arr[i]);
}
for(i=0;i<size;i++)
{
for(j=i+1;j<size;j++)
{
if(arr[i]==arr[j])
{
for(k=j;k<size-1;k++)
{
arr[k]=arr[k+1];
}
size--;
j--;
}
}
}
printf("\narray elements after deletion of duplicate elements:");
for(i=0;i<size;i++)
{
printf("\n%d",arr[i]);
}
define the number of elements in an array:
enter 12 elements in an array:
array elements after deletion of duplicate elements:
11
34
56
78
12
90
8
0
Activate Windows
Go to Settings to activate Windows.
96°F
Mostly sunny
Search
ENG
2:21 PM


SIMATS
ENGINEERING
C Programming
MADDIRALA SAI RAVINDRANATH REDDY
192211942
2nd Max
Reverse
Average
Max
Count E-O
Sum E-O
Product
Fun-Min-Max
2nd Min
Run
Save
#include<stdio.h>
int main()
{
int arr[100]={0};
int i,x,pos,n=10;
for(i=0;i<10;i++)
arr[i]=i+1;
for(i=0;i<n;i++)
printf("%d ",arr[i]);
printf("\n");
x=50;
pos=5;
n++;
for(i=n-1;i>=pos;i--)
arr[i]=arr[i-1];
arr[pos-1]=x;
for(i=0;i<n;i++)
printf("%d ",arr[i]);
printf("\n");
return 0;
}
1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9 10
1 2 3 4 50 5 6 7 8 9 10
Activate Windows
Go to Settings to activate Windows.

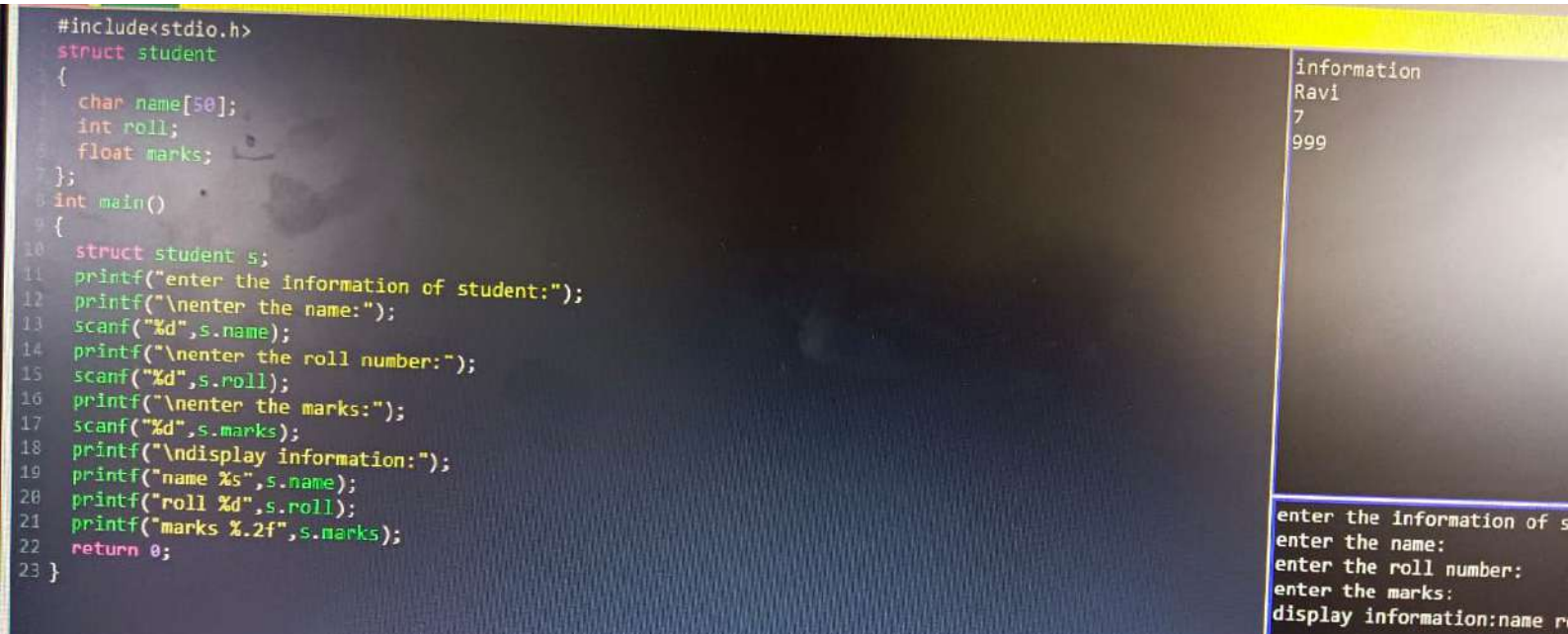

Product
Fun-Min-Max
2nd Min
Run
Save
#include<stdio.h>
int main()
{
int n,arr[n],i;
printf("enter the size of an array:");
scanf("%d",&n);
printf("\nenter the elements:");
for(i=0;i<n;i++)
{
scanf("%d",&arr[i]);
}
int rev[n],j=0;
for(i=n-1;i>=0;i--)
{
rev[j]=arr[i];
j++;
}
printf("\nthe reversed arry:");
for(i=0;i<n;i++)
{
printf("%d ",rev[i]);
}
}
5
1 2 3 4 5
enter the size of an array:
enter the elements:
the reversed arry:5 4 3 2 1
Activate Windows
Go to Settings to activate Windo
96°F
Mostly sunny
Search
ENG


#include<stdio.h>
int main()
{
int smallest,secondsmallest;
int arr[100],size,i;
printf("how many elements do you want:");
scanf("%d",&size);
printf("enter %d elements:",&size);
for(i=0;i<size;i++)
{
scanf("%d",&arr[i]);
if(arr[0]<arr[1])
{
smallest=arr[0];
secondsmallest=arr[1];
}
else
{
smallest=arr[1];
secondsmallest=arr[0];
}
for(i=2;i<size;i++)
{
if(arr[i]<smallest)
{
secondsmallest=smallest;
smallest=arr[i];
}
else if(arr[i]<secondsmallest)
{
secondsmallest=arr[i];
192211942
10
12 34 56 78 9 90 11 13 57
how many elements do you want:enter -614467956
elements:secondsmallest is 12 element
96°F
Mostly sunny
Activate Windows
Go to Settings to activate

```c
#include<conio.h>
int main()
{
  int arr[50],pos,i,num;
  printf("enter the number of elements in an array:");
  scanf("%d",&num);
  printf("enter %d elements in an array:");
  for(i=0;i<num;i++)
  {
    printf("arr[%d]=",i);
    scanf("%d",&arr[i]);
  }
  printf("define the position of element where u want to delete:");
  scanf("%d",&pos);
  if(pos>=num+1)
  {
    printf("deletion is not possible in the arry:");
  }
  else
  {
    for(i=pos-1;i<num-1;i++)
    {
      arr[i]=arr[i+1];
      }
    printf("\nthe resultant array is:\n");
    for(i=0;i<num-1;i++)
    {
      printf("arr[%d]=",i);
      printf("%d",arr[i]);
    }
  }
  return 0;
```

```
2
3
4
5
6
7
```

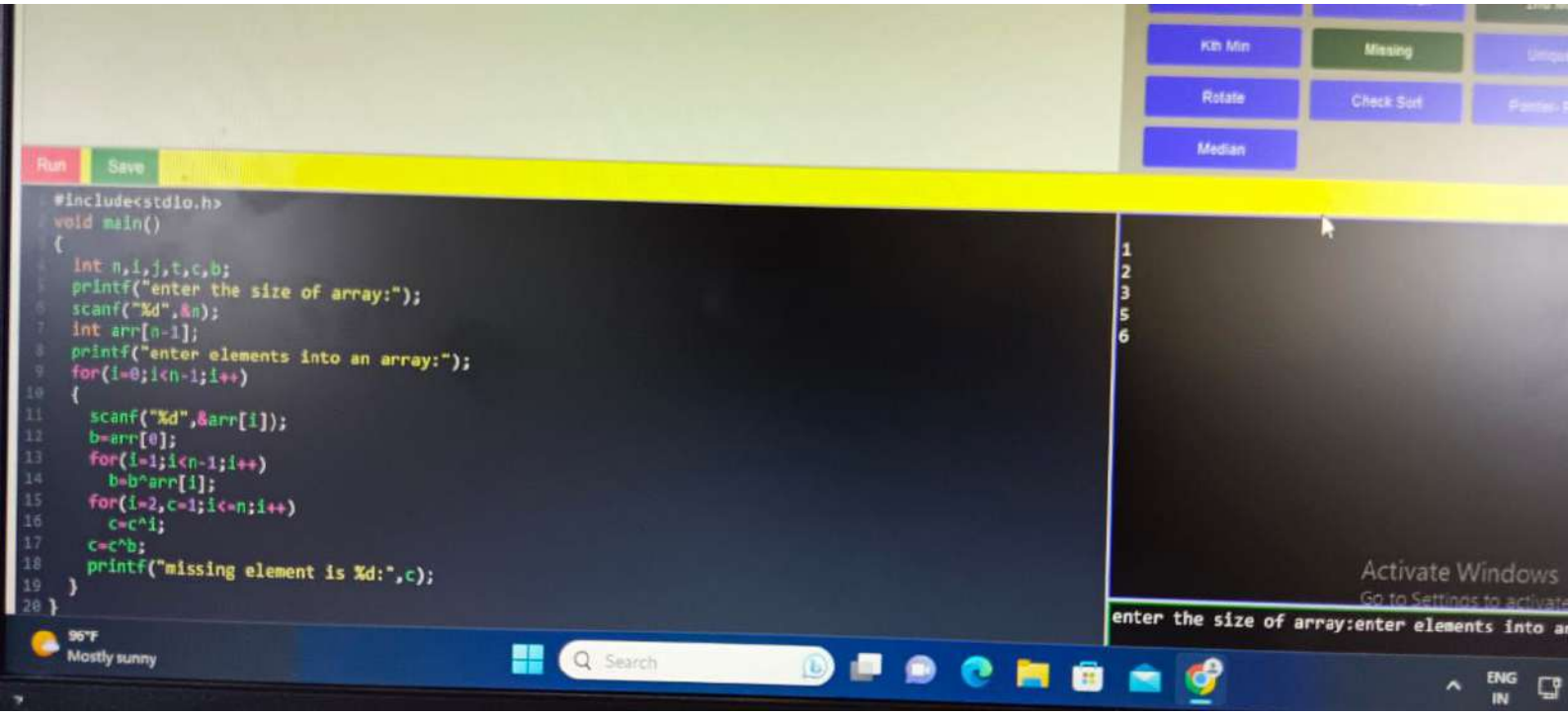
```
You can not do delete operation
 Can not make directory
 Can not List the Directory or Get Path
```

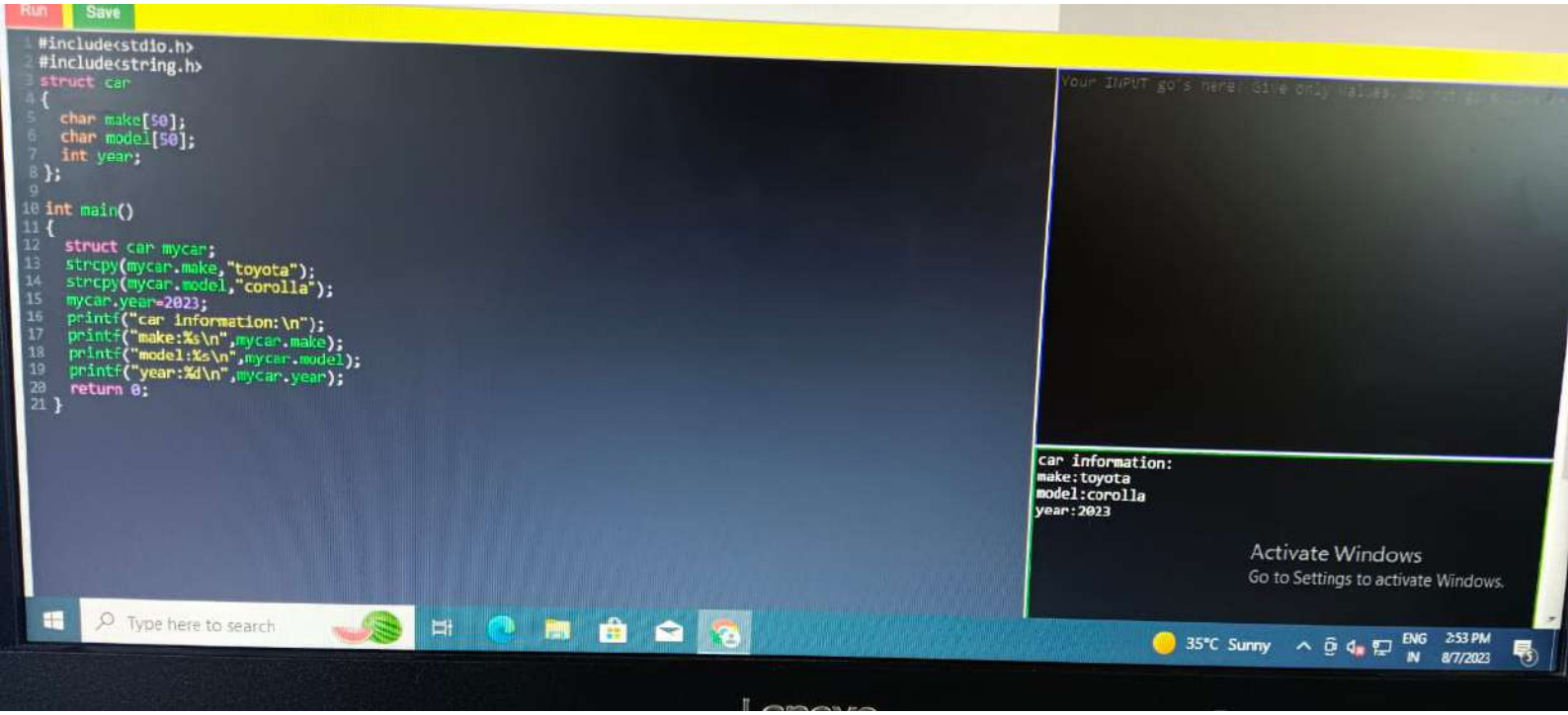


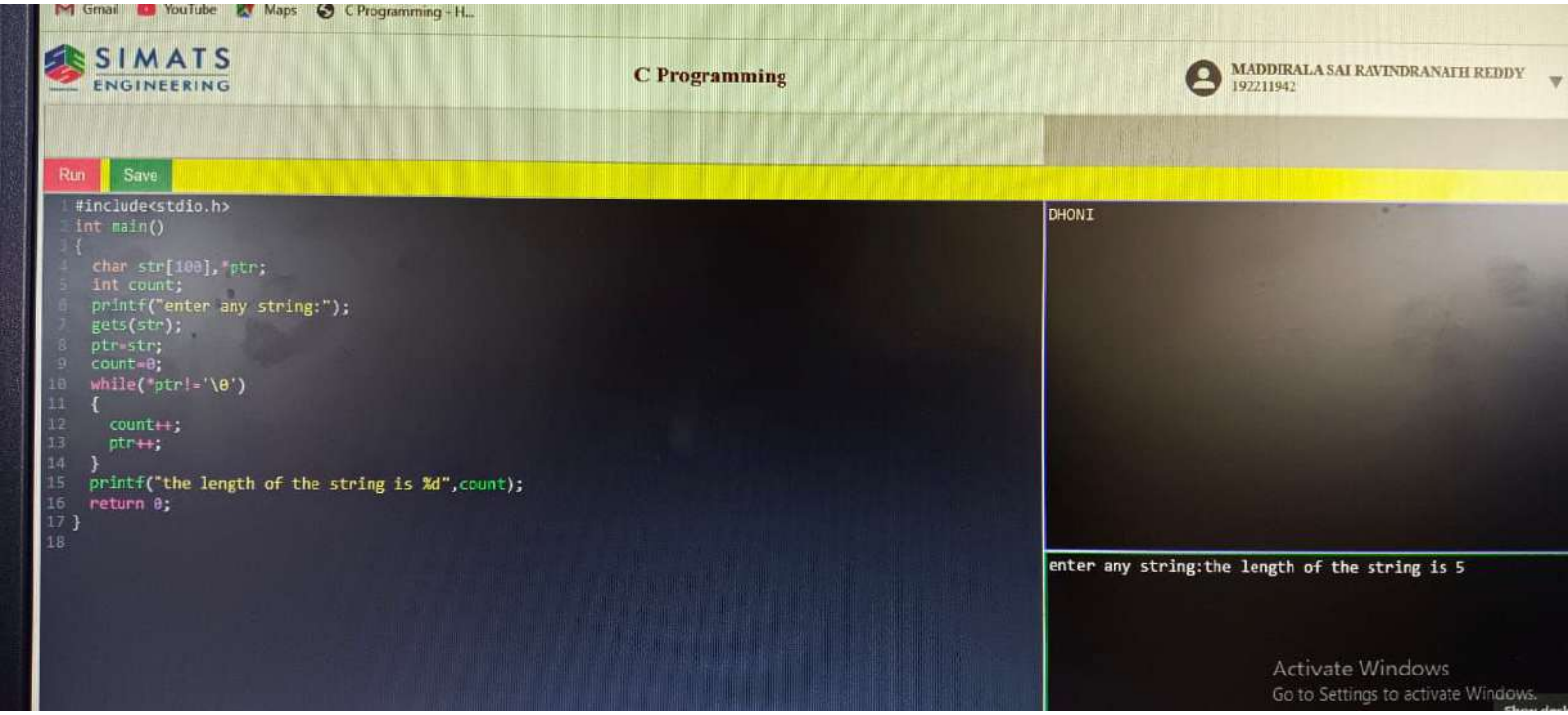
```c
#include<stdio.h>
struct student
{
  char name[50];
  int roll;
  float marks;
};
int main()
{
  struct student s;
  printf("enter the information of student:");
  printf("\nenter the name:");
  scanf("%d",s.name);
  printf("\nenter the roll number:");
  scanf("%d",s.roll);
  printf("\nenter the marks:");
  scanf("%d",s.marks);
  printf("\ndisplay information:");
  printf("name %s",s.name);
  printf("roll %d",s.roll);
  printf("marks %.2f",s.marks);
  return 0;
}
```

```
information
Ravi
7
999
```

```
enter the information of s
enter the name:
enter the roll number:
enter the marks:
display information:name r
```


Kth Min
Missing
Rotate
Check Sort
Median
Run
Save
#include<stdio.h>
void main()
{
int n,i,j,t,c,b;
printf("enter the size of array:");
scanf("%d",&n);
int arr[n-1];
printf("enter elements into an array:");
for(i=0;i<n-1;i++)
{
scanf("%d",&arr[i]);
b=arr[0];
for(i=1;i<n-1;i++)
b=b^arr[i];
for(i=2,c=1;i<=n;i++)
c=c^i;
c=c^b;
printf("missing element is %d:",c);
}
}
1
2
3
5
6
Activate Windows
enter the size of array:enter elements into a
96°F
Mostly sunny
Search
ENG
IN


Run
Save
#include<stdio.h>
#include<string.h>
struct car
{
char make[50];
char model[50];
int year;
};
int main()
{
struct car mycar;
strcpy(mycar.make,"toyota");
strcpy(mycar.model,"corolla");
mycar.year=2023;
printf("car information:\n");
printf("make:%s\n",mycar.make);
printf("model:%s\n",mycar.model);
printf("year:%d\n",mycar.year);
return 0;
}
car information:
make:toyota
model:corolla
year:2023
Activate Windows
Go to Settings to activate Windows.
Type here to search
35°C Sunny
ENG
IN
2:53 PM
8/7/2023


Gmail
YouTube
Maps
C Programming - H...
SIMATS
ENGINEERING
C Programming
MADDIRALA SAI RAVINDRANATH REDDY
192211942
Run
Save
#include<stdio.h>
int main()
{
char str[100],*ptr;
int count;
printf("enter any string:");
gets(str);
ptr=str;
count=0;
while(*ptr!='\0')
{
count++;
ptr++;
}
printf("the length of the string is %d",count);
return 0;
}
DHONI
enter any string:the length of the string is 5
Activate Windows
Go to Settings to activate Windows.

```c
#include <stdio.h>
void swap(char *x, char *y)
{
    char temp = *x;
    *x = *y;
    *y = temp;
}
void reverse(char *str, int k)
{
    static int i = 0;
    if (*(str + k) == '\0') {
        return;
    }

    reverse(str, k + 1);

    if (i <= k) {
        swap(&str[i++], &str[k]);
    }
}

int main()
{
    char str[] = "sse";

    reverse(str, 0);
    printf("Reverse of the given string is %s", str);

    return 0;
}
```

```
ram
```

```
Reverse of the given string is ess
```

```c
#include <stdio.h>

void towers(int, char, char, char);

int main()
{
    int num;

    printf("Enter the number of disks : ");
    scanf("%d", &num);
    printf("The sequence of moves involved in the Tower of Hanoi are :\n");
    towers(num, 'A', 'C', 'B');
    return 0;
}
void towers(int num, char frompeg, char topeg, char auxpeg)
{
    if (num == 1)
    {
        printf("\n Move disk 1 from peg %c to peg %c", frompeg, topeg);
        return;
    }
    towers(num - 1, frompeg, auxpeg, topeg);
    printf("\n Move disk %d from peg %c to peg %c", num, frompeg, topeg);
    towers(num - 1, auxpeg, topeg, frompeg);
}
```

```
ram
```

```
Enter the number of disks : The sequence of moves involved in the
Tower of Hanoi are :

 Move disk 1 from peg A to peg C
 Move disk 2 from peg A to peg B
 Move disk 1 from peg C to peg B
 Move disk 3 from peg A to peg C
 Move disk 1 from peg B to peg A
 Move disk 2 from peg B to peg C
 Move disk 1 from peg A to peg C
```

```c
#include <stdio.h>
#include <string.h>
void swap(char* x, char* y)
{
    char temp;
    temp = *x;
    *x = *y;
    *y = temp;
}
void permute(char* a, int l, int r)
{
    int i;
    if (l == r)
        printf("%s\n", a);
    else {
        for (i = l; i <= r; i++) {
            swap((a + l), (a + i));
            permute(a, l + 1, r);
            swap((a + l), (a + i));
        }
    }
}
int main()
{
    char str[] = "ABC";
    int n = strlen(str);
    permute(str, 0, n - 1);
    return 0;
}
```

```
ram
```

```
ABC
ACB
BAC
BCA
CBA
CAB
```

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

void check(char [], int);

int main()
{
    char word[15];

    printf("Enter a word to check if it is a palindrome\n");
    scanf("%s", word);
    check(word, 0);

    return 0;
}

void check(char word[], int index)
{
    int len = strlen(word) - (index + 1);
    if (word[index] == word[len])
    {
        if (index + 1 == len || index == len)
        {
            printf("The entered word is a palindrome\n");
            return;
        }
        check(word, index + 1);
    }
    else
    {
        printf("The entered word is not a palindrome\n");
    }
}
```

```
ram
```

```
Enter a word to check if it is a palindrome
The entered word is not a palindrome
```

```c
#define ARRAY_SIZE(a)  sizeof(a)/sizeof(a[0])
int sumArrayElements(int arr[], const int N)
{
    if (N <= 0)
    {
        return 0;
    }
    return (sumArrayElements(arr, N - 1) + arr[N - 1]);
}
int main()
{
    int arr[] = { 1, 2, 3, 4, 5 };
    const int N = ARRAY_SIZE(arr);
    printf("%d\n", sumArrayElements(arr, N));
    return 0;
}
```

```
Your INPUT go's here! Give only values. do not give like a=10
```

```
15
```

```c
#include <stdio.h>
int hcf(int n1, int n2);
int main() {
    int n1, n2;
    printf("Enter two positive integers: ");
    scanf("%d %d", &n1, &n2);
    printf("G.C.D of %d and %d is %d.", n1, n2, hcf(n1, n2));
    return 0;
}

int hcf(int n1, int n2) {
    if (n2 != 0)
        return hcf(n2, n1 % n2);
    else
        return n1;
}
```

```
Your INPUT go's here! Give only values. do not give like a=10
```

```
Enter two positive integers: G.C.D of 0 and 16 is 16.
```

Run Save Changes Updated. Saved 293

```c
#include <stdio.h>
int sum (int a);
int main()
{
    int num, result;

    printf("Enter the number:\n ");
    scanf("%d", &num);
    result = sum(num);
    printf("Sum of digits in %d is %d\n", num, result);
    return 0;
}
int sum (int num)
{
    if (num != 0)
    {
        return (num % 10 + sum (num / 10));
    }
    else
    {
       return 0;
    }
}
```

```
22
```

```
Enter the number:
 Sum of digits in 22 is 4
```

```c
#include <stdio.h>
int power(int n1, int n2);
int main() {
    int base, a, result;
    printf("Enter base number:\n ");
    scanf("%d", &base);
    printf("Enter power number(positive integer):\n ");
    scanf("%d", &a);
    result = power(base, a);
    printf("%d^%d = %d", base, a, result);
    return 0;
}

int power(int base, int a) {
    if (a != 0)
        return (base * power(base, a - 1));
    else
        return 1;
}
```

```
Your INPUT go's here! Give only values. do not give like a=10
```

```
Enter base number: Enter power number(positive integer): 5^3 = 125
```

```c
#include<stdio.h>
int main()
{
  int i,j,k,rows;
  printf("Enter no.of rows:\n");
  scanf("%d",&rows);
  int space=rows-1,num=1;
  for(i=1;i<=rows;i++)
  {
    for(j=1;j<=space;j++)
    {
      printf(" ");
    }
    for(k=1;k<=num;k++){
      printf("*");
    }
    if(space>i)
    {
      space=space-1;
      num=num+2;
    }
    if(space<i){
      space=space+1;
      num=num-2;
    }
    printf("\n");
  }
}
```

```
s
```

```
Enter no.of rows:
    *
   ***
  *****
   ***
    *
```

Run Save

```c
#include<stdio.h>
long int multiplyNumbers(int n);
int main() {
    int n;
    printf("Enter a positive integer: \n");
    scanf("%d",&n);
    printf("Factorial of %d = %ld", n, multiplyNumbers(n));
    return 0;
}

long int multiplyNumbers(int n) {
    if (n>=1)
        return n*multiplyNumbers(n-1);
    else
        return 1;
}

```

```
8
```

Run Save

```c

#include <stdio.h>
int fib(int n)
{
    int a = 0, b = 1, c, i;
    if (n == 0)
        return a;
    for (i = 2; i <= n; i++) {
        c = a + b;
        a = b;
        b = c;
    }
    return b;
}

int main()
{
    int n = 9;
    printf("%d", fib(n));
    getchar();
    return 0;
}
```

```
8
```

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
char* copyString(char s[])
{
    int i;
    char* s2;
    s2 = (char*)malloc(20);

    for (i = 0; s[i] != '\0'; i++) {

        s2[i] = s[i];
    }
    return (char*)s2;
}
int main()
{
    char s1[20] = "c programming";
    char* s2;
    s2 = copyString(s1);
    printf("%s", s2);
    return 0;
}
```

```
5
8
```

```
c programming
```

```c
#include<stdio.h>
#include<string.h>
#define N 10

void print(int *num, int n)
{
    int i;
    for ( i = 0 ; i < n ; i++)
        printf("%d ", num[i]);
    printf("\n");
}
int main()
{
    int num[N];
    int *ptr;
    int temp;
    int i, n, j;
    printf("\nHow many number you want to enter: ");
        scanf("%d", &n);
    printf("\nEnter a list of numbers to see all combinations:\n");
    for (i = 0 ; i < n; i++)
        scanf("%d", &num[i]);
    for (j = 1; j <= n; j++) {
        for (i = 0; i < n-1; i++) {
            temp = num[i];
            num[i] = num[i+1];
            num[i+1] = temp;
            print(num, n);
    }
    }
    return 0;
}
```

```
Your INPUT go's here! Give only values. do not give like a=10
```

```
How many number you want to enter:
Enter a list of numbers to see all combinations:
```

```c
#include <stdio.h>
#define MAX_SIZE 100
int sum(int arr[], int start, int len);
int main()
{
    int arr[MAX_SIZE];
    int N, i, sumofarray;

    printf("Enter size of the array: \n");
    scanf("%d", &N);
    printf("Enter elements in the array: \n");
    for(i=0; i<N; i++)
    {
        scanf("%d", &arr[i]);
    }


    sumofarray = sum(arr, 0, N);
    printf("Sum of array elements: %d", sumofarray);

    return 0;
}

int sum(int arr[], int start, int len)
{
    if(start >= len)
        return 0;

    return (arr[start] + sum(arr, start + 1, len));
}
```

```
Your INPUT go's here! Give only values. do not give like a=10
```

```
Enter size of the array:
Enter elements in the array:
Sum of array elements: 15
```

```c
#include<stdio.h>

int find_len (char [], int);

int main ()
{
    char str[100]="Let's Learn C Programming";
    int len = 0;

    len = find_len (str, 0);

    printf ("The length of the given string is: %d\n", len);
    return 0;

}

int find_len (char str[], int index){

    static int l = 0;

    if (str[index] == '\0')
        return l;

    else
        l ++;
    find_len (str, index + 1);
}
```

```
Your INPUT go's here! Give only values. do not give like a=10
```

```
The length of the given string is: 25
```

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

int main()
{
    char s[1000],c;
    int i,count=0;

    printf("Enter  the string :\n");
    gets(s);
    printf("Enter character to be searched: \n");
    c=getchar();

    for(i=0;s[i];i++)
    {
        if(s[i]==c)
        {
          count++;
        }
    }

    printf("character '%c' occurs %d times \n ",c,count);


    return 0;
}
```

```
Your INPUT go's here! Give only values. do not give like a=10
```

```
Enter  the string :
Enter character to be searched:
character '1' occurs 2 times
```

```c
#include <stdio.h>
#include <math.h>
int reverse(int num);
int main()
{
    int num, rev;
    printf("Enter any number: ");
    scanf("%d", &num);
    rev = reverse(num);

    printf("Reverse of %d = %d", num, rev);

    return 0;
}
int reverse(int num)
{
    int digit = (int) log10(num);
    if(num == 0)
        return 0;

    return ((num%10 * pow(10, digit)) + reverse(num/10));
}
```

```
Your INPUT go's here! Give only values. do not give like a=10
```

```
Enter any number: Reverse of 16 = 61
```

```c
#include <stdio.h>
#include <string.h>
void swap(char* x, char* y)
{
    char temp;
    temp = *x;
    *x = *y;
    *y = temp;
}
void permute(char* a, int l, int r)
{
    int i;
    if (l == r)
        printf("%s\n", a);
    else {
        for (i = l; i <= r; i++) {
            swap((a + l), (a + i));
            permute(a, l + 1, r);
            swap((a + l), (a + i));
        }
    }
}
int main()
{
    char str[] = "ABC";
    int n = strlen(str);
    permute(str, 0, n - 1);
    return 0;
}
```

```
Your INPUT go's here! Give only values. do not give like a=10
```

```
ABC
ACB
BAC
BCA
CBA
CAB
```

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

void check(char [], int);

int main()
{
    char word[15];

    printf("Enter a word to check if it is a palindrome\n");
    scanf("%s", word);
    check(word, 0);

    return 0;
}

void check(char word[], int index)
{
    int len = strlen(word) - (index + 1);
    if (word[index] == word[len])
    {
        if (index + 1 == len || index == len)
        {
            printf("The entered word is a palindrome\n");
            return;
        }
        check(word, index + 1);
    }
    else
    {
        printf("The entered word is not a palindrome\n");
    }
}
```

```
Your INPUT go's here! Give only values. do not give like a=10
```

```
Enter a word to check if it is a palindrome
The entered word is not a palindrome
```

```c
#include <stdio.h>

int decimal_binary(int n)
{
    if (n==0)
        return 0;
    else
        return ((n%2)+10*decimal_binary(n/2));
}

void main()
{
   int no;

   printf("Enter a decimal number\n");
   scanf("%d",&no);
   printf("Decimal(%d) = Binary(%d)\n",no,decimal_binary(no));
}
```

```
3
```

```
Your OUTPUT go's here!
```

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

int Nth_of_GP(int a, int r, int n) {

    return( a * (int)(pow(r, n - 1)) );
}

int main() {

    int a = 1;
    int r = 2;
    int n = 8;
    printf("The %dth term of the series is: %d",n, Nth_of_GP(a, r, n) );
    return 0;
          }
```

```
3
```

```
The 8th term of the series is: 128
```

```c
#include<stdio.h>
int SumEven(int num1, int num2)
{
   if(num1>num2)
    return 0;
return num1+SumEven(num1+2,num2);
}
int main()
{
    int num1=2,num2;
    printf("Enter your Limit:\n");
    scanf("%d",&num2);
    printf("Sum of all Even numbers in the given range is: %d",SumEven(num1,num2));
}
```

```
3
```

```
Enter your Limit:
Sum of all Even numbers in the given range is: 30
```

```c
#include <stdio.h>
#include <string.h>
int main()
{
    char str[40];
    printf (" \n Enter a string to be reversed: ");
    scanf ("%s", str);

    printf (" \n After the reverse of a string: %s ", strrev(str));
    return 0;
}
```

```
25
```

```
Enter a string to be reversed:
After the reverse of a string:
```

```c
#include <stdio.h>

void towers(int, char, char, char);

int main()
{
    int num;

    printf("Enter the number of disks : ");
    scanf("%d", &num);
    printf("The sequence of moves involved in the Tower of Hanoi are :\n");
    towers(num, 'A', 'C', 'B');
    return 0;
}
void towers(int num, char frompeg, char topeg, char auxpeg)
{
    if (num == 1)
    {
        printf("\n Move disk 1 from peg %c to peg %c", frompeg, topeg);
        return;
    }
    towers(num - 1, frompeg, auxpeg, topeg);
    printf("\n Move disk %d from peg %c to peg %c", num, frompeg, topeg);
    towers(num - 1, auxpeg, topeg, frompeg);
}
```

```
3
```

```
Enter the number of disks : The sequence of moves involved in the
Tower of Hanoi are :

 Move disk 1 from peg A to peg C
 Move disk 2 from peg A to peg B
 Move disk 1 from peg C to peg B
 Move disk 3 from peg A to peg C
 Move disk 1 from peg B to peg A
 Move disk 2 from peg B to peg C
 Move disk 1 from peg A to peg C
```

```c
#include <stdio.h>

int addNumbers(int n);

int main() {

  int num;
  printf("Enter a positive integer: ");
  scanf("%d", &num);
  printf("Sum = %d", addNumbers(num));
  return 0;
}

int addNumbers(int n) {
  if (n != 0)
    return n + addNumbers(n - 1);
  else
    return n;
}
```

```
25
```

```
Enter a positive integer: Sum = 36
```

```c
#include <stdio.h>
int main()
{
  int n, s[1000], a = 1, d = 1, i;
  scanf("%d", &n);
  for (i = 0; i < n; i++)
    scanf("%d", &s[i]);
  i = 0;
  while ((a == 1 || d == 1) && i < n - 1) {
    if (s[i] < s[i+1])
      d = 0;
    else if (s[i] > s[i+1])
      a = 0;
    i++;
  }
  if (a == 1)
    printf("The array is sorted in ascending order.\n");
  else if (d == 1)
    printf("The array is sorted in descending order.\n");
  else
    printf("The array is not sorted.\n");

  return 0;
}
```

```
5
8
```

```
The array is not sorted.
```

```c
#include <stdio.h>

int main()
{
    printf("Studytonight - Best place to learn");
    char aa[100], bb[100];

    printf("\nEnter the first string: ");
    gets(aa);
    printf("\nEnter the second string to be concatenated: ");
    gets(bb);
    char *a = aa;
    char *b = bb;
    while(*a)
    {
        a++;
    }
    while(*b)
    {
        *a = *b;
        b++;
        a++;
    }
    *a = '\0';
    printf("The string after concatenation is: %s ", aa);
    printf("Coding is Fun !");
    return 0;
}
```

```
5
8
```

```
Studytonight - Best place to learn


Enter the first string:
Enter the second string to be concatenated:

The string after concatenation is: 8p◆8◆8

Coding is Fun !
```

```c
#include <stdio.h>
#define MAX_SIZE 100
void printArr(int *arr, int size);
int main()
{
    int arr[MAX_SIZE];
    int size;
    int *left = arr;
    int *right;
    printf("Enter size of array: ");
    scanf("%d", &size);
    right = &arr[size - 1];
    printf("Enter elements in array: ");
    while(left <= right)
    {
        scanf("%d", left++);
    }
    printf("\nArray before reverse: ");
    printArr(arr, size);
    left = arr;
    while(left < right)
    {
        *left  ^= *right;
        *right ^= *left;
        *left  ^= *right;
        left++;
        right--;
    }
    printf("\nArray after reverse: ");
    printArr(arr, size);
    return 0;
}
void printArr(int * arr, int size)
{
    int * arrEnd = (arr + size - 1);

    while(arr <= arrEnd)
    {
        printf("%d, ", *arr);

        arr++;
    }
}
```

```
5
8
```

```
Enter size of array: Enter elements in array:
Array before reverse: 1, 2, 3, 4, 5,
Array after reverse: 5, 4, 3, 2, 1,
```

```c
#include <stdio.h>
int main() {
    int a, b, temp;
    int *ptr1, *ptr2;
    printf("Enter the value of a and b: ");
    scanf("%d %d", &a, &b);
    printf("\nBefore swapping a = %d and b = %d", a, b);

    ptr1 = &a;
    ptr2 = &b;

    temp = *ptr1;
    *ptr1 = *ptr2;
    *ptr2 = temp;
    printf("\nAfter swapping a = %d and b = %d", a, b);
    return 0;
}
```

```
5
8
```

```
Enter the value of a and b:
Before swapping a = 5 and b = 8
After swapping a = 8 and b = 5
```

```c
#include <stdio.h>
#define ROW 2
#define COL 3
int sum(int(*array)[3]);
int main(void)
{
    int a[ROW][COL] = { {1 , 2, 3},
                        {4 , 5, 6} };
    printf(" sum = %d\n", sum (a));
    return 0;
}
int sum(int(*array)[3])
{
    int i,j, sum = 0;
    for (i =0; i < ROW ; i ++) {
        for (j =0; j < COL ; j ++) {
            sum = sum + *(*( array +i )+j);
        }
    }
}
```

```
Your INPUT go's here! Give only values. do not give like a=10
```

```
sum = 6
```

Run Save

```c
#include<stdio.h>
int main()
{
    char str[100];
    char *ptr=str;
    int count=0;
    printf("Enter the string:\n");
    gets(str);
    while(*ptr!='\0')
    {
        count++;
        ptr++;
    }
    printf("Length of string= %d",count);
    return 0;
}
```

Your INPUT go's here! Give only values. do not give like a=10

```
Enter the string:
Length of string= 3
```

```c
#include <stdio.h>
void sort(int n, int* ptr)
{
    int i, j, t;

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

        for (j = i + 1; j < n; j++) {

            if (*(ptr + j) < *(ptr + i)) {

                t = *(ptr + i);
                *(ptr + i) = *(ptr + j);
                *(ptr + j) = t;
            }
        }
    }
    for (i = 0; i < n; i++)
        printf("%d ", *(ptr + i));
}
int main()
{
    int n = 5;
    int arr[] = { 0, 23, 14, 12, 9 };
    sort(n, arr);
    return 0;
}
```

```
Your INPUT go's here! Give only values. do not give like a=10
```

```
0 9 12 14 23
```

```c
#include <stdio.h>
int power(int n1, int n2);
int main() {
    int base, a, result;
    printf("Enter base number: ");
    scanf("%d", &base);
    printf("Enter power number(positive integer): ");
    scanf("%d", &a);
    result = power(base, a);
    printf("%d^%d = %d", base, a, result);
    return 0;
}

int power(int base, int a) {
    if (a != 0)
        return (base * power(base, a - 1));
    else
        return 1;
}
```

```
25
```

```
Enter base number: Enter power number(positive integer): 8^9 =
134217728
```

```c
#include <stdio.h>
int nth_ap(int a, int d, int n) {
   return (a + (n - 1) * d);
}

int main() {

   int a = 2;
   int d = 1;
   int n = 5;
   printf("The %dth term of AP :%d", n, nth_ap(a,d,n));
   return 0;
         }
```

```
3
```

```
The 5th term of AP :6
```

```c
#include <stdio.h>
int sum_of_digit(int n)
{
    if (n == 0)
    return 0;
    return (n % 10 + sum_of_digit(n / 10));
}

int main()
{
    int num = 12345;
    int result = sum_of_digit(num);
    printf("Sum of digits in %d is %d\n", num, result);
    return 0;
}

```

```
3
```

```
Sum of digits in 12345 is 15
```

Run Save

```c
#include <stdio.h>
#define ARRAY_SIZE(a)  sizeof(a)/sizeof(a[0])
int main()
{
    int arr[] = { 1, 2, 3, 4, 5 };
    int evenNumCount = 0, oddNumCount = 0;
    int i;
    const int N = ARRAY_SIZE(arr);
    for( i = 0; i < N; i++)
    {
        if(arr[i] % 2 == 0)
        {
            evenNumCount++;
        }
        else
        {
            oddNumCount++;
        }
    }
    printf("Even elements = %d\n", evenNumCount);
    printf("Odd elements = %d", oddNumCount);
    return 0;
}
```

```
5
8
```

Your OUTPUT go's here!

```c
#include <stdio.h>

int main()
{
  long array[100], *maximum, size, c, location = 1;
  printf("Enter the number of elements in array\n");
  scanf("%ld", &size);
  printf("Enter %ld integers\n", size);
  for ( c = 0 ; c < size ; c++ )
    scanf("%ld", &array[c]);
  maximum  = array;
  *maximum = *array;
  for (c = 1; c < size; c++)
  {
    if (*(array+c) > *maximum)
    {
       *maximum = *(array+c);
       location = c+1;
    }
  }
  printf("Maximum element is present at location number %ld and it's value is %ld.\n", location, *maximum);
  return 0;
}
```

```
5
8
```

```
Your OUTPUT go's here!
```

```c
#include<stdio.h>
long int multiplyNumbers(int n);
int main() {
    int n;
    printf("Enter a positive integer: \n");
    scanf("%d",&n);
    printf("Factorial of %d = %ld", n, multiplyNumbers(n));
    return 0;
}

long int multiplyNumbers(int n) {
    if (n>=1)
        return n*multiplyNumbers(n-1);
    else
        return 1;
}

```

```
8
```

```
Enter a positive integer: Factorial of 8 = 40320
```