

ARRAYS

LEVEL 3

1.AVERAGE OF AN ARRAY

main.c	Run	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n=5; 5 int arr[n]; 6 for(int i=0;i<n;i++) 7 { 8 scanf("%d",&arr[i]); 9 } 10 int sum=0; 11 for(int i=0;i<n;i++) 12 { 13 sum=sum+arr[i]; 14 } 15 int avg=sum/n; 16 printf("%d",avg); 17 }</pre>		<pre>/tmp/tPqtmq1Ga0.o 5 8 9 6 3 6 === Code Execution Successful ===</pre>

2.FINDING MAXIMUM / LARGEST NUMBER

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int arr[100],n; 5 scanf("%d",&n); 6 for(int i=0;i<n;i++) 7 { 8 scanf("%d",&arr[i]); 9 } 10 int max=arr[0]; 11 for(int i=0;i<n;i++) 12 { 13 if(arr[i]>max) 14 max=arr[i]; 15 } 16 printf("%d",max); 17 }</pre>	<pre>/tmp/4aIGPwJFUM.o 5 2 52 86 489 21 489 === Code Execution Successful ===</pre>

3. PEAK ELEMENT

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int arr[100],n; 5 scanf("%d",&n); 6 for(int i=0;i<n;i++) 7 { 8 scanf("%d",&arr[i]); 9 } 10 if(n==1) 11 printf("%d ",arr[0]); 12 if(arr[0]>=arr[1]) 13 printf("%d ",arr[0]); 14 if(arr[n-1]>=arr[n-2]) 15 printf("%d ",arr[n-1]); 16 for(int i=1;i<n-1;i++) 17 { 18 if(arr[i]>=arr[i+1]&&arr[i]>=arr[i-1]) 19 printf("%d ",arr[i]); 20 } 21 } 22 23</pre>	<pre>/tmp/kpuxbE6u2f.o 6 7 6 5 6 7 9 7 9 === Code Execution Successful ===</pre>

4. LEFT ROTATION

main.c	Output
<pre>2 int main() 3 { 4 int arr[100],n; 5 scanf("%d",&n); 6 for(int i=0;i<n;i++) 7 { 8 scanf("%d",&arr[i]); 9 } 10 int k; 11 scanf("%d",&k); 12 for(int i=0;i<k;i++) 13 { 14 int temp=arr[0]; 15 for(int j=0;j<n-1;j++) 16 { 17 arr[j]=arr[j+1]; 18 } 19 arr[n-1]=temp; 20 } 21 22 for(int i=0;i<n;i++) 23 { 24 printf("%d",arr[i]); 25 } 26 } 27</pre>	<pre>/tmp/ecQ1C2nB2o.o 5 7 8 9 6 3 1 89637 === Code Execution Successful ===</pre>

5.RIGHT ROTATION

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int arr[100],n; 5 scanf("%d",&n); 6 for(int i=0;i<n;i++) 7 { 8 scanf("%d",&arr[i]); 9 } 10 int k; 11 scanf("%d",&k); 12 for(int i=0;i<k;i++) 13 { 14 int temp=arr[n-1]; 15 for(int j=n-1;j>0;j--) 16 { 17 arr[j]=arr[j-1]; 18 } 19 arr[0]=temp; 20 } 21 22 for(int i=0;i<n;i++) 23 { 24 printf("%d",arr[i]); 25 } 26 }</pre>	<pre>/tmp/HyWknEMc4L.o 5 7 4 1 2 5 1 57412 === Code Execution Successful ===</pre>

6.NUMBER OF OCCURENCE OF A NUMBER

C Online Compiler

```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr[n];
7     for(int i=0;i<n;i++)
8     {
9         scanf("%d",&arr[i]);
10    }
11    int num;
12    scanf("%d",&num);
13    int count=0;
14    for(int i=0;i<n;i++)
15    {
16        if(num==arr[i])
17            count++;
18    }
19    printf("%d",count);
20 }
```

Output

```
/tmp/xbTNUFobhJ.o
8
4 8 9 6 8 5 7 6
8
2

=== Code Execution Successful ===
```

7.SUM AND PRODUCT OF A NUMBER

```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr[n];
7     for(int i=0;i<n;i++)
8     {
9         scanf("%d",&arr[i]);
10    }
11    int sum=0;
12    int product=1;
13    for(int i=0;i<n;i++)
14    {
15        sum=sum+arr[i];
16    }
17    for(int i=0;i<n;i++)
18    {
19        product=product*arr[i];
20    }
21    printf("%d",sum);
22    printf("\n%d",product);
23 }
```

Output

```
/tmp/kTxdMjA2Po.o
6
4 8 5 2 3 1
23
960

=== Code Execution Successful ===
```

8.SQUARE OF ARRAY ELEMENTS

main.c		Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int sqr; 12 for(int i=0;i<n;i++) 13 { 14 arr[i]=arr[i]*arr[i]; 15 } 16 for(int i=0;i<n;i++) 17 { 18 printf("%d ",arr[i]); 19 } 20 }</pre>	<pre>/tmp/tFdSwCf0iF.o 5 4 2 6 3 2 16 4 36 9 4 === Code Execution Successful ===</pre>	

9.DIFFERENCE BETWEEN MAXIMUM AND MINIMUM NUMBER

main.c		Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int max=arr[0]; 12 int min=arr[0]; 13 for(int i=0;i<n;i++) 14 { 15 if(arr[i]>max) 16 max=arr[i]; 17 else if(arr[i]<min) 18 min=arr[i]; 19 } 20 printf("%d %d\n",max,min); 21 int diff=max-min; 22 printf("%d",diff); 23 }</pre>	<pre>/tmp/nzENzMywc6.o 5 78 96 52 41 23 96 23 73 === Code Execution Successful ===</pre>	

10.NO OF ELEMENTS DIVISIBLE BY A NUMBER

main.c	Run	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int count=0; 12 int num; 13 scanf("%d",&num); 14 for(int i=0;i<n;i++) 15 { 16 if(arr[i]%num==0) 17 count++; 18 } 19 printf("%d",count); 20 }</pre>		<pre>/tmp/RRwk6tyT7v.o 5 7 8 9 6 5 2 2 === Code Execution Su</pre>

11.REMOVING DUPLICATE ELEMENTS

main.c	Run	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int count; 12 for(int i=0;i<n;i++) 13 { 14 count=0; 15 for(int j=i+1;j<n;j++) 16 { 17 if(arr[i]==arr[j]) 18 count++; 19 } 20 if(count==0) 21 { 22 printf("%d ",arr[i]); 23 } 24 } 25 } 26 }</pre>		<pre>/tmp/NFs3GLvWub.o 5 4 2 5 4 3 2 5 4 3 === Code Execution Successful ===</pre>

12. NEGATIVE ELEMENTS

main.c	Run	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 for(int i=0;i<n;i++) 12 { 13 if(arr[i]<0) 14 { 15 printf("%d",arr[i]); 16 } 17 } 18 }</pre>		<pre>/tmp/ROVzh8eT7o.o 6 7 8 -5 -6 2 4 -5-6 === Code Execution Su</pre>

13.POSITIVE ELEMENTS

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 for(int i=0;i<n;i++) 12 { 13 if(arr[i]>0) 14 { 15 printf("%d",arr[i]); 16 } 17 } 18 }</pre>	<pre>/tmp/d68XzN7sw0.o 6 7 8 5 -2 -3 6 7856 === Code Execution Successful ===</pre>

14.DELETING AN ELEMENT AT A POSITION IN A ARRAY

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int pos; 12 scanf("%d",&pos); 13 for(int i=pos-1;i<n;i++) 14 { 15 arr[i]=arr[i+1]; 16 } 17 for(int i=0;i<n-1;i++) 18 { 19 printf("%d ",arr[i]); 20 } 21 }</pre>	<pre>/tmp/0fTULNBvd6.o 5 7 9 6 3 2 3 7 9 3 2 === Code Execution Successful ===</pre>

15. SUM OF DUPLICATE ELEMENTS

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n,count; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int sum=0; 12 for(int i=0;i<n;i++) 13 { 14 count=0; 15 for(int j=i+1;j<n;j++) 16 { 17 if(arr[i]==arr[j]) 18 count++; 19 } 20 } 21 if(count!=0) 22 { 23 printf("%d",arr[i]); 24 sum=sum+arr[i]; 25 } 26 } 27 printf("\n%d",sum); 28 }</pre>	<pre>/tmp/5PCvNf04fn.o 6 7 9 6 8 7 9 79 16 === Code Execution Successful ===</pre>

16. NON PRIME NUMBERS

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n,count; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int prime; 12 for(int i=0;i<n;i++) 13 { 14 prime=0; 15 for(int j=2;j<arr[i];j++) 16 { 17 if(arr[i]%j==0) 18 prime=1; 19 } 20 if(prime==1) 21 printf("%d ",arr[i]); 22 } 23 }</pre>	<pre>/tmp/hF3wiAIHRR.o 8 7 6 3 22 5 86 20 13 6 22 86 20 === Code Execution Successful ===</pre>

17.SUM OF 1ST ,2ND , LAST AND LAST BEFORE NUMBER

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n,count; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int sum=arr[0]+arr[1]+arr[n-1]+arr[n-2]; 12 printf("%d",sum); 13 }</pre>	<pre>/tmp/yqlj88n1MZ.o 5 7 8 9 6 3 24 === Code Execution Successful ===</pre>

18.PRINTING UNIQUE NUMBERS

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n,count; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 12 for(int i=0;i<n;i++) 13 { 14 count =0; 15 for(int j=0;j<n;j++) 16 { 17 if(arr[i]==arr[j]&& i!=j) 18 count=1; 19 } 20 if(count==0) 21 printf("%d ",arr[i]); 22 } 23 }</pre>	<pre>/tmp/ZUYZQ3q8xp.o 8 7 6 3 8 9 7 3 5 6 8 9 5 === Code Execution Successful ===</pre>

19. Second largest number



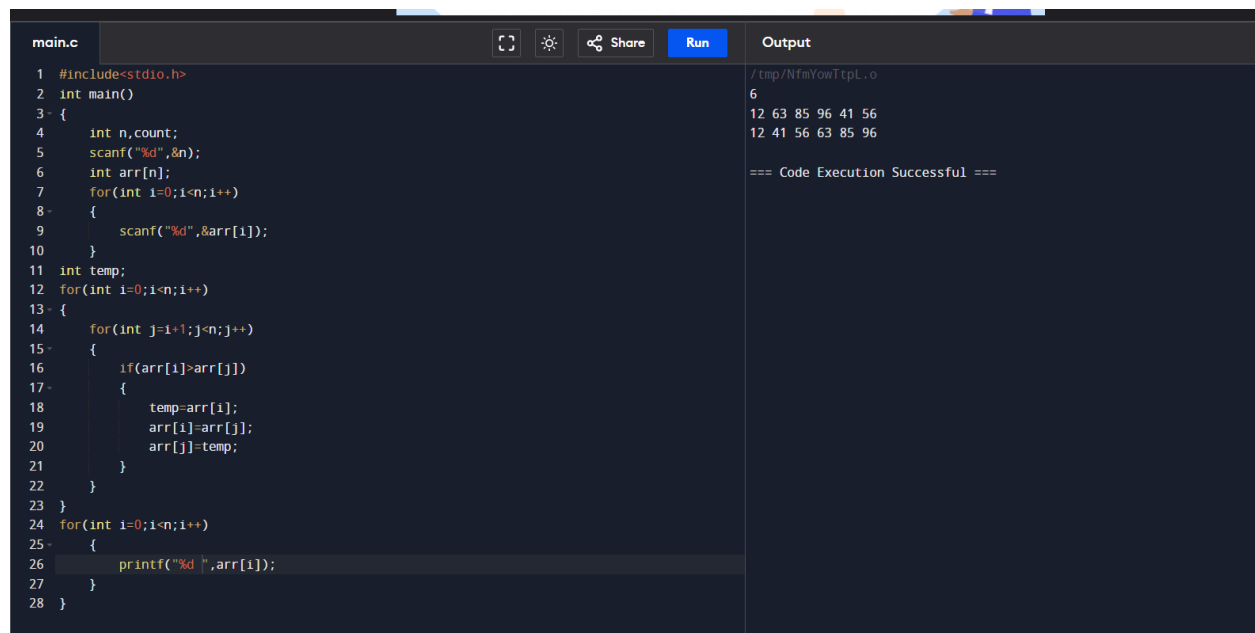
The screenshot shows the Programiz C Online Compiler interface. The code in `main.c` is as follows:

```
1 #include<stdio.h>
2 int main()
3 {
4     int n, count;
5     scanf("%d", &n);
6     int arr[n];
7     for(int i=0; i<n; i++)
8     {
9         scanf("%d", &arr[i]);
10    }
11
12    int largest=arr[0];
13    int secondlargest=arr[0];
14    for(int i=0; i<n; i++)
15    {
16        if(arr[i]>largest)
17        {
18            secondlargest=largest;
19            largest=arr[i];
20        }
21        else if(arr[i]>secondlargest && arr[i]<largest)
22        {
23            secondlargest=arr[i];
24        }
25    }
26    printf("%d", secondlargest);
27 }
28
```

The output on the right shows the execution results:

```
/tmp/o114L32tMx.o
6
77 96 85 12 425 525
425
=== Code Execution Successful ===
```

20. ASCENDING ORDER



The screenshot shows the Programiz C Online Compiler interface. The code in `main.c` is as follows:

```
1 #include<stdio.h>
2 int main()
3 {
4     int n, count;
5     scanf("%d", &n);
6     int arr[n];
7     for(int i=0; i<n; i++)
8     {
9         scanf("%d", &arr[i]);
10    }
11    int temp;
12    for(int i=0; i<n; i++)
13    {
14        for(int j=i+1; j<n; j++)
15        {
16            if(arr[i]>arr[j])
17            {
18                temp=arr[i];
19                arr[i]=arr[j];
20                arr[j]=temp;
21            }
22        }
23    }
24    for(int i=0; i<n; i++)
25    {
26        printf("%d ", arr[i]);
27    }
28 }
```

The output on the right shows the execution results:

```
/tmp/NfmYowTtpL.o
6
12 63 85 96 41 56
12 41 56 63 85 96
=== Code Execution Successful ===
```

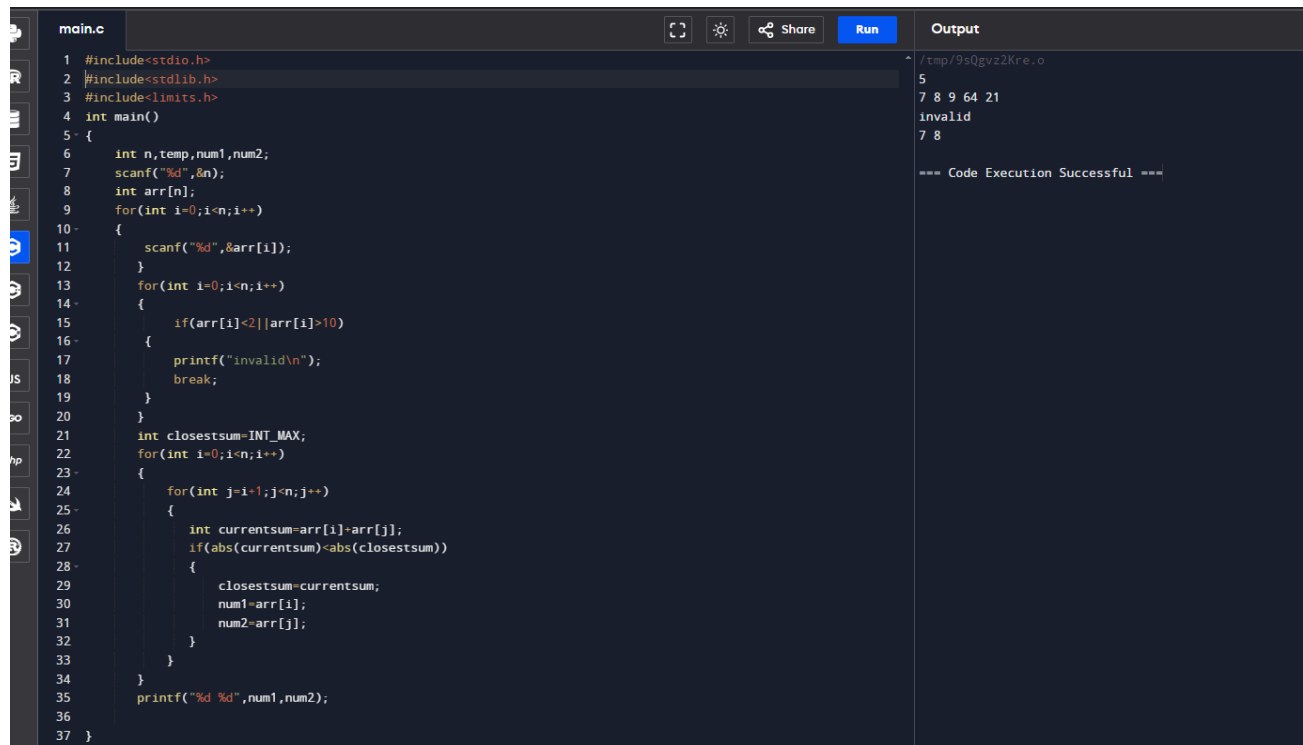
21.REMOVING EVEN NUMBERS

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int arr[10],n,i,out,in,temp; 5 scanf("%d",&n); 6 for(i=0;i<n;i++) 7 { 8 scanf("%d",&arr[i]); 9 } 10 11 for(int i=0;i<n;i++) 12 { 13 if(arr[i]%2!=0) 14 { 15 printf("%d",arr[i]); 16 } 17 } 18 }</pre>	<pre>/tmp/PmQtb6AMq1.o 5 7 8 9 6 2 79 === Code Execution Successful ===</pre>

22.REMOVING FIRST OCCURRENCE OF A NUMBER

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int arr[10],n,i,out,in,temp; 5 scanf("%d",&n); 6 for(i=0;i<n;i++) 7 { 8 scanf("%d",&arr[i]); 9 } 10 int num; 11 scanf("%d",&num); 12 int found=0; 13 for(int i=0;i<n;i++) 14 { 15 if(!found&&arr[i]==num) 16 { 17 for(int j=1;j<n-1;j++) 18 { 19 arr[j]=arr[j+1]; 20 } 21 break; 22 } 23 } 24 for(int i=0;i<n-1;i++) 25 { 26 printf("%d",arr[i]); 27 } 28 }</pre>	<pre>/tmp/GKFeMzkPzZ.o 6 7 8 9 7 5 2 7 89752 === Code Execution Successful ===</pre>

23. Array Input range 2-10 if exceeds print "Invalid". print two elements which is the sum of two elements closest to 0(zero). Input: [-1,-10,8,2] output: [-1,2]. Explanation -1+2=-1 is the closest to 0 than other combinations.



```
main.c
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<limits.h>
4 int main()
5 {
6     int n,temp,num1,num2;
7     scanf("%d",&n);
8     int arr[n];
9     for(int i=0;i<n;i++)
10    {
11        scanf("%d",&arr[i]);
12    }
13    for(int i=0;i<n;i++)
14    {
15        if(arr[i]<2||arr[i]>10)
16        {
17            printf("invalid\n");
18            break;
19        }
20    }
21    int closestsum=INT_MAX;
22    for(int i=0;i<n;i++)
23    {
24        for(int j=i+1;j<n;j++)
25        {
26            int currentsum=arr[i]+arr[j];
27            if(abs(currentsum)<abs(closestsum))
28            {
29                closestsum=currentsum;
30                num1=arr[i];
31                num2=arr[j];
32            }
33        }
34    }
35    printf("%d %d",num1,num2);
36 }
37 }
```

Output

```
/tmp/9sQgvz2Kre.o
5
7 8 9 64 21
invalid
7 8
=== Code Execution Successful ===
```

2

24.MEDIAN

```
#include<stdio.h>
int main()
{
    int n,temp,num1,num2;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
    }
    for(int i=0;i<n;i++)
    {
        for(int j=i+1;j<n;j++)
        {
            if(arr[i]>arr[j])
            {
                temp=arr[i];
                arr[i]=arr[j];
                arr[j]=temp;
            }
        }
    }
    for(int i=0;i<n;i++)
    {
        printf("%d ",arr[i]);
    }
    double median;
    if(n%2==0)
    {
        median=(arr[n/2]+arr[n/2-1])/2;
    }

    if(n%2!=0)
    {
        median=arr[n/2];
    }
    printf("\n%f",median);
}
```

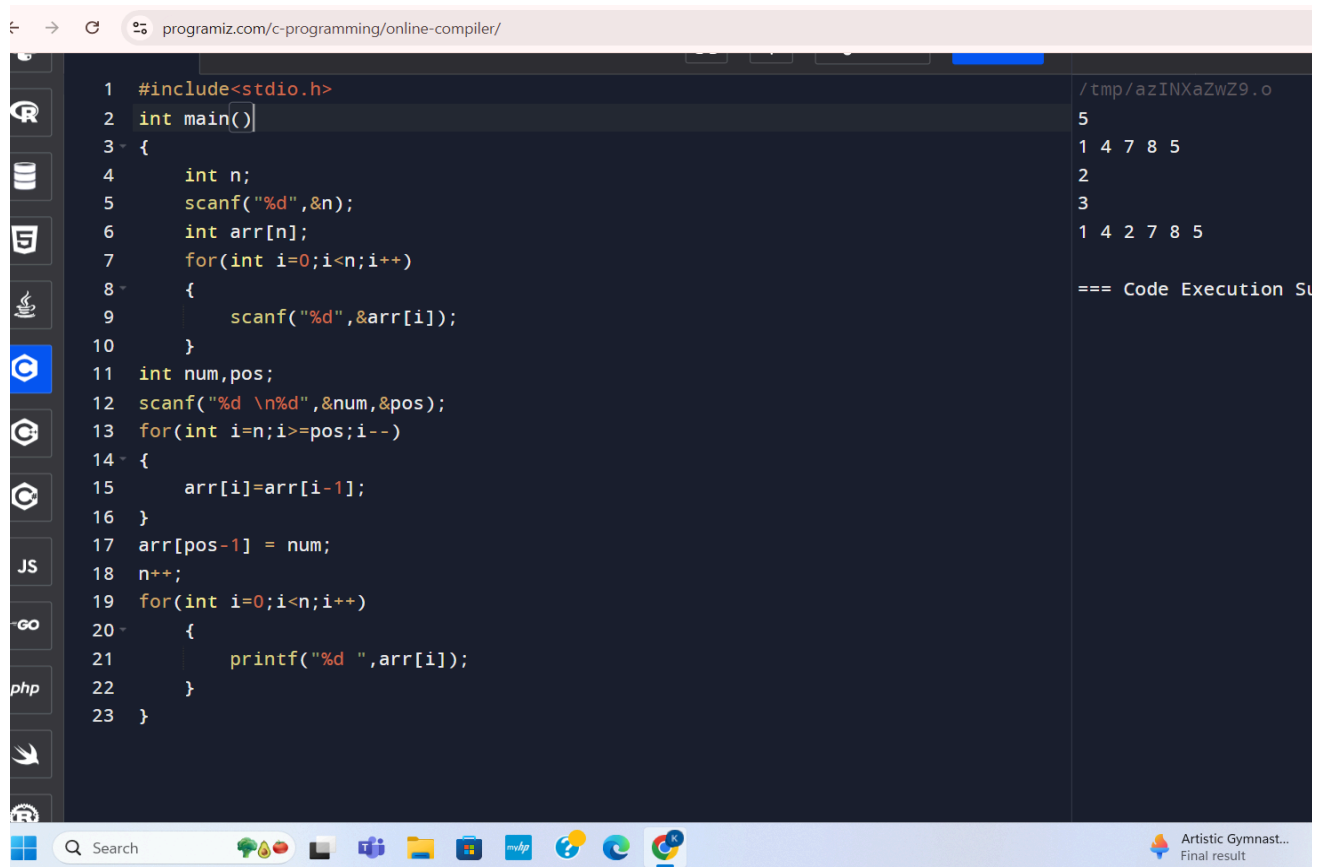
/tmp/re8p003h9v.o
6
7 8 9 64 21 32
7 8 9 21 32 64
15.000000
=== Code Execution Successful ===

25.CHUNKSIZE

```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n,chunksize;
5     scanf("%d",&n);
6     int arr[n];
7     for(int i=0;i<n;i++)
8     {
9         scanf("%d",&arr[i]);
10    }
11    scanf("%d",&chunksize);
12    for(int i=0;i<n;i+=chunksize)
13    {
14        printf("[");
15        for(int j=i;j<i+chunksize && j<n;j++)
16        {
17            printf("%d",arr[j]);
18            if(j<i+chunksize-1 && j<n-1)
19            {
20                printf(",");
21            }
22        }
23        printf("]");
24    }
25 }
```

/tmp/tKT15aWKmL.o
5
10 52 85 63 41
2
[10,52][85,63][41]
=== Code Execution Successful ===

26.INSERTING ELEMENT AT PARTICULAR POSITION



The screenshot shows a web browser window with the URL `programiz.com/c-programming/online-compiler/`. The main area contains a C program with 23 lines of code. The code defines a `main` function that takes an integer `n` and reads `n` integers into an array `arr`. It then reads a number `num` and a position `pos`, and shifts the elements from `pos` to the end of the array one position to the right. Finally, it prints the entire array. On the right side, the output of the program is shown, including the file path `/tmp/azINXaZwZ9.o`, the number `5`, the array `1 4 7 8 5`, the number `2`, the number `3`, and the modified array `1 4 2 7 8 5`. Below the output, it says `=== Code Execution Su`. The bottom of the browser window shows a Windows taskbar with various application icons.

```
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr[n];
7     for(int i=0;i<n;i++)
8     {
9         scanf("%d",&arr[i]);
10    }
11    int num,pos;
12    scanf("%d \n%d",&num,&pos);
13    for(int i=n;i>=pos;i--)
14    {
15        arr[i]=arr[i-1];
16    }
17    arr[pos-1] = num;
18    n++;
19    for(int i=0;i<n;i++)
20    {
21        printf("%d ",arr[i]);
22    }
23 }
```

/tmp/azINXaZwZ9.o
5
1 4 7 8 5
2
3
1 4 2 7 8 5
=== Code Execution Su

27.FIND FREQUENCY OF ELEMENTS

```
main.c  [Icons] [Share] [Run] Output
1  #include<stdio.h>
2  int main()
3  {
4      int n,count;
5      scanf("%d",&n);
6      int arr[n];
7      int freq[n];
8      for(int i=0;i<n;i++)
9      {
10         scanf("%d",&arr[i]);
11         freq[i]=1;
12     }
13     for(int i=0;i<n;i++)
14     {
15         count=1;
16         for(int j=i+1;j<n;j++)
17         {
18             if(arr[i]==arr[j])
19             {
20                 count++;
21                 freq[j]=0;
22             }
23         }
24         if(freq[i]!=0)
25             freq[i]=count;
26     }
27     for(int i=0;i<n;i++)
28     {
29         if(freq[i]!=0)
30         {
31             printf("%d %d\n",arr[i],freq[i]);
32         }
33     }
34 }
```

/tmp/lyHEapke1p.o
6
1 5 2 1 4 2
1 2
5 1
2 2
4 1

=== Code Execution Successful ===

28.DELETING DUPLICATE ELEMENTS(PRINTING 1ST AND DELETING REST ALL)

main.c		Run	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int count; 12 for(int i=0;i<n;i++) 13 { 14 for(int j=i+1;j<n;j++) 15 { 16 if(arr[i]==arr[j]) 17 { 18 for(int k=j;k<n-1;k++) 19 { 20 arr[k]=arr[k+1]; 21 } 22 n--; 23 j--; 24 } 25 } 26 } 27 } 28 } 29 for(int i=0;i<n;i++) 30 { 31 printf("%d ",arr[i]); 32 } 33 } 34 }</pre>		Run	<pre>/tmp/IbwaWD1ZWG.o 10 10 20 10 1 100 10 2 1 5 10 10 20 1 100 2 5 === Code Execution Successful ===</pre>

29.MERGE 2 ARRAYS AND ASCENDING ORDER

main.c	Run	Output
<pre> 1 #include<stdio.h> 2 int main() 3 { 4 int n1,n2,n3,temp; 5 scanf("%d %d",&n1,&n2); 6 n3=n1+n2; 7 int arr1[n1],arr2[n2],arr3[n3]; 8 for(int i=0;i<n1;i++) 9 { 10 scanf("%d",&arr1[i]); 11 } 12 for(int i=0;i<n2;i++) 13 { 14 scanf("%d",&arr2[i]); 15 } 16 for(int i=0;i<n1;i++) 17 { 18 arr3[i]=arr1[i]; 19 } 20 21 for(int i=0,j=n1;j<n3 && i<n2;i++,j++) 22 { 23 arr3[j]=arr2[i]; 24 } 25 for(int i=0;i<n3;i++) 26 { 27 for(int j=i+1;j<n3;j++) 28 { 29 if(arr3[i]>arr3[j]) 30 { 31 temp=arr3[i]; 32 arr3[i]=arr3[j]; 33 arr3[j]=temp; 34 } 35 } 36 } 37 38 for(int i=0;i<n3;i++) 39 { 40 printf("%d ",arr3[i]); 41 } 42 } </pre>	Run	<pre> /tmp/4ApsspGx.c.o 5 4 7 8 9 6 22 14 25 36 78 6 7 8 9 14 22 25 36 78 === Code Execution Successful === </pre>

30.REVERSE ARRAY ELEMENTS

main.c	Run	Output
<pre> 1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 for(int i=n-1;i>=0;i--) 12 { 13 printf("%d ",arr[i]); 14 } 15 } </pre>	Run	<pre> /tmp/WEGwGNBJbF.o 5 78 52 63 14 12 12 14 63 52 78 === Code Execution Successful === </pre>

31.PRINTING EVEN AND ODD NUMBERS IN SEPARATE 2 ARRAYS

main.c		Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n],evenarr[n],oddarr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int even=0,odd=0; 12 for(int i=0;i<n;i++) 13 { 14 if(arr[i]%2==0) 15 { 16 evenarr[even]=arr[i]; 17 even++; 18 } 19 else 20 { 21 oddarr[odd]=arr[i]; 22 odd++; 23 } 24 } 25 for(int i=0;i<even;i++) 26 { 27 printf("%d ",evenarr[i]); 28 } 29 printf("\n"); 30 for(int i=0;i<odd;i++) 31 { 32 printf("%d ",oddarr[i]); 33 } 34 }</pre>	<pre>/tmp/yNM805MNp2.o 5 11 25 36 45 88 36 88 11 25 45 === Code Execution Successful ===</pre>	

32.SEARCHING AN ELEMENT AND PRINTING ITS POSITION

```
1  #include<stdio.h>
2  int main()
3  {
4      int n;
5      scanf("%d",&n);
6      int arr[n],evenarr[n],oddarr[n];
7      for(int i=0;i<n;i++)
8      {
9          scanf("%d",&arr[i]);
10     }
11     int num;
12     int count=0;
13     scanf("%d",&num);
14     for(int i=0;i<n;i++)
15     {
16         if(arr[i]==num)
17         {
18             count=1;
19         }
20         if(count==1)
21         {
22             printf("%d",i+1);
23             break;
24         }
25     }
26 }
```

/tmp/PUNQCtZMuJ.o
6
11 22 33 44 55 66
55
5
=== Code Execution Success

33.

```

main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr[n],evenarr[n],oddarr[n];
7     for(int i=0;i<n;i++)
8     {
9         scanf("%d",&arr[i]);
10    }
11    //separating even and odd numbers in 2 different arrays
12
13    int even=0,odd=0;
14    for(int i=0;i<n;i++)
15    {
16        if(arr[i]%2==0)
17        {
18            evenarr[even]=arr[i];
19            even++;
20        }
21        else
22        {
23            oddarr[odd]=arr[i];
24            odd++;
25        }
26    }
27    //merging the 2 arrays in even and odd basis respectively
28    int n3;
29    n3=odd+even;
30    int mergearr[n3];
31
32    for(int i=0;i<even;i++)
33    {
34        mergearr[i]=evenarr[i];

```

/tmp/wz4zmvCXS.o

```

6
11 22 55 66 33 44
22 66 44 11 55 33
11 22 33 44 55 66

```

=== Code Execution Successful ===

```

main.c
33 {
34     mergearr[i]=evenarr[i];
35 }
36 for(int i=0,j=even,i<odd;i++,j++)
37 {
38     mergearr[j]=oddarr[i];
39 }
40
41 for(int i=0;i<n3;i++)
42 {
43     printf("%d ",mergearr[i]);
44 }
45 printf("\n");
46
47 //sorting the merged array in ascending order
48
49 int temp;
50 for(int i=0;i<n3;i++)
51 {
52     for(int j=i+1;j<n3;j++)
53     {
54         if(mergearr[i]>mergearr[j])
55         {
56             temp=mergearr[i];
57             mergearr[i]=mergearr[j];
58             mergearr[j]=temp;
59         }
60     }
61 }
62 for(int i=0;i<n3;i++)
63 {
64     printf("%d ",mergearr[i]);
65 }
66 }

```

/tmp/wz4zmvCXS.o

```

6
11 22 55 66 33 44
22 66 44 11 55 33
11 22 33 44 55 66

```

=== Code Execution Successful ===

34. Write a program in C to find a pair with a given sum in the array.

```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr[n],evenarr[n],oddarr[n];
7     for(int i=0;i<n;i++)
8     {
9         scanf("%d",&arr[i]);
10    }
11    int num,sum;
12    scanf("%d",&num);
13    int count=0;
14    for(int i=0;i<n;i++)
15    {
16        for(int j=i+1;j<n;j++)
17        {
18            sum=arr[i]+arr[j];
19            if(sum==num)
20            {
21                printf("%d %d",i,j);
22                count=1;
23                break;
24            }
25        }
26        if(count)
27            break;
28    }
29    if(count==0)
30    {
31        printf("NOT FOUND");
32    }
33 }
34
```

Output

```
/tmp/2zUmeL906e.o
5
14 12 36 52 15
64
1 3

=== Code Execution Successful ===
```

35.removing 1st occurrence of a number

```
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr[n];
7     for(int i=0;i<n;i++)
8     {
9         scanf("%d",&arr[i]);
10    }
11    int num;
12    scanf("%d",&num);
13
14    for(int i=0;i<n;i++)
15    {
16        if( arr[i]==num)
17        {
18            for(int j=1;j<n-1;j++)
19            {
20                arr[j]=arr[j+1];
21            }
22
23            break;
24        }
25    }
26    for(int i=0;i<n-1;i++)
27    {
28        printf("%d",arr[i]);
29    }
30 }
```

Output

```
/tmp/wgHtL6mZymQ.o
6
1 4 8 5 1 4
4
18514

=== Code Execution Successful ===
```

36.SUM OF NEIGHBOR PEAK ELEMENTS

main.c	Run	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[100]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 for(int i=0;i<n;i++) 12 { 13 if(arr[i]>=arr[i+1] && arr[i]>=arr[i-1]) 14 { 15 arr[i]+=arr[i+1]+arr[i-1]; 16 } 17 } 18 for(int i=0;i<n;i++) 19 { 20 printf("%d ",arr[i]); 21 } 22 } 23 }</pre>		<pre>/tmp/lfx63eQWsu.o 6 7 4 5 2 1 8 11 4 11 2 1 9 === Code Execution Successful</pre>

37. Swapping TWO elements:

main.c	Run	Output
<pre>1 2 #include<stdio.h> 3 int main(){ 4 int a[100],n,i,k1,k2; 5 scanf("%d",&n); 6 for(i=0;i<n;i++){ 7 scanf("%d",&a[i]); 8 } 9 scanf("%d",&k1); 10 scanf("%d",&k2); 11 if(k1>= 0 && k1 < n && k2 >= 0 && k2 < n){ 12 int temp=a[k1]; 13 a[k1]=a[k2]; 14 a[k2]=temp; 15 } 16 for(i=0;i<n;i++){ 17 printf("%d ",a[i]); 18 } 19 }</pre>		<pre>/tmp/IcsxLkLFF1.o 5 1 2 3 4 5 3 2 1 2 4 3 5 === Code Execution Successful ===</pre>

38. Sorted or NOT Sorted:

```
#include<stdio.h>

int main(){
    int a[100], i, j, n;
    scanf("%d", &n);
    for(i=0; i<n;i++){
        scanf("%d", &a[i]);
    }
    int as=1;
    for(j=0;j<n-1;j++){
        if(a[j]>a[j+1]){
            as=0;
            break;
        }
    }
    int ds=1;
    for(j=0;j<n-1;j++){
        if(a[j]<a[j+1]){
            ds=0;
            break;
        }
    }
    if(as==1 || ds==1){
        printf("array is sorted");
    }
    else{
        printf("array is not sorted");
    }
}
```

/tmp/h1rgMTChKD.o
6
11 12 13 14 15 16
array is sorted
=== Code Execution Successful ===

39. Sort the array in ascending order and print even numbers first and odd numbers next


```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr[n];
7     int even=0;
8     int evenarr[n];
9     for(int i=0;i<n;i++)
10     {
11         scanf("%d",&arr[i]);
12     }
13     for(int i=0;i<n;i++)
14     {
15         for(int j=i+1;j<n;j++)
16         {
17             if(arr[i]>arr[j])
18             {
19                 int temp=arr[i];
20                 arr[i]=arr[j];
21                 arr[j]=temp;
22             }
23         }
24     }
25     for(int i=0;i<n;i++)
26     {
27         if(arr[i]%2==0)
28         {
29             evenarr[even]= arr[i];
30             even++;
31         }
32     }
33     for(int i=0;i<n;i++)
34     {
35         if(arr[i]%2!=0)
36         {
37             evenarr[even]=arr[i];
38             even++;
39         }
40     }
41     for(int i=0;i<even;i++)
42     {
43         printf("%d",evenarr[i]);
44     }
45 }
46
```

Output

//tmp/nJUBaP8e1g
6
1 7 8 5 9 6
681579
=== Code Executi

40.Removing the occurrence of the digit is :

main.c	Run	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 int even=0; 8 int evenarr[n]; 9 for(int i=0;i<n;i++) 10 { 11 scanf("%d",&arr[i]); 12 } 13 int num; 14 scanf("%d",&num); 15 for(int i=0;i<n;i++) 16 { 17 if(arr[i]==num) 18 { 19 for(int j=i;j<n-1;j++) 20 { 21 arr[j]=arr[j+1]; 22 } 23 n--; 24 } 25 } 26 for(int i=0;i<n;i++) 27 { 28 printf("%d ",arr[i]); 29 } 30 }</pre>	Run	<pre>/tmp/02dkRYdhD6.o 6 7 8 5 2 5 1 5 7 8 2 1 === Code Execution</pre>

41. Product of an array without multiplying the present index :

```
main.c  [Icons] [Share] [Run] Output
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr[n];
7     for(int i=0;i<n;i++)
8     {
9         scanf("%d",&arr[i]);
10    }
11    int temparr[n];
12    int p;
13    for(int i=0;i<n;i++)
14    {
15        p=1;
16        for(int j=0;j<n;j++)
17        {
18            if(i!=j)
19            {
20                p=p*arr[j];
21            }
22        }
23        temparr[i]=p;
24    }
25    for(int i=0;i<n;i++)
26    {
27        printf("%d ",temparr[i]);
28    }
29 }
30 }
```

/tmp/pMQItB2RsQ.o
5
1 2 3 4 5
120 60 40 30 24
=== Code Execution Successful ===

42. How Many Numbers Are Smaller Than the Current Number
Input: nums=[8,1,2,2,3] Output:[4,0,1,1,3]

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int temp[n]; 12 for(int i=0;i<n;i++) 13 { 14 int count=0; 15 for(int j=0;j<n;j++) 16 { 17 if(arr[i]>arr[j] && i!=j) 18 { 19 count++; 20 } 21 } 22 temp[i]=count; 23 } 24 for(int i=0;i<n;i++) 25 { 26 printf("%d ",temp[i]); 27 } 28 } 29</pre>	<pre>/tmp/VlKtdfYt1b.o 6 7 8 5 6 2 1 4 5 2 3 1 0 === Code Execution Successful ===</pre>

43.RANKING

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int rank[n]; 12 int temp; 13 for(int i=0;i<n;i++) 14 { 15 temp=1; 16 for(int j=0;j<n;j++) 17 { 18 if(arr[i]>arr[j]) 19 { 20 temp++; 21 } 22 } 23 rank[i]=temp; 24 } 25 for(int i=0;i<n;i++) 26 { 27 printf("%d ",rank[i]); 28 } 29 } 30</pre>	<pre>/tmp/8hmXXHrfJD.o 5 52 41 63 95 35 3 2 4 5 1 === Code Execution Successful ===</pre>

44.REMOVING EVEN NUMBER IN EVEN INDEX

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int a[100],i,j,n; 5 printf("enter n : "); 6 scanf("%d",&n); 7 printf("enter the elements : "); 8 for(i=0;i<n;i++) 9 { 10 scanf("%d",&a[i]); 11 } 12 for(i=2;i<n;i=i+2) 13 { 14 if(a[i]%2==0) 15 { 16 for(j=1;j<n;j++) 17 { 18 a[j]=a[j+1]; 19 } 20 n--; 21 i--; 22 } 23 } 24 for(i=0;i<n;i++) 25 { 26 printf("%d ",a[i]); 27 } 28 } 29</pre>	<pre>/tmp/BpmvkUoT6E.o enter n : 6 enter the elements : 4 9 8 6 2 14 9 8 6 2 1 4 9 6 1 === Code Execution Successful ===</pre>

45.REMOVING THE ELEMENTS GREATER THAN THE NUMBER

C Online Compiler

main.c

Share

Run

```
1  #include<stdio.h>
2  int main()
3  {
4      int n;
5      scanf("%d",&n);
6      int arr[n];
7      for(int i=0;i<n;i++)
8      {
9          scanf("%d",&arr[i]);
10     }
11     int num;
12     scanf("%d",&num);
13     for(int i=0;i<n;i++)
14     {
15         if(arr[i]<num)
16         {
17             printf("%d ",arr[i]);
18         }
19     }
20 }
```

/tmp/6jYKvooDgr.o

6

25 36 14 85 96 45

40

25 36 14

=== Code Execution S

46. Last repeated element :

main.c		Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 int rev; 8 int count=0; 9 for(int i=0;i<n;i++) 10 { 11 scanf("%d",&arr[i]); 12 } 13 for(int i=0;i<n;i++) 14 { 15 for(int j=n-1;j>=0;j--) 16 { 17 if(arr[i]==arr[j] && i!=j) 18 { 19 rev=arr[i]; 20 count=1; 21 } 22 } 23 } 24 if(count ==0) 25 { 26 printf("no elements duplicate"); 27 } 28 else 29 printf("%d",rev); 30 31 } 32</pre>	<pre>/tmp/NtcztkX0FR.. 8 7 5 3 6 2 5 7 8 7 === Code Executi</pre>	

47. Find the element in an array that are greater than all elements to their right and print them in ascending order sample input : 5 arr: 5 3 20 15 8 output: 8 15 20

main.c	Run	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 int rev; 8 int count=0; 9 for(int i=0;i<n;i++) 10 { 11 scanf("%d",&arr[i]); 12 } 13 int r; 14 r=arr[n-1]; 15 printf("%d ",r); 16 for(int i=n-2;i>=0;i--) 17 { 18 if(arr[i]>r) 19 { 20 printf("%d ",arr[i]); 21 r=arr[i]; 22 } 23 } 24 }</pre>		<pre>/tmp/wbLnw4a6tg.o 6 56 45 36 98 75 12 12 75 98 === Code Execution Successful ===</pre>

48.MAX sum of array elements

main.c	Run	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr[n]; 7 for(int i=0;i<n;i++) 8 { 9 scanf("%d",&arr[i]); 10 } 11 int max=0; 12 for(int i=0;i<n;i++) 13 { 14 for(int j=i;j<n;j++) 15 { 16 int sum=0; 17 for(int k=i;k<n;k++) 18 { 19 sum=sum+arr[k]; 20 if(max<sum) 21 max=sum; 22 } 23 } 24 } 25 printf("%d",max); 26 }</pre>		<pre>/tmp/kbfU09m0Wl.o 6 7 8 9 6 5 4 42 === Code Execution Successful ===</pre>

49. Sum of non repeated numbers : (unique element logic)

main.c

Run

Share

```
1 #include<stdio.h>
2 int main()
3 {
4     int a[100],i,j,n,sum=0;
5     printf("Enter n :");
6     scanf("%d",&n);
7     printf("Enter the elements :");
8     for(i=0;i<n;i++)
9     {
10         scanf("%d",&a[i]);
11     }
12     for(i=0;i<n;i++)
13     {
14         int c=0;
15         for(j=0;j<n;j++)
16         {
17             if(a[i]==a[j] && i!=j)
18             {
19                 c++;
20             }
21         }
22     }
23     if(c==0)
24     {
25         sum=sum+a[i];
26     }
27 }
28
29
30 printf("%d",sum);
31 }
```

/tmp/yXDXdaFRGE.o

enter n :6

Enter the elements :7 8 9 5 7 8

14

=== Code Execution Successful ===

50.LEFT CHAR ROTATION

51. Printing the index of the pairs that gives the target element :

```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr[n];
7     for(int i=0;i<n;i++)
8     {
9         scanf("%d",&arr[i]);
10    }
11    int target;
12    scanf("%d",&target);
13    for(int i=0;i<n;i++)
14    {
15        for(int j=i+1;j<n;j++)
16        {
17            if(arr[i]+arr[j]==target)
18            {
19                printf("%d %d ",i,j);
20            }
21        }
22    }
23 }
```

Output

```
/tmp/GZqrfh0F34.o
6
7 8 5 2 3 1
13
1 2

=== Code Execution Successful ===
```

52. Converting angle to radian :

```
main.c
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     double angle_degree, angle_radian;
6     printf("Enter the angle in degrees: ");
7     scanf("%lf", &angle_degree);
8     angle_radian = angle_degree * (M_PI / 180.0);
9     printf("Angle in radians: %.2f\n", angle_radian);
10    return 0;
11 }
12
```

Output

```
/tmp/AkzjmgRZQ1.o
Enter the angle in degrees: 145
Angle in radians: 2.53

=== Code Execution Successful ===
```

53.SUM OF TWO MATRICES

The screenshot displays the C Online Compiler interface. The main editor shows a C program named `main.c` that reads two matrices from standard input and prints their sum.

```
#include<stdio.h>
int main()
{
    int n;
    scanf("%d",&n);
    int arr1[n][n],arr2[n][n],arr3[n][n];
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
        {
            scanf("%d",&arr1[i][j]);
        }
    }
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
        {
            scanf("%d",&arr2[i][j]);
        }
    }
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
        {
            arr3[i][j]=arr1[i][j]+arr2[i][j];
        }
    }
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
        {
            printf("%d ",arr3[i][j]);
        }
        printf("\n");
    }
}
```

The right-hand side of the compiler shows the output of the program:

```
/tmp/sl2qEKX01Z.o
3
3 1 5 2 6 4 7 8 2
5 6 3 2 1 4 2 6 4
8 7 8
4 7 8
9 14 6

=== Code Execution Success
```

main.c	Output
<pre>1 #include<stdio.h> 2 int main() 3 { 4 int n; 5 scanf("%d",&n); 6 int arr1[n][n]; 7 for(int i=0;i<n;i++) 8 { 9 for(int j=0;j<n;j++) 10 { 11 scanf("%d",&arr1[i][j]); 12 } 13 } 14 15 int num; 16 scanf("%d",&num); 17 for(int i=0;i<n;i++) 18 { 19 for(int j=0;j<n;j++) 20 { 21 arr1[i][j]=num*arr1[i][j]; 22 } 23 } 24 for(int i=0;i<n;i++) 25 { 26 for(int j=0;j<n;j++) 27 { 28 printf("%d ",arr1[i][j]); 29 } 30 printf("\n"); 31 } 32 }</pre>	<pre>/tmp/9Rof6uuySa.o 3 1 5 6 3 2 4 8 9 4 2 2 10 12 6 4 8 16 18 8 === Code Execution Successful ===</pre>

55. Matrix multiplication

```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n,sum;
5     scanf("%d",&n);
6     int arr1[n][n],arr2[n][n],arr3[n][n];
7     for(int i=0;i<n;i++)
8     {
9         for(int j=0;j<n;j++)
10        {
11            scanf("%d",&arr1[i][j]);
12        }
13    }
14    for(int i=0;i<n;i++)
15    {
16        for(int j=0;j<n;j++)
17        {
18            scanf("%d",&arr2[i][j]);
19        }
20    }
21    for(int i=0;i<n;i++)
22    {
23        for(int j=0;j<n;j++)
24        {
25            sum=0;
26            for(int k=0;k<n;k++)
27            {
28                sum=sum+arr1[i][k]+arr2[k][j];
29            }
30            arr3[i][j]=sum;
31        }
32    }
33
34    for(int i=0;i<n;i++)
35    {
36        for(int j=0;j<n;j++)
37        {
38            printf("%d ",arr3[i][j]);
39        }
40        printf("\n");
41    }
42 }
43 }
```

Output

```
/tmp/3J27Q1kSsT.o
3
1 2 4 5 3 6 2 4 6
1 5 6 3 4 8 7 2 3
18 18 24
25 25 31
23 23 29

=== Code Execution Successful ===
```

56. Equal or unequal matrix

>>>

```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n,sum;
5     scanf("%d",&n);
6     int arr1[n][n],arr2[n][n];
7     for(int i=0;i<n;i++)
8     {
9         for(int j=0;j<n;j++)
10        {
11            scanf("%d",&arr1[i][j]);
12        }
13    }
14    for(int i=0;i<n;i++)
15    {
16        for(int j=0;j<n;j++)
17        {
18            scanf("%d",&arr2[i][j]);
19        }
20    }
21    int count =1;
22    for(int i=0;i<n;i++)
23    {
24        for(int j=0;j<n;j++)
25        {
26            if(arr1[i][j]!=arr2[i][j])
27            {
28                count =0;
29                break;
30            }
31        }
32    }
33    if(count==0)
34    {
35        printf("equal matrix");
36    }
37    else
38    {
39        printf("not equal matrix");
40    }
41 }
42
```

Output

```
/tmp/Stgu1tVu5Y.o
3
1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9
equal matrix

=== Code Execution Successful ===
```

57.SUM OF MAIN DIAGONAL

```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr1[n][n];
7     for(int i=0;i<n;i++)
8     {
9         for(int j=0;j<n;j++)
10        {
11            scanf("%d",&arr1[i][j]);
12        }
13    }
14    int sum=0;
15    for(int i=0;i<n;i++)
16    {
17        sum=sum+arr1[i][i];
18    }
19    printf("%d",sum);
20 }
21
22
```

Output

```
/tmp/SYKvabZLxY.o
3
1 2 3 4 5 6 7 8 9
15

=== Code Execution Successful ===
```

58.UPPER TRIANGULAR MATRIX

```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr1[n][n];
7     for(int i=0;i<n;i++)
8     {
9         for(int j=0;j<n;j++)
10        {
11            scanf("%d",&arr1[i][j]);
12        }
13    }
14    int upper=0;
15    for(int i=0;i<n;i++)
16    {
17        for(int j=0;j<n;j++)
18        {
19            if(j<i && arr1[i][j]!=0)
20            {
21                upper=1;
22            }
23        }
24    }
25    if(upper==0)
26    {
27        printf("upper triangular");
28    }
29    else
30    printf("not upper triangular");
31 }
```

```
/tmp/5mvcnf4v1v.o
3
1 2 3
0 2 3
0 0 3
upper triangular

=== Code Execution Successful ===
```

59.LOWER TRIANGULAR MATRIX

```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr1[n][n];
7     for(int i=0;i<n;i++)
8     {
9         for(int j=0;j<n;j++)
10        {
11            scanf("%d",&arr1[i][j]);
12        }
13    }
14    int lower=0;
15    for(int i=0;i<n;i++)
16    {
17        for(int j=0;j<n;j++)
18        {
19            if(j>i && arr1[i][j]!=0)
20            {
21                lower=1;
22            }
23        }
24    }
25    if(lower==0)
26    {
27        printf("lower triangular");
28    }
29    else
30    printf("not lower triangular");
31 }
```

```
/tmp/l2inQ1q5jX.o
3
1 0 0
1 2 0
1 2 3
lower triangular

=== Code Execution Successful ===
```

60.TRANSPOSE OF MATRIX

main.c

Share

Run

```
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int arr1[n][n],arr2[n][n];
7     for(int i=0;i<n;i++)
8     {
9         for(int j=0;j<n;j++)
10         {
11             scanf("%d",&arr1[i][j]);
12         }
13     }
14     for(int i=0;i<n;i++)
15     {
16         for(int j=0;j<n;j++)
17         {
18             arr2[j][i]=arr1[i][j];
19         }
20     }
21
22     for(int i=0;i<n;i++)
23     {
24         for(int j=0;j<n;j++)
25         {
26             printf("%d ",arr2[i][j]);
27         }
28         printf("\n");
29     }
30 }
```

/tmp/kz62trxgvt.o

3

1 2 3

4 5 6

7 8 9

1 4 7

2 5 8

3 6 9

=== Code Execution Successful ===