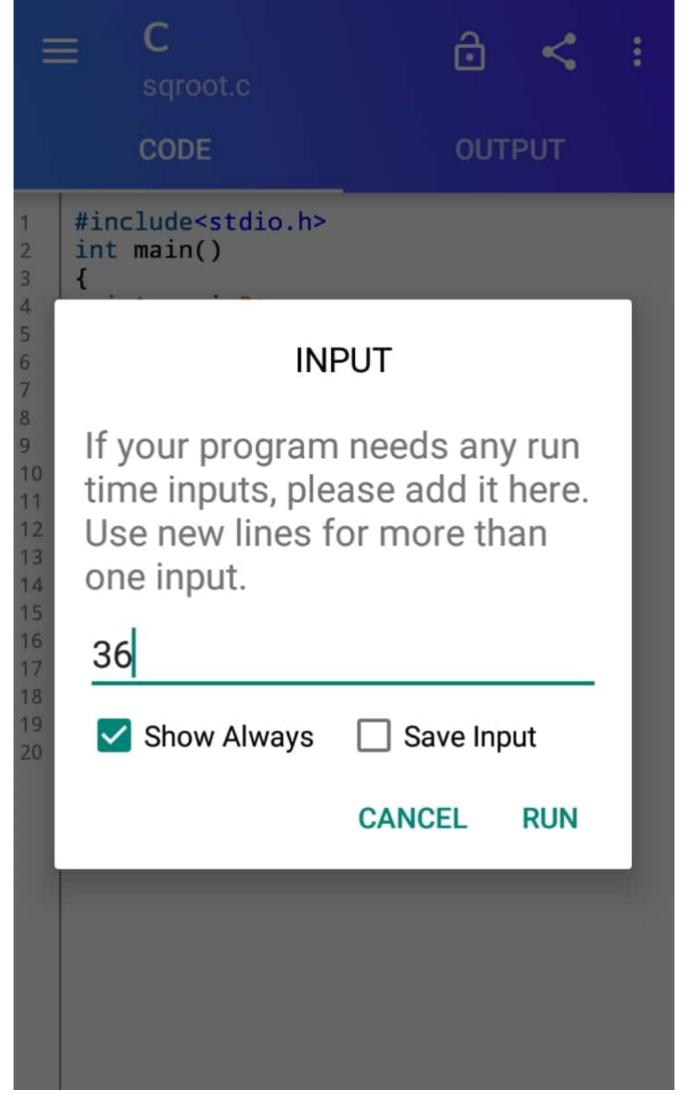
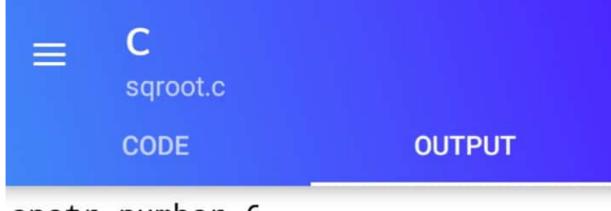


```
#include<stdio.h>
    int main()
2
3
    {
      int n , i=0 ;
4
      int root=1;
5
      printf("enetr number ");
6
7
      scanf("%d",&n);
8
      while(1)
9
10
      {
         i=i+1;
11
         root = (n/root+root)/2;
12
        if(i == n+1)
13
         {
14
           break;
15
         }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





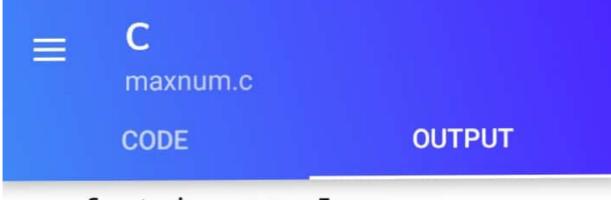
enetr number 6



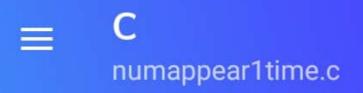


OUTPUT

```
//3. MAXIMUM HIGHT OF STAIRS
1
    #include<stdio.h>
2
    int main()
3
    {
4
         int blocks=20;
5
         int stairs=0;
6
         while(blocks!=0)
7
8
         {
             if(stairs+1<= blocks)</pre>
9
             {
10
                  stairs++;
11
                  blocks=blocks-stairs;
12
             }
13
             else
14
15
             {
                  break;
16
             }
17
18
         printf("no of stairs are
19
    %d", stairs);
    }
20
21
```



no of stairs are 5





2

3

4 5

6 7

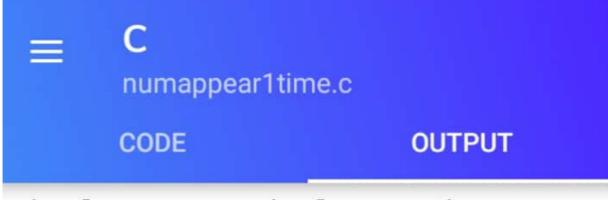
8

9

17

OUTPUT

```
//1. number which appears one time
    #include<stdio.h>
   int main()
            int a[]=\{5,3,3,4,5,2,4\};
            int res=a[0];
            for(int i=1; i<7; i++)
10
                 res=res^a[i];
11
12
            printf("single repeated element
13
    is %d", res);
14
15
16
```



single repeated element is 2



TAB

OUTPUT

```
#include <stdio.h>
1
2
    int PathCounting(int m, int n)
3
4
        int ctr[m][n];
5
        for (int i = 0; i < m; i++)
6
7
        ctr[i][0] = 1;
8
9
        for (int j = 0; j < n; j++)
10
11
        ctr[0][j] = 1;
12
         }
13
        for (int i = 1; i < m; i++)
14
15
             for (int j = 1; j < n; j++)
16
17
           ctr[i][j] = ctr[i-1][j] + ctr[i]
18
    [j-<mark>1</mark>];
           }
19
20
        return ctr[m-1][n-1];
21
    }
22
23
    int main()
24
    {
25
26
        int p,q;
        printf("enter no. of rows of matrix
27
    ");
         scanf("%d",&p);
28
        printf("\nenter no. of column of
29
    matrix ");
```

&

RUN



TAB

OUTPUT

```
ctr[i][0] = 1;
8
9
        for (int j = 0; j < n; j++)
10
11
        ctr[0][j] = 1;
12
        }
13
        for (int i = 1; i < m; i++)
14
15
             for (int j = 1; j < n; j++)
16
17
           ctr[i][j] = ctr[i-1][j] + ctr[i]
18
    [j-<mark>1</mark>];
19
20
        return ctr[m-1][n-1];
21
    }
22
23
    int main()
24
    {
25
        int p,q;
26
        printf("enter no. of rows of matrix
27
    ");
        scanf("%d",&p);
28
        printf("\nenter no. of column of
29
    matrix ");
        scanf("%d",&q);
30
      printf("\nThe size of matrix is : %d,
31
    %d\n",p,q);
      printf("\nThe all possible paths from
32
    top left to bottom right is: %d
    \n",PathCounting(p,q));
33
```

RUN

&

```
pathcounting.c
        CODE
                               OUTPUT
   #include <stdio.h>
2
3
4
5
                   INPUT
6
7
8
    If your program needs any run
9
    time inputs, please add it here.
10
11
    Use new lines for more than
12
    one input.
13
14
15
     3
16
     4
17
18
19
     Show Always
                           Save Input
20
21
22
                        CANCEL
                                    RUN
23
24
    {
25
        int p,q;
26
        printf("enter no. of rows of matrix
27
    ");
        scanf("%d",&p);
28
        printf("\nenter no. of column of
29
   matrix ");
                                 &
                                       RUN
TAB
```



OUTPUT

enter no. of rows of matrix
enter no. of column of matrix
The size of matrix is : 3, 4

The all possible paths from top left to bottom right is: 10