



# Coding Challenge #29 (Question)

Consider the series: 0,0,2,1,4,2,6,3,8,4,10,5,12,6,14,7,16,8

Write a program to find the  $n^{\text{th}}$  term in this series.

*This series is a mixture of 2 series. All the odd terms in this series form even numbers in ascending order starting 0 and all even terms are derived from the previous term (x) using the formula  $(x/2)$ .*

Input Format:

The value n is a positive integer that should be read from STDIN.

Output Format:

The  $n^{\text{th}}$  term that is calculated by the program should be written to STDOUT. Other than the value of the  $n^{\text{th}}$  term no other character /string or message should be written to STDOUT.

Sample Input 0:

5

Sample Output 0:

4

Sample Input 1:

10

Sample Output 1:

4



# Coding Challenge #29 (C Solution)

```
#include<stdio.h>

int main()
{
    int n,a,d,t_s1,t_s2,n_term;
    scanf("%d",&n);
    if(n%2==1)
    {
        a=0,d=2;
        t_s1=(n+1)/2;
        n_term=a+(t_s1-1)*d;
        printf("%d",n_term);
    }
    else
    {
        a=0,d=1;
        t_s2=n/2;
        n_term=a+(t_s2-1)*d;
        printf("%d",n_term);
    }
}
```



# Coding Challenge #29 (JAVA Solution)

```
import java.util.Scanner;
class Main {
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int a = 0, b = 0;
        if(n % 2 == 0)
        {
            for(int i = 1 ; i <= (n-2) ; i = i+2)
            {
                a = a + 2;
                b = a / 2;
            }
            System.out.print(b);
        }
        else
        {
            for(int i = 1 ; i < (n-2) ;
            i = i+2)
            {
                a = a + 2;
                b = a / 2;
            }
            a = a + 2;
            System.out.print(a);
        }
    }
}
```



# Coding Challenge #30 (Question)

Program to find the hypotenuse of a triangle.

*Get the opposite and adjacent sides from the user and calculate and display the hypotenuse of the given triangle. The output is a floating-point value with precision 2.*

Sample Input 0:

2.5

3.5

Sample Output 0:

4.30

Sample Input 1:

5.8

6.8

Sample Output 1:

8.94



# Coding Challenge #30 (C Solution)

```
#include <stdio.h>

#include<math.h>

int main()

{

    float hyp, opp, adj;

    scanf("%f%f", &opp, &adj);

    hyp=sqrt((opp*opp) + (adj*adj));

    printf("%0.2f", hyp);

    return 0;

}
```



# Coding Challenge #30 (JAVA Solution)

```
import java.util.*;

import java.lang.*;

class Main {

    public static void main (String[] args)

    {

        Scanner sc=new Scanner(System.in);

        float hyp, adj;

        float opp=sc.nextFloat();

        adj=sc.nextFloat();

        hyp=(float) Math.sqrt((opp*opp) + (adj*adj));

        System.out.printf("%.2f", hyp);

    }

}
```