S.No: 7	Exp. Name: ***Total Number of Squares***	Date:2024-10-24

Aim:

Tina's brother gave her a fun mathematical challenge involving a chessboard-like grid. The task is to determine the total number of squares that can be found in an $n \times n$ board, where each square is $1\ \text{cm} \times 1\ \text{cm}$. However, Tina needs to count not only the smallest squares but also the larger squares that can be formed on the board.

For example, on a 3×3 board, there are 9 squares of size 1×1, 4 squares of size 2×2, and 1 square of size 3×3, resulting in a total of 14 squares.

Your task is to help Tina calculate the total number of squares of all possible sizes on the board.

Constraints

- $2 \leq n \leq 20$

Input Format

- The input consists of a single integer n, representing the size of the board.

Output Format

- Print the total number of squares in the $n \times n$ board.

Formula Hint

The formula to calculate the total number of squares on an $n \times n$ board is given by:

$$\text{Total squares } = \frac{n \times (n+1) \times (2n+1)}{6}$$

Source Code:

squares.c

```
#include<stdio.h>
int main(){
   int a;
   int sum=0;
   scanf("%d",&a);
   for(int i=1;i<=a;i++)
      {
         sum=sum +(i*i);
      }
   printf("%d",sum);
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
2
5

Test Case - 2
User Output
4
30

Test Case - 3
User Output
6
91

Test Case - 4
User Output
8
204