Print squares of first n natural numbers without using *, / and –

Given a natural number ‘n’, print squares of first n natural numbers without using *, / and -.

```
Input:  n = 5
Output: 0 1 4 9 16

Input:  n = 6
Output: 0 1 4 9 16 25
```

We strongly recommend to minimize the browser and try this yourself first.

Method 1: The idea is to calculate next square using previous square value. Consider the following relation between square of x and (x-1). We know square of (x-1) is $(x-1)^2 – 2*x + 1$. We can write x^2 as

$$x^2 = (x-1)^2 + 2*x - 1$$
$$x^2 = (x-1)^2 + x + (x - 1)$$

When writing an iterative program, we can keep track of previous value of x and add the current and previous values of x to current value of square. This way we don’t even use the ‘-‘ operator.

```
// C++ program to print squares of first 'n' natural numl
// wothout using *, / and -
#include<iostream>
using namespace std;

void printSquares(int n)
{
    // Initialize 'square' and previous value of 'x'
    int square = 0, prev_x = 0;

    // Calculate and print squares
```

```
    for (int x = 0; x < n; x++)
    {
        // Update value of square using previous value
        square = (square + x + prev_x);

        // Print square and update prev for next iterati
        cout << square << " ";
        prev_x = x;
    }
}

// Driver program to test above function
int main()
{
   int n = 5;
   printSquares(n);
}
```

Output:

```
0 1 4 9 16
```

Method 2: Sum of first n odd numbers are squares of natural numbers from 1 to n. For example 1, 1+3, 1+3+5, 1+3+5+7, 1+3+5+7+9, ….

Following is C++ program based on above concept. Thanks to Aadithya Umashanker and raviteja for suggesting this method.

```
// C++ program to print squares of first 'n' natural numl
// wothout using *, / and -
#include<iostream>
using namespace std;

void printSquares(int n)
{
    // Initialize 'square' and first odd number
    int square = 0, odd = 1;

    // Calculate and print squares
    for (int x = 0; x < n; x++)
    {
```

```
        // Print square
        cout << square << " ";

        // Update 'square' and 'odd'
        square = square + odd;
        odd = odd + 2;
    }
}

// Driver program to test above function
int main()
{
   int n = 5;
   printSquares(n);
}
```

Output:

```
0 1 4 9 16
```