You are given the following pyramid:

1

2 3

4 5 6

7 8 9 10

..........

Your task is to find the sum of all prime numbers in the nth line.

**Example:**

* primesum(1) = 0
* primtsum(2) = 5
* **[input] integer n**
  + 1 ≤ n ≤ 1000
* **[output] integer**

<https://codefights.com/challenge/hGhzddqBDHQTEMFPM>

#include <iostream>

#include <vector>

#include <stdio.h>

#include <math.h>

int primesum(int n)

{

struct Helper{

bool esPrimo(int n)

{

if (n < 2) return false;

if (n == 2) return true;

if (n % 2 == 0) return false;

int sqr = (int) sqrt(n);

for (int i = 3; i <= sqr; i += 2)

{

if (n % i == 0) return false;

}

return true;

}

};

std::vector<std::vector<int> > matriz ; //= new List<List<int>>();

int num = 1;

for (int i = 0; i <= n; i++)

{

std::vector<int> fila; // = new List<int>();

for (int j = 0; j < i; j++)

{

fila.push\_back(num);

num++;

}

matriz.push\_back(fila);

}

/\*

for (int i = 0; i < matriz.size(); i++)

{

for (int j = 0; j < matriz[i].size(); j++)

{

Console.Write(matriz[i][j] + " ");

}

Console.WriteLine();

}\*/

// List<int> filaElegida = new List<int>();

int sum = 0;

Helper h;

for (int i = 0; i < matriz[n].size(); i++)

{

// filaElegida.Add(matriz[n][i]);

if (h.esPrimo(matriz[n][i]))

{

sum += matriz[n][i];

}

}

//Console.ReadLine();

return sum;

}

int main() {

int res = primesum(2);

printf("%d ", res);

return 0;

}