Fibonacci-Básico

|  |  |  |  |
| --- | --- | --- | --- |
| Puntos |  | Límite de memoria | 32MB |
| Límite de tiempo (caso) | 1s | Límite de tiempo (total) | 60s |

**Descripción**

Dado un entero M, imprime todos los números de la serie de fibonacci menores a M.

La serie de fibonacci se calcula:

1 2 3 5 8 13 21 34...

**Entrada**

Una línea con el entero M.

**Salida**

Los números de la serie fibonacci menores a M, separados por un espacio.

**Ejemplos**

| **Entrada** | **Salida** | **Descripción** |
| --- | --- | --- |
| 26 | 1 2 3 5 8 13 21 |  |

*Fuente: lacj20*

Problema subido por: [lacj20](https://omegaup.com/profile/lacj20/)

| **ID** | **Status** | **Porcentaje** | **Penalty** | **Lenguaje** | **Memoria** | **Tiempo** | **Detalles** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| [Nuevo envío](https://omegaup.com/arena/problem/Fibonacci-Basico#problems/new-run) | | | | | | | | |
| 1d363fc9 | Respuesta incorrecta | 0.00% | 0 | cpp | 3.21 | 0.02 |  |  |
| 53bb73d4 | Respuesta incorrecta | 0.00% | 0 | cpp | 3.01 | 0.01 |  |  |
| 3924ae2a | Respuesta correcta | 100.00% | 0 | cpp | 3.21 | 0.01 |  |  |
| **Envíos** | | | | | | | |

<https://omegaup.com/arena/problem/Fibonacci-Basico#problems>

#include <iostream>

#include <stdio.h>

#include <map>

#include <limits>

#include <vector>

#include <conio.h>

using namespace std;

int main() {

int M;

scanf("%d", &M);

if(M == 1) {

printf("1");

} else {

std::vector<int> fib;

fib.push\_back(1);

fib.push\_back(2);

for (int i = 2; ; i++){

if (fib[i - 1] + fib[i - 2] >= M){

break;

}

fib.push\_back(fib[i - 1] + fib[i - 2]);

}

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

printf("%d ", fib[i]);

}

}

getch();

return 0;

}