Vetor V[] com **N**=15 elementos e **M**=3

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  | i | i+1 |  |  |  |  |  |  |  |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | índice do vetor |
| 1 | 3 | 4 | 5 | 6 | 7 | 2 | 9 | 10 | 11 | 8 | 12 | 13 | 14 | 15 |  |
| 1 | 0 | 1 | 2 | 0 | 1 | 2 | 0 | 1 | 2 | 2 | 0 | 1 | 2 | 0 | V[i] % M |

V[i] % M

1 % 3

Considere um vetor com **N** números inteiros positivos e um inteiro positivo **M**, o seu programa deve ordenar estes **N** números em ordem crescente do valor do seu **módulo** utilizando o inteiro **M**.

Entrada

M=3

int V[]={7,15};

MuitaSort(V,M);

// imprime vetor =>

Se houver um **empate** entre um número **ímpar** e um número **par** (para os quais o seus **módulos** calculados com inteiro **M** dá o mesmo valor) então o número **ímpar** irá **preceder** o número **par**.

Se houver um **empate** entre dois números **ímpares** (para os quais os seus **módulos** calculados com inteiro **M** dá o mesmo valor), então o maior número ímpar irá preceder o menor número ímpar.

Se houve um **empate** entre dois números **pares** (para os quais os seus **módulos** calculados com inteiro **M** dá o mesmo valor), então o menor número par irá preceder o maior número par.