it#include <bits/stdc++.h>

int main(){

vector<int> vec(10);

// functions

//vector vacio?

vec.empty();

//ingresar valores al vector de forma dinamica

vec.push\_back(value);

//valor del tamaño del vector

vec.size();

/\*vectores

\*matrices = vectores de vectores de tamaño NxM

\*/

vector<vector<int> > matrix(N, vector<int>(M,0));

// other way

int N=4;int M=4;

for(int i=0; i< N; i++){

matrix[i] = vector<int>(M);

}

// recorrer matrix

int s=0;

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

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

s+= matrix[i][j];

}

}

// inciar la diagonal en uno

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

matrix[i][i] = 1;

}

// Guardar pares en un vector

pair<int, int> p(1,1);

cout<<p.first<<","<<p.second<<"/n";

p = make\_pair(3,4);// mismo tipo de dato

pair <int, string> p1 = make\_pair(3,"texto");//con varios tipos de datos

cout<<p.first<<","<<p.second<<"/n";

//asignar valores

p.first=3;

p.second="El texto";

cout<<p.first<<","<<p.second<<"/n";

// stranger things

vector<pair<int,int> > vec(10);

//vector<pair<int,string> > vec(10);

cout << vec[0].first<<","<<vec[0]<<"/n";

/\*

ITERATORS REcorerlos de forma rara .\_.

\*/

vector<int> vec(10);

for(vector<int>:: iterator it= vec.begin();

it != vect.end(); ++it){

cout<< \*it << " ";

}

/\*

si pero no jajja

for(auto iterator it= vec.begin();

it != vect.end(); ++it)

\*/

// templates en c++?

}