Ejercicio 1:

#include <iostream>

using *namespace* std;

*template* <*typename* T>

*class* Cubo{

*public:*

    T lado;

*int* cantidad\_veces\_a\_sumar;

    Cubo(T *lado*) : lado(*lado*), cantidad\_veces\_a\_sumar(6) {}

    // Area del cubo = l\*\*2 \* 6 = l\*l sumado 6 veces = l\*l + l\*l + l\*l + l\*l + l\*l + l\*l

    T hallar\_area\_recursivo\_aux(*int* *cantidad\_veces\_a\_sumar*)

    {

        if (*cantidad\_veces\_a\_sumar* == 0)

            return 0;

        return lado \* lado + hallar\_area\_recursivo\_aux(*cantidad\_veces\_a\_sumar* - 1);

    }

    T hallar\_area\_recursivo()

    {

        return hallar\_area\_recursivo\_aux(this->cantidad\_veces\_a\_sumar);

    }

    // Volumen del cubo = l\*\*3 = l\*l\*l

*double* hallar\_potencia(T *base*, *int* *exp*)

    {

        if (*exp* == 0)

            return 1;

        else if (*exp* < 0)

            return 1 / hallar\_potencia(*base*, -*exp*);

        else

            return *base* \* hallar\_potencia(*base*, *exp* - 1);

    }

    T hallar\_volumen\_recursivo()

    {

        return hallar\_potencia(lado, 3);

    }

};

*int* main()

{

*double* lado;

    do

    {

        cout << "Ingrese el valor del lado en rango de 1 a 30: ";

        cin >> lado;

    } while (lado < 1 || lado > 30);

    Cubo<*int*> \*cuboINT = new Cubo<*int*>(lado);

    Cubo<*double*> \*cuboDOUBLE = new Cubo<*double*>(lado);

    cout << "El area del cubo entero con lado " << *int*(lado) << " es: " << cuboINT->hallar\_area\_recursivo() << endl;

    cout << "El area del cubo double con lado " << lado << " es: " << cuboDOUBLE->hallar\_area\_recursivo() << endl;

    cout << "El volumen del cubo entero con lado " << *int*(lado) << " es: " << cuboINT->hallar\_volumen\_recursivo() << endl;

    cout << "El volumen del cubo double con lado " << lado << " es: " << cuboDOUBLE->hallar\_volumen\_recursivo() << endl;

    return 0;

}

#include <iostream>

Ejercicio 2:

#include <vector>

#include <math.h>

#include <map>

using *namespace* std;

*bool* esPrimo(*int* *x*)

{

    if (x == 1)

        return false;

*int* limite = sqrt(x);

    for (*int* i = 2; i <= limite; i++)

    {

        if (x % i == 0)

            return false;

    }

    return true;

}

*void* recorrer\_vector(map<*int*, *int*> &*m*, vector<*int*> &*v*)

{

*int* n = v.size();

    for (*int* i = 0; i < n; i++)

    {

*int* x = v[i];

        if (esPrimo(x))

            m[x]++;

    }

}

*void* hallar\_numeros\_primos\_misma\_frecuencia(vector<*int*> &*v1*, vector<*int*> &*v2*)

{

    map<*int*, *int*> m1;

    map<*int*, *int*> m2;

    recorrer\_vector(m1, v1);

    recorrer\_vector(m2, v2);

    for (*auto* i : m1)

        if (i.second == m2[i.first])

            cout << i.first << " ";

}

*int* main()

{

*int* n, m;

    vector<*int*> v1, v2;

    do

    {

        cout << "Ingrese tamano V1: ";

        cin >> n;

    } while (n < 1 || n > 26);

    do

    {

        cout << "Ingrese tamano V2: ";

        cin >> m;

    } while (m < 1 || m > 26);

    cout << "V1: " << endl;

    for (*int* i = 0; i < n; i++)

    {

*int* x;

        do

        {

            cin >> x;

        } while (x <= 0);

        v1.push\_back(x);

    }

    cout << "V2: " << endl;

    for (*int* i = 0; i < m; i++)

    {

*int* x;

        do

        {

            cin >> x;

        } while (x <= 0);

        v2.push\_back(x);

    }

    hallar\_numeros\_primos\_misma\_frecuencia(v1, v2);

    return 0;

}

#include <iostream>

Ejercicio 3:

#include <vector>

#include <utility>

#include <algorithm>

using *namespace* std;

*bool* comparacion(pair<string, *int*> *p1*, pair<string, *int*> *p2*)

{

    return (*p1*.second > *p2*.second);

}

*int* main()

{

*int* c;

    vector<pair<string, *int*>> numeros\_candidatos;

    do

    {

        cout << "Ingresar numeros de candidatos: ";

        cin >> c;

    } while (c < 0 || c > 30);

    for (*int* i = 0; i < c; i++)

    {

        string nombre\_producto;

*int* numero\_compras;

        cin >> nombre\_producto >> numero\_compras;

        numeros\_candidatos.push\_back(make\_pair(nombre\_producto, numero\_compras));

    }

    for (*auto* i : numeros\_candidatos)

        cout << i.first << " - " << i.second << endl;

    cout << endl;

    sort(numeros\_candidatos.begin(), numeros\_candidatos.end(), comparacion);

    if (c < 2)

        cout << " ";

    else

    {

        for (*int* i = 0; i < 2; i++)

            cout << numeros\_candidatos[i].first << " - " << numeros\_candidatos[i].second << endl;

    }

    cout << endl;

    cout << numeros\_candidatos[0].first << " - " << numeros\_candidatos[0].second << " - "

         << "Producto mas vendido" << endl;

    return 0;

}