PISTAS

// Pista 1: Definir dos contenedores diferentes de tipos diferentes

std::vector<int> numeros;

std::list<std::string> palabras;

// Pista 2: Inicializar los contenedores con algunos valores

numeros.push\_back(5);

numeros.push\_back(2);

numeros.push\_back(7);

numeros.push\_back(1);

numeros.push\_back(8);

palabras.push\_back("perro");

palabras.push\_back("gato");

palabras.push\_back("elefante");

palabras.push\_back("leon");

palabras.push\_back("tigre");

// Pista 3: Crear una función lambda que acepte un argumento y devuelva un valor booleano

auto es\_par = [](int n) { return n % 2 == 0; };

auto es\_letra\_e = [](const std::string& s) { return s == "e"; };

// Pista 4: Utilizar el algoritmo std::replace\_if para reemplazar los elementos que cumplen con la condición

std::replace\_if(numeros.begin(), numeros.end(), es\_par, 0);

palabras.remove\_if(es\_letra\_e);

// Pista 5: Ejemplo de iteración

for (auto iter = numeros.begin(); iter != numeros.end(); ++iter)

{

std::cout << \*iter << " ";

}

std::cout << std::endl;

for (auto iter = palabras.begin(); iter != palabras.end(); ++iter)

{

std::cout << \*iter << " ";

}

std::cout << std::endl;