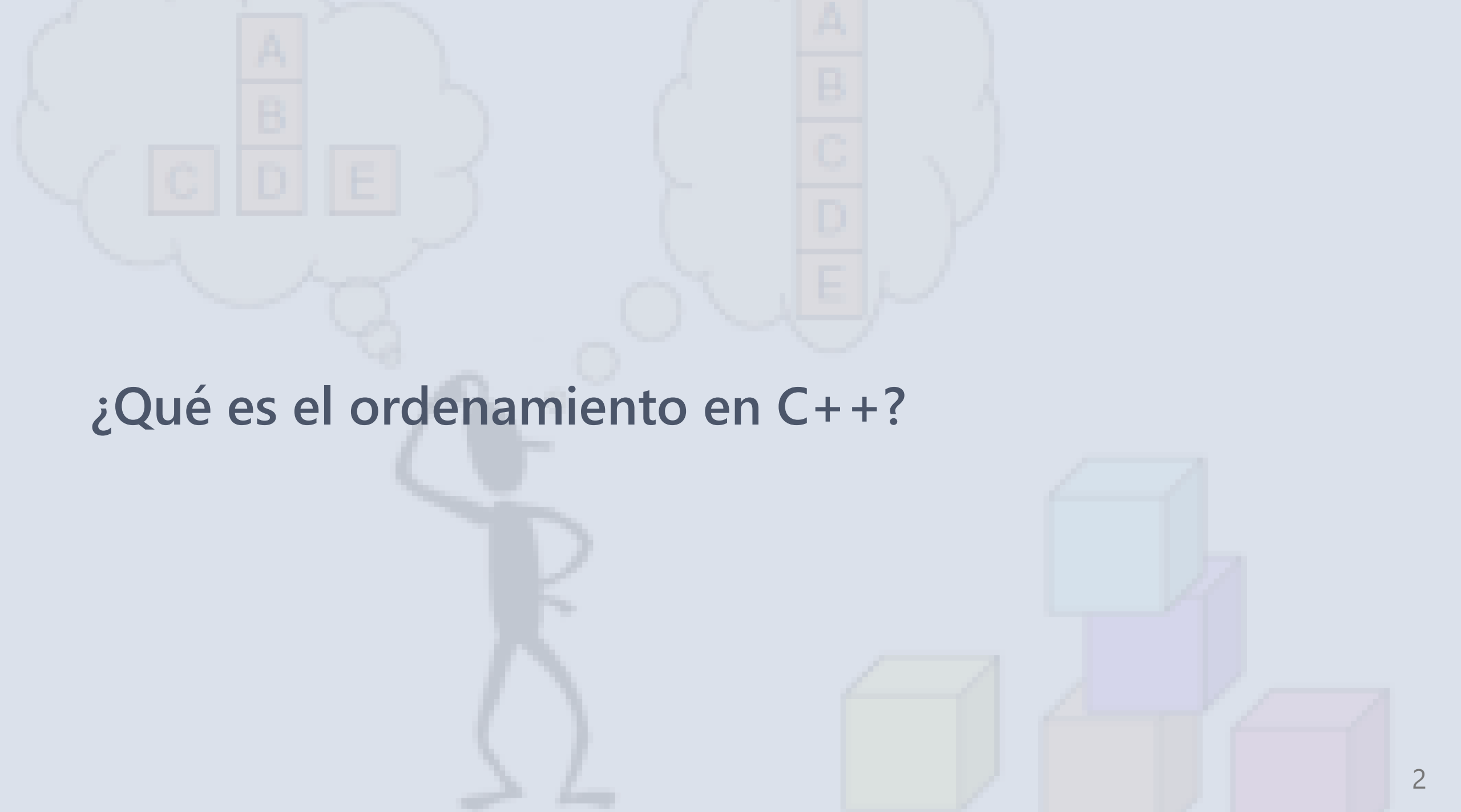
Ordenamiento y Clasificación +

Libreria STD y sus estructuras de datos basicas

Por Ariel Parra.



¿Qué es el ordenamiento en C++?

El ordenamiento o en ingles sorting, son algoritmos que se utilizan para organizar los elementos de una colección en un orden especifico.

los algortimos mas comunes son los de la burbuja, seleccion, insercion, merge sort y quick sort.

Burbuja Seleccion e Insercion

Burbuja O(N^2): compara e intercambia elementos adyacentes si estan en el orden incorrecto.

Seleccion O(N^2): busca el elemento mas pequeno y lo coloca en la posicion adecuada.

Insercion O(N^2): construye una sublista ordenada y coloca los elementos restantes en su lugar.

Merge sort y Quick sort

Merge sort O(Nlog(N)): este divide la coleccion en mitades, ordena cada mitad y luego fusiona las mitades ordenadas.

Quick sort O(Nlog(N)): selecciona un elemento llamado pivote y particiona la coleccion en dos partes ordenandolas por separado.

¿Qué es la clasificación en C++?

La clasificacion o busqueda son algoritmos que dividen un conjunto en categorias o clases en funcion de un criterio.

Los dos principales metodos de busqueda son los de busqueda lineal y la busqueda binaria.

Busqueda lineal

La busqueda lineal recorre los elementos de la coleccion secuencialmente hasta encontrar el elemento buscado.

Si el elemento se encuentra se devuelve su posicion; de lo contrario se indica que no se encontro.

La complejidad de la busqueda lineal es O(N) donde N es el tamano de la coleccion.

Busqueda binaria

Se aplica a colecciones ordenadas y divide repetidamente la busqueda a la mitad.

Se compara el elemento buscado con el elemento central de la coleccion y se descarta la mitad no deseada.

La complejidad temporal de la busqueda binaria es de O(log(N)), donde n es el tamaño de la coleccion.

¿Qué son las estructuras de datos?

Son una forma organizacion de datos almacenados y elejidos dependiendo del uso y eficacia.

Los cuatro estructuras de datos que usaramos mas seguido en en la libreria std son los strings, vectores, sets y pairs.

¿Qué es la libreria STD de C++ y?

Standard Library es una colección de clases y funciones, que crean la base del lenguaje C++, estas se basan en la Standard Template Library (STL).
Esta libreria simplifica el uso y manipulacion de varias estructuras de datos podientolas usar con operadores booleanos y de asignacion como si fuesen cualquier tipo de dato de C++.

librerias más usadas:

<iostream>: Funciones de entrada y salida estándar como el std::cin y std::cout .

<fstream>: Permite el manejo de archivos.

<algorithm>: Contiene los algoritmos mas conocigos y eficientes, como std::sort() que usa una variacion de quick sort.

<utility>: utilidades y funciones generales especialmente el tipo std::pair.

<vector>: Implementa arreglos dinámicos y funciones para manipularlos.

<string>: Ofrece funcionalidades para manipular cadenas de caracteres, dandole funciones similares a las de los vectores.

<set>: Implenta funciones para manipular sets.

STD::string

son cadenas de caracteres de tamano dinamico.

La complejidad de tiempo y espacio de sus funciones suele ser O(N) y O(1).

función	Descripción
string _nombre_;	inicializa un string
string _nombre_ = "_valor_";	inicializa un string con un valor
nombre.c_str();	convierte el string std a string de c
nombre.lenght();	devuelve el tamano del string
nombre.empty();	comprueba si esta vacia
nombre.clear();	limpia el string
nombre.insert(_posicion_,_string_);	inserta un string en la posicion dada
nombre.append(_string_);	agrega contenido al string
nombre.compare(_string_);	compara dos strings

STD::vector

Gestiona automaticamente la memoria y almacena cualquier tipo de dato asignado.
La complejidad de tiempo y espacio de sus funcinoes suele ser O(N) y O(1).

función	**Descripción**
vector <_tipo_> _nombre_;	inicializa un vector
vector <_tipo_> _nombre_(_tamaño_,*valorInicial*);	inicializa un vector con un tamano definido y valores iniciales
nombre.size();	devuelbe el tamano del vector
nombre.empty();	verifica si esta vacio
nombre.clear();	limpia el vector
nombre.pushback(_valor_);	agrega un valor al final
nombre.count(_valor_);	cuenta cuantos elementos hay con ese valor

función	Descripción
nombre.pop_back();	elimina el ultimo elemento del vector
nombre.insert(_posicion_,_valor_);	inserta un valor en esa posicion
nombre.erase(_posicion_);	elimina un valor en la posicion dada
nombre.swap(_otroVector_);	intercambia el contenido con otro
nombre.front();	devuelve el primer elemento del vector
nombre.back();	devuelve el ultimo elemento del vector

STD::set

Estos son vectores con elementos no repetidos y con un orden ascendente.

La complejidad de tiempo y espacio de sus funcinoes suele ser O(log(N)).

función	Descripción
set <_tipo_> _nombre_;	inicializa un set
nombre.insert(_value_);	inserta un valor al set
nombre.erase(_valor_);	elimina el valor dado si esta en el set
nombre.empty()	verifica si esta vacio
nombre.clear();	limpia el set
nombre.count(_valor_);	cuenta cuantos elementos hay (0 o 1)

STD::pair

Un pair es una tipo de dato para variables con dos tipos de datos.

La complejidad de tiempo y espacio de sus funcinoes suele ser O(1).

función	Descripción
pair <_tipo1_,_tipo2_> _nombre_;	inicializa un pair
pair <_tipo1_,_tipo2_> _nombre_(_valor1_,_valor2_);	inicializa un pair con valores iniciales
nombre.first;	devuelve el primer elemento
nombre.second;	devuelve el segundo elemento
swap(_pair1_,_pair2_);	intercambia sus contenidos (en si no es especifica de pair)

Referencias

https://en.wikipedia.org/wiki/C%2B%2B_Standard_Library

https://en.wikipedia.org/wiki/Standard_Template_Library

https://cplusplus.com/reference/string/string/

https://cplusplus.com/reference/vector/vector/

https://cplusplus.com/reference/set/set/

https://cplusplus.com/reference/utility/pair/

https://en.wikipedia.org/wiki/Data_structure

https://www.geeksforgeeks.org/sorting-algorithms/