

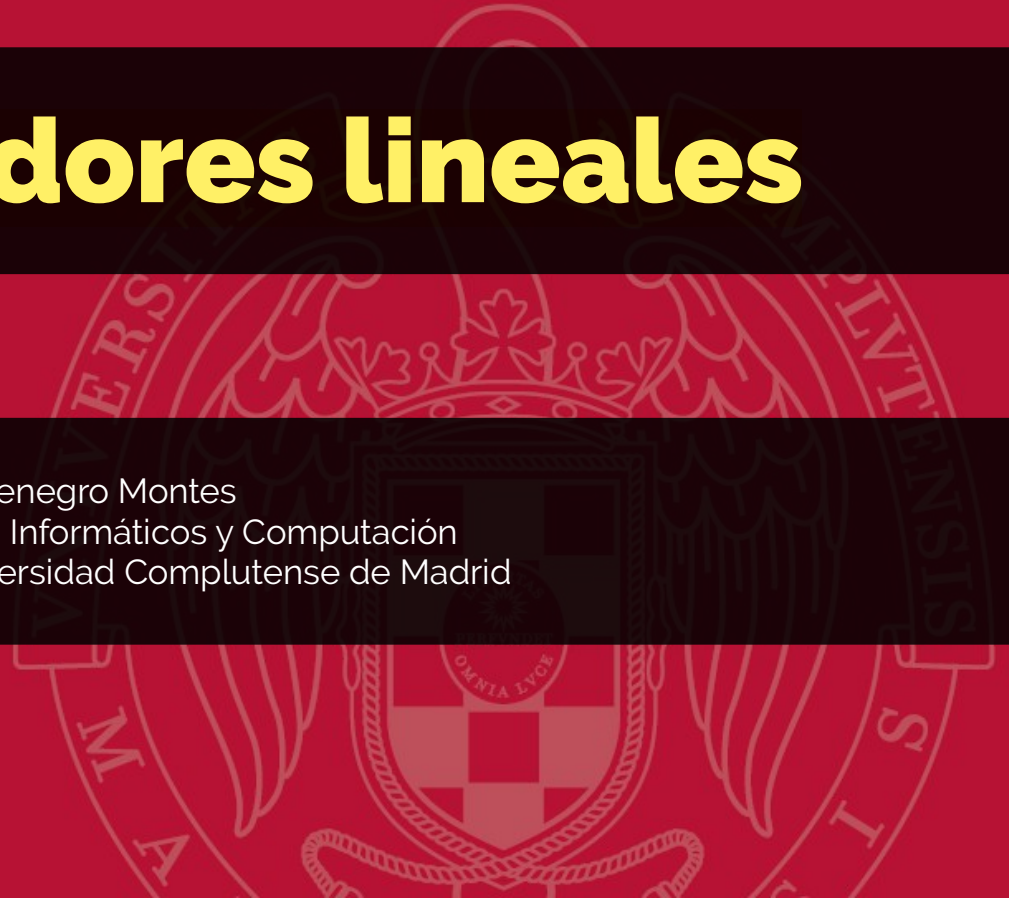
ESTRUCTURAS DE DATOS

NOTAS SOBRE C++

STL: Contenedores lineales

Manuel Montenegro Montes

Departamento de Sistemas Informáticos y Computación
Facultad de Informática – Universidad Complutense de Madrid



¿Qué es la STL?

STL = *Standard Template Library*

- Es una librería estándar de C++ que proporciona una serie de utilidades al programador/a.
 - Tipos abstractos de datos para almacenar colecciones de elementos: listas, pilas, colas, conjuntos, diccionarios, etc.
 - Iteradores.
 - Algoritmos sobre estos tipos abstractos de datos.

Tipos de datos lineales en la STL

Clase	Fich. cabecera	Estructura
<code>std::vector</code>	<code><vector></code>	TAD Lista (arrays)
<code>std::list</code>	<code><list></code>	TAD Lista (listas doblemente enlazadas)
<code>std::forward_list</code>	<code><forward_list></code>	TAD Lista (listas enlazadas simples)
<code>std::deque</code>	<code><deque></code>	TAD doble cola
<code>std::stack</code>	<code><stack></code>	TAD pila
<code>std::queue</code>	<code><queue></code>	TAD cola

Operaciones

- Tienen exactamente el mismo nombre que las que hemos visto a lo largo del curso:
 - `push_back()`
 - `push_front()`
 - `operator[]`
 - `begin()`
 - `etc.`



Algunas excepciones

- vector no implementa push_front() o pop_front().
Nos tendremos que buscar otro TAD. Porque tienen coste lineal respecto al número de elementos.
- list no implementa at() ni el operador [].
por la misma razón
- No tienen ninguna función display(), ni sobrecargan el operador <<.

crearlo nosotros O utilizar copy, lo veremos más adelante.

Ejemplo

```
int main() {  
    vector<int> v;  
    for (int i = 0; i < 10; i++) {  
        v.push_back(i * 3);  
    }
```

[0, 3, 6, 9, ...]

```
    cout << v.size() << endl;  
    int suma = 0;  
    for (int x : v) {  
        suma += x;  
    }
```

```
    cout << "Suma total: " << suma << endl;  
    return 0;  
}
```