

## Práctica 2

**Importante:** Los ficheros deben entregarse a través de web (por **Domjudge** y **Blackboard**).

Para los ejercicios se deberán entregar **3 ficheros** con nombres:

Burbuja.cpp, insercion.cpp y seleccion.cpp

Estos ficheros serán proporcionados a través de blackboard y deberán ser modificados.

**La fecha de entrega:** consultar la página de la actividad en blackboard

**(2 puntos) Burbuja:** Implementa un algoritmo que permita ordenar un array de enteros utilizando el algoritmo burbuja. Puedes encontrar una descripción de este algoritmo en la siguiente dirección:

[https://es.wikipedia.org/wiki/Ordenamiento\\_de\\_burbuja](https://es.wikipedia.org/wiki/Ordenamiento_de_burbuja)  
(Fíjate en la animación, la cual es muy ilustrativa).

El **input** constará de dos partes. En primer lugar, con formato "%d\n" se indicará la cantidad N de enteros que se introducirán a continuación. La segunda parte de la entrada será una lista de enteros separados

El **output** será una línea por cada iteración del algoritmo de ordenación en el que se mostrará una lista de números mostrando la ordenación parcial. **OJO:** Observad que existe un espacio después de cada número incluso en el caso del último.

Input:

Output:

10	8 7 6 5 4 3 2 1 0 9
9 8 7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0 8 9
	6 5 4 3 2 1 0 7 8 9
	5 4 3 2 1 0 6 7 8 9
	4 3 2 1 0 5 6 7 8 9
	3 2 1 0 4 5 6 7 8 9
	2 1 0 3 4 5 6 7 8 9
	1 0 2 3 4 5 6 7 8 9
	0 1 2 3 4 5 6 7 8 9

**(2 puntos) Inserción:** Implementa un algoritmo que permita ordenar un array de enteros utilizando el algoritmo de ordenamiento por inserción. Puedes encontrar una descripción de este algoritmo en la siguiente dirección:

[https://es.wikipedia.org/wiki/Ordenamiento\\_por\\_inserci%C3%B3n](https://es.wikipedia.org/wiki/Ordenamiento_por_inserci%C3%B3n)  
(Fíjate en la animación, la cual es muy ilustrativa).

El **output** será una línea por cada iteración del algoritmo de ordenación en el que se mostrará una lista de números mostrando la ordenación parcial. **OJO:** Observad que existe un espacio después de cada número incluso en el caso del último.

**(2 puntos) Selección:** Implementa un algoritmo que permita ordenar un array de enteros utilizando el algoritmo de ordenamiento por selección. Puedes encontrar una descripción de este algoritmo en la siguiente dirección:

(Fíjate en la animación, la cual es muy ilustrativa).

El **output** será una línea por cada iteración del algoritmo de ordenación en el que se mostrará una lista de números mostrando la ordenación parcial. **OJO:** Observad que existe un espacio después de cada número incluso en el caso del último.

[illegible]

## Algoritmos y estructuras de datos

---

**(4 puntos)** Para cada uno de los algoritmos de ordenación implementados se deberá incluir una pequeña descripción de su funcionamiento, sus precondiciones mediante `assertdomjudge` si las hubiera y el análisis de su complejidad temporal ( $T(n)$ ) y espacial ( $M(n)$ ) junto con su orden de complejidad en notación  $O()$  para ambas. Esta información deberá incluirse en la cabecera de cada función.