

Nom i Cognoms: _____ Firma: _____

Abans de començar:

- **CAMPUS VIRTUAL**: descarrega't el codi de la teva pràctica 2 del Campus Virtual i descomprimeix-lo.
- En aquest control s'ha d'utilitzar **NetBeans 8.2, C++ versió 11**.
- S'ha de lliurar al campus virtual **un únic fitxer zip** amb el vostre nom i cognoms. Per exemple, LisaSimpson_control_P2.zip que contindrà el codi dels dos exercicis. Cada exercici és un projecte NetBeans.

(2.5 PUNTS) EXERCICI 1: En el TAD ArrayDeque, afegireu i implementareu una nova funció anomenada:

void enqueueIfNotExist(const int) ;

Aquesta funció afegeix un element a l'inici del deque només si aquest element no està ja inserit en el deque. En el cas que ja estigui en el deque, no l'afegeix i retorna una excepció.

- Definiu el nou mètode en el TAD definit en el projecte de l'**exercici 1** de la pràctica. Es valorarà positivament que feu la millor implementació possible i tingueu en compte el cost computacional.
- Afegiu a l'inici del **main** el següent tros de codi, tot mantenint després el codi que vàreu fer per ArrayDeque, a la dreta teniu la sortida per pantalla del codi:

```
int main(int argc, char** argv) {
    ArrayDeque q(5);
    int elements[3] = {1, 2, 3};
    for (int i = 0; i < 3; i++) q.enqueueBack(elements[i]);
    q.print();
    q.enqueueIfNotExist(4); q.print();
    q.enqueueIfNotExist(5); q.print();
    try {
        q.enqueueIfNotExist(4);
    } catch ( TODO ) { // agafeu aquí la vostra excepció
        // imprimiu el missatge de l'excepció
    }
}
q.dequeueFront();
q.enqueueIfNotExist(7); q.print();
```

[1,2,3] [4,1,2,3] [5,4,1,2,3] Exception: Element exists in DEQUE [7,4,1,2,3]	SORTIDA PER PANTALLA
--	---------------------------------

(3 PUNTS) EXERCICI 2: : En el TAD LinkedDeque, afegireu i implementareu una nova funció anomenada:

`void enqueueAt(int& key, int pos);`

Aquesta funció afegeix un element (*key*) a la posició (*pos*) indicada del deque, tenint en compte que el front del deque és la posició 0 i els elements següents tenen posicions successives. Si *pos* és més gran que el nombre d'elements a la cua, s'afegirà l'element al final de tot.

- Definiu i implementeu la funció en el projecte NetBeans de l'**exercici 2** de la pràctica.
- Afegiu a l'inici del main el següent tros de codi, tot mantenint després el codi que vàreu fer per la LinkedDeque.

```
LinkedDeque q;  
int elements[5] = {1,2,3,4,5};  
  
for (int i = 0; i < 5; i++) { q.enqueueBack(elements[i]); q.print();}  
  
int elem = 5; q.enqueueAt(elem, 2); q.print();  
elem = 6; q.enqueueAt(elem, 7); q.print();  
elem = 7; q.enqueueAt(elem, 0); q.print();
```

- La sortida del main anterior és la següent:

```
[1]  
[1,2]  
[1,2,3]  
[1,2,3,4]  
[1,2,3,4,5]  
enqueueAt 5 in position 2  
[1,2,5,3,4,5]  
enqueueAt 6 in position 7  
[1,2,5,3,4,5,6]  
enqueueAt 7 in position 0  
[7,1,2,5,3,4,5,6]
```

IMPORTANT

- **No podeu usar les funcions del TAD LinkedDeque per implementar aquesta funció.**
- **Els recorreguts han de ser amb punters, no s'han de fer amb size.**