

Qüestions exercici 2:

- Has implementat la LinkedDeque amb templates? Sigui quina sigui la teva resposta, justifica el motiu pel qual has pres aquesta decisió.

Si . Hem implementat LinkedDeque amb templates, d'aquesta forma, implementem un codi genèric que es pot utilitzar amb qualsevol tipus de dada. Utilitzant templates no s'ha de reescriure codi per tal d'utilitzar-lo amb diferents tipus de dades.

- Tenint en compte la teva implementació del TAD LinkedDeque, indica per a cadascuna de les operacions del TAD LinkedDeque quin és el seu cost computacional teòric. Justifica la resposta.

Operació	Cost	Justificació
isEmpty	O(1)	Booleà, no fa cap recorregut.
insertFront	O(1)	Crea un nou node i reassigna els punters. No fa cap recorregut.
insertRear	O(1)	Crea un nou node i reassigna els punters. No fa cap recorregut.
deleteFront	O(1)	Assigna front a un nou node, reassigna punters i front, elimina el nou node. No fa cap recorregut.
deleteRear	O(1)	Assigna rear a un nou node, reassigna punters i rear, elimina el nou node. No fa cap recorregut.
print	O(n)	Tantes vegades com elements tingui LinkedDeque.
size	O(1)	Enter, no fa cap recorregut.
getFront	O(1)	És un getter, no fa cap recorregut.
getRear	O(1)	És un getter, no fa cap recorregut.

- Creieu que la classe Node hauria estat millor implementar-la amb encadenaments simples? Justifica la teva resposta.

Crec que no, ja que utilitzant encadenaments dobles com que sempre sabem l'anterior i el posterior d'un element, les implementacions de les operacions del TAD es simplifiquen considerablement. Per exemple al fer deleteRear no cal recórrer tot el deque per reassignar els punters ja que directament tenim anterior i posterior d'aquest.