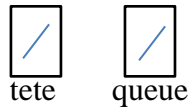


## **B1.1**

Au départ la file est vide :



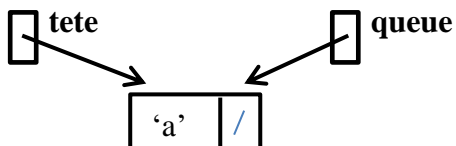
Test 1 : file.defile()

***FileVideException***

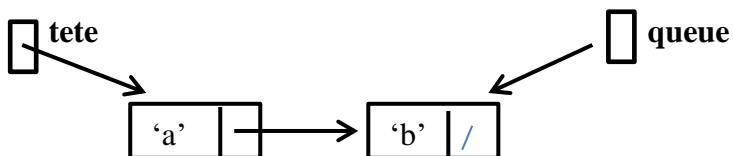
Test 2 : file.premier()

***FileVideException***

Test3 : file.enfile('a')

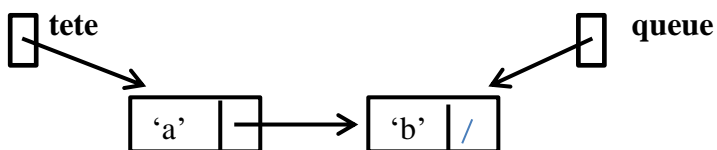


Test 4 : file.enfile('b')



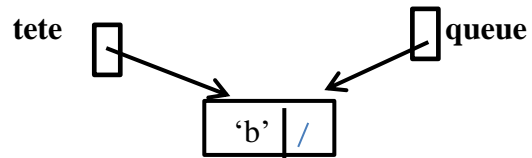
Test 5 : file.premier()

caractère renvoyé : a



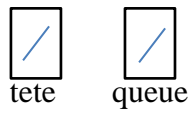
Test 6 : file.defile()

caractère renvoyé : a



Test 7 : file.defile()

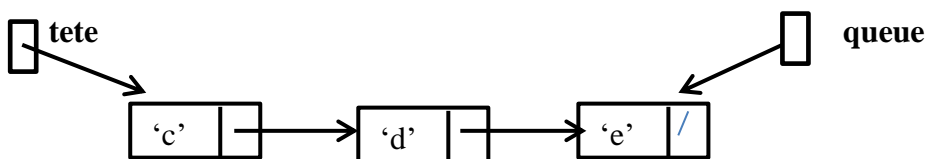
caractère renvoyé : b



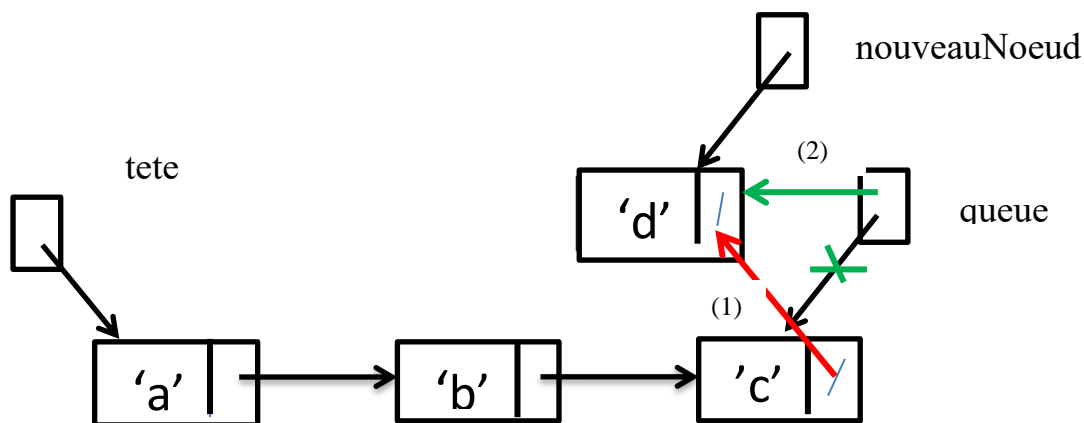
Test 8 : file.defile()

***FileVideException***

Test 9 : file.enfile('c');  
file.enfile('d');  
file.enfile('e');



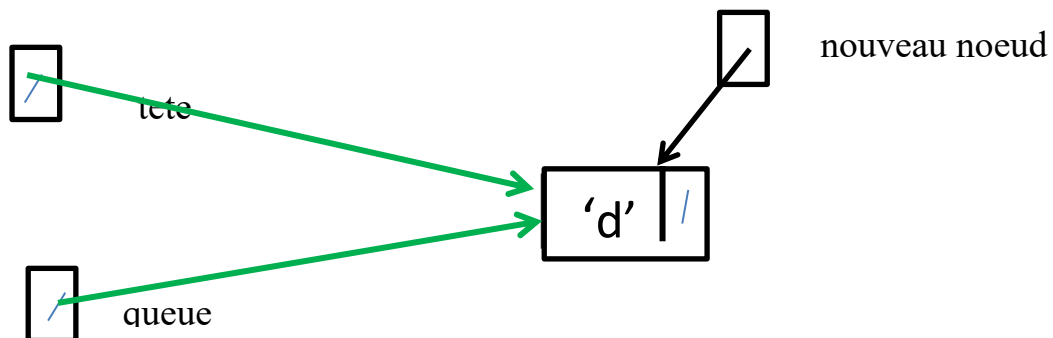
## B1.2



Instructions :

- (1) `queue.suivant = nouveauNoeud ;`
- (2) `queue = nouveauNoeud ;`

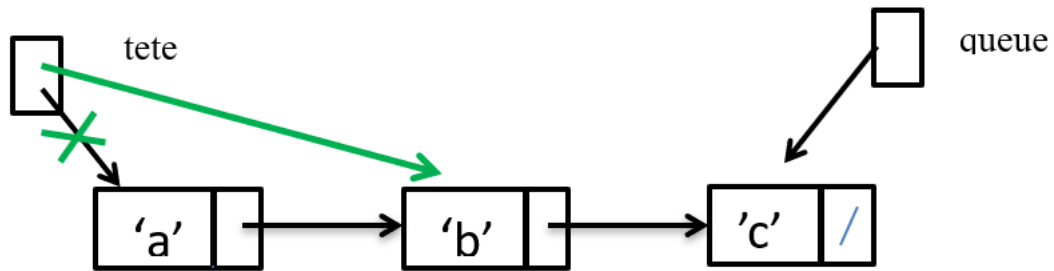
Attention, il est nécessaire de prévoir un cas particulier pour gérer le cas de la file vide :



Comme `queue` est *null*, l'instruction `queue.suivant = nouveauNoeud` provoquerait une exception de type *NullPointerException*.  
De plus, la tête aussi est nouvelle. Elle ne doit pas rester à *null*.

```
tete = nouveauNoeud ;  
queue = nouveauNoeud ;
```

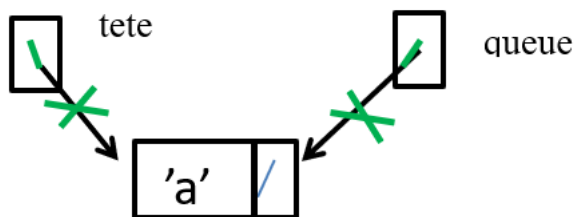
### B1.3



Instructions :

```
tete = tete.suivant ;
```

Attention, il est nécessaire de prévoir un cas particulier dans le cas où il ne reste qu'un nœud dans la chaîne.



`tete = tete.suivant` a comme effet de mettre la tête à *null* → OK  
Mais *queue* aussi doit être mis à *null* !

```
tete = null;  
queue = null;
```