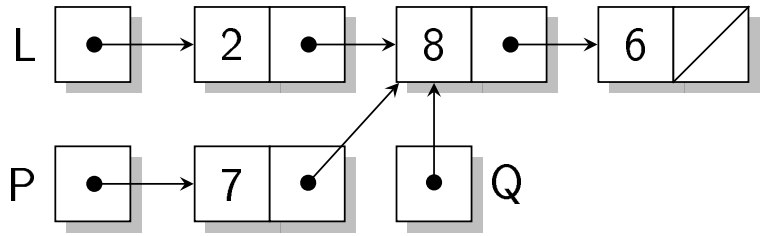
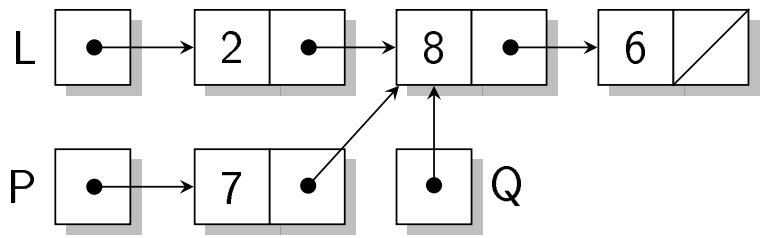
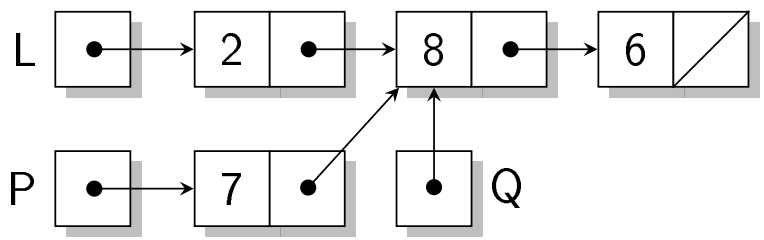
Exercice 2



1. $(Q \uparrow succ) \uparrow info \leftarrow P \uparrow info$

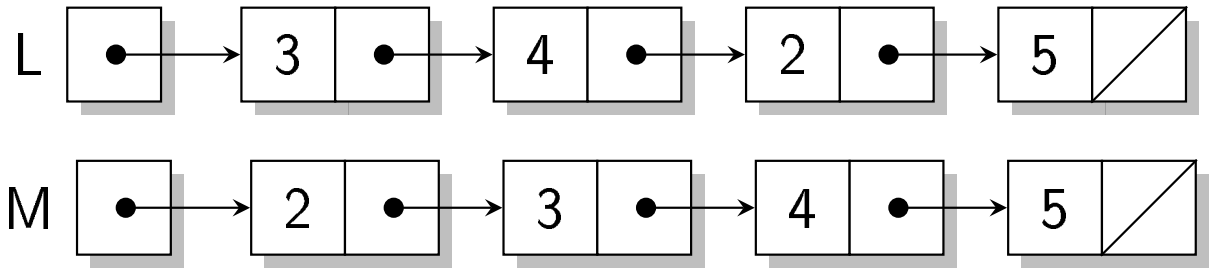


2. $(L \uparrow succ) \leftarrow q \uparrow succ$



3. insérerAprès$(P, P, 3)$

Exercice 3



Algo : estTrié(d L :liste) : booléen

si L=NULL ou L→succ=NULL **alors**

renvoyer vrai

sinon si (L↑info) > (L←succ)↑info **alors**

renvoyer faux

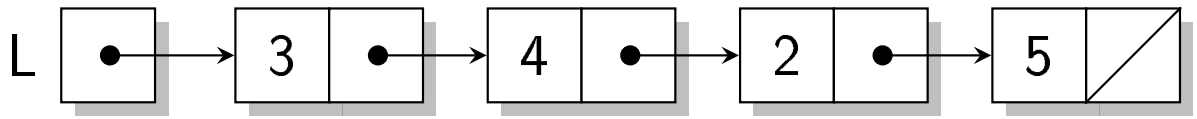
sinon

renvoyer estTrié(L↑succ)

On peut simplifier en

renvoyer L=NULL **ou** L→succ=NULL **ou** ((L↑info) ≤ (L←succ)↑info **et** estTrié(L↑succ))

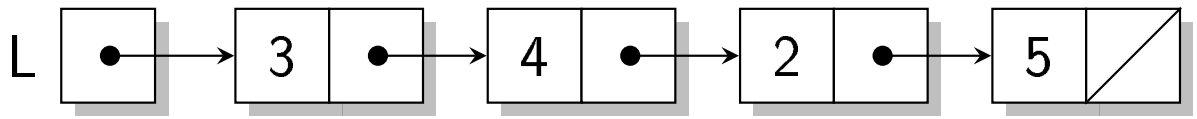
Exercice 4 - version récursive



Algo : adresseDernier(d L :liste) : Liste

```
si L↑succ=NULL alors
    renvoyer L
sinon
    renvoyer adresseDernier(L↑succ)
```

Exercice 4 - version itérative



Algo : adresseDernier(d L :liste) : Liste

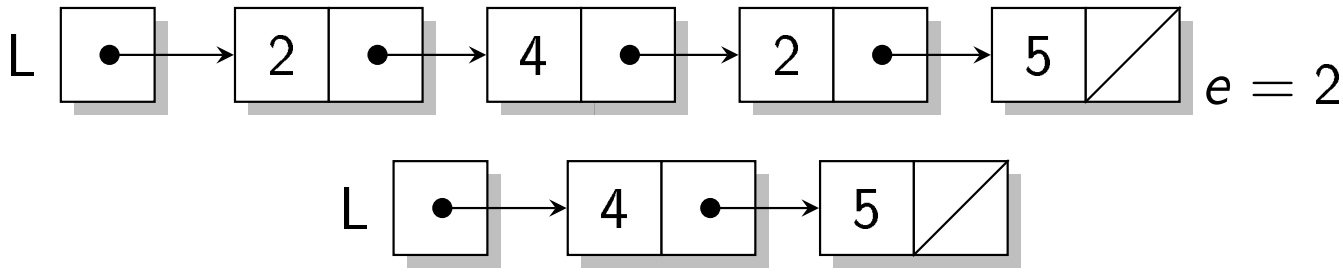
M←L

Tant que M↑succ≠NULL **faire**

M←M↑succ

renvoyer M

Exercice 5 - avec supprimer

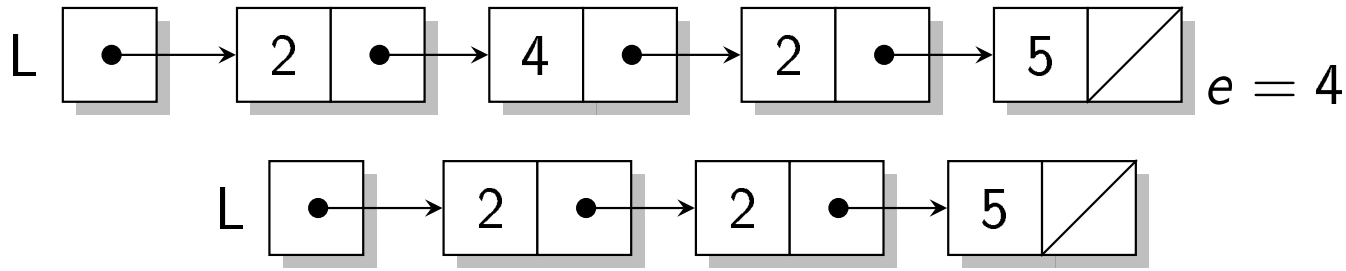


Algo : supprimeVal(dr L :liste, d e :entier)

```
M←L
Tant que M≠ NULL faire
    si M↑info=e alors
        supprimer(L,M)
    sinon
        M←M↑succ
```

Complexité ? $O(n^2)$

Exercice 5 - suppression directe



Algo : supprimeVal2(dr L :liste, d e :entier)

Tant que L $\neq$ NULL **et** L↑info=e **faire**

L $\leftarrow$ L↑succ

M$\leftarrow$L

Tant que (M↑succ)$\neq$ NULL **faire**

si (M↑succ)↑info=e **alors**

(M↑succ)$\leftarrow$(M↑succ)↑succ

sinon

M$\leftarrow$M↑succ