

## Algorithmique 1 : Devoir Surveillé 2

Pile et File

Durée : 60mn  
Sans documents

### Exercice 2.1 *Utilisation d'une pile*

Soit une chaîne de caractères dans laquelle le caractère # représente l'effacement du caractère précédent (touche backspace du clavier). Donner une fonction **Reduire** qui transforme la chaîne d'entrée en chaîne réduite (les # sont enlevés et les caractères effacés).

Par exemple si en entrée on a la chaîne **###abababab#cdef#**, le résultat sera la chaîne réduite **bababdef**.

### Exercice 2.2 *Utilisation d'une file*

On définit un ensemble  $A$  de la façon suivante :

- L'entier 1 appartient à  $A$ .
- Si  $i \in A$  alors  $2i + 1 \in A$  et  $3i + 2 \in A$ .

Donner une procédure **Elements** qui prend comme argument un entier  $n$  et qui, utilisant une file, affiche tous les éléments de  $A$  inférieurs à  $n$ . On ne demande pas de les afficher dans l'ordre croissant, et on considèrera comme correctes des solutions où un même élément est affiché plusieurs fois.

### Exercice 2.3 *Implémentation*

Soit l'implémentation du type abstrait **listeDC**, liste doublement chaînée, définie comme il suit :

```
type cellule= structure
    valeurElement: type_predefini;
pointeurPrecedent: ^cellule;
pointeurSuivant: ^cellule;
    finstructure;
type listeDC=structure
    premier: ^cellule;
    dernier: ^cellule;
    finstructure;
```

Et les primitives usuelles :

```
fonction premier(val L: listeDC): ^type_predefini;
fonction dernier(val L: listeDC): ^type_predefini;
fonction suivant(val L: listeDC, val P: ^type_predefini): ^type_predefini;
fonction precedent(val L: listeDC, val P: ^type_predefini): ^type_predefini;
fonction listeVide(val L: listeDC): boolean;
fonction creerListe(ref L: listeDC): vide;
fonction insererApres(val X: type_predefini, ref L: listeDC,
                    val P: ^type_predefini): vide;
fonction insererEnTete(val X: type_predefini, ref L: listeDC): vide;
fonction supprimerApres(ref L: listeDC, val P: ^type_predefini): vide;
fonction supprimerEnTete(ref L: listeDC): vide;
```

On suppose que les types `pile` et `file` sont implémentées par le type `listeDC`

1. écrire les primitives `empiler` et `depiler`.
2. écrire les primitives `enfiler` et `defiler`.