

Algorithmique 1

TD 3

1 Piles

Dans les exercices suivants on considère le type :

```
type_pile= pile de type_predefini;
```

défini en cours avec les primitives suivantes :

```
fonction valeur(ref P: pile de type_predefini): type_predefini;
fonction pileVide(ref P: pile de type_predefini): boolean;
fonction creerPile(ref P: pile de type_predefini):vide;
fonction empiler(ref P: pile de type_predefini, val v : type_predefini):vide
fonction depiler(ref P: pile de type_predefini):vide ;
```

Exercice 3.1

Evaluer à l'aide des primitives du type abstrait Pile la fonction suivante et donner le contenu de la pile après exécution.

```
fonction essaiPile():pile de char;
var P: pile de char;
var C:char;
  début
    créerPile(P);
    empiler(P, 'A');
    depiler(P);
    empiler(P, 'B');
    C=valeur(P);
    empiler(P, 'a');
    empiler(P,C);
    retourner(P);
  fin
finfonction
```

Exercice 3.2

On se donne trois piles P_1 , P_2 et P_3 , la pile P_1 contient une suite de nombres entiers positifs.

1. Ecrire un algorithme pour déplacer les entiers de P_1 dans P_2 de façon à avoir dans P_2 tous les nombres pairs au dessous des nombres impairs.

2. Ecrire un algorithme pour copier dans P_2 les nombres pairs contenus dans P_1 . Le contenu de P_1 après exécution de l'algorithme doit être identique à celui avant exécution. Les nombres pairs dans P_2 doivent être dans l'ordre où ils apparaissent dans P_1 .

Exercice 3.3

Utiliser une pile pour

1. évaluer une expression arithmétique postfixée codée sur un tableau de caractères, en supposant pour simplifier que tous les opérateurs sont binaires et limités à $+, -, *, /$.
2. transformer une expression arithmétique infixée en une expression arithmétique postfixée codée sur un tableau de caractères, en supposant pour simplifier que tous les opérateurs sont binaires et limités à $+, -, *, /$.

Exercice 3.4

Un problème fréquent d'un compilateur et des traitements de textes est de déterminer si les parenthèses d'une chaîne de caractères sont balancées et proprement incluses l'une dans l'une. Par exemple, la chaîne $((()) ()) ()$ est bien balancée et proprement écrite. Mais les chaînes $() ()$ (ou $())$ ne le sont pas.

Ecrire une fonction :

1. qui retourne TRUE si une chaîne de caractères est proprement écrite et bien balancée, et FALSE sinon.
2. qui retourne la position de la première parenthèse qui déroge à cette règle si la chaîne n'est pas bien écrite et bien balancée.

2 Files

Dans les exercices suivants on considère le type :

```
type_file= file de type_predefini;
```

défini en cours avec les primitives suivantes :

```
fonction valeur(ref F: file de type_predefini): type_predefini;  
fonction fileVide(val F: file de type_predefini): boolean;  
fonction creerFile(F: file de type_predefini):vide;  
fonction enfiler(ref F: file de type_predefini, val v : type_predefini):vide;  
fonction defiler(ref F: file de type_predefini):vide ;
```

Exercice 3.5

Evaluer à l'aide des primitives du type abstrait File la fonction suivante et donner le contenu de la pile après exécution.

```
fonction essaiFile():file de char;  
var F: file de char;  
var C:char;  
  début  
    créerFile(F);
```

```

    enfiler(F, 'A');
    enfiler(F, 'B');
    enfiler(F, 'C');
    defiler(F);
    C=valeur(F);
    enFiler(F, 'a');
    defiler(F);
    enFiler(F, 'b');
    enFiler(F,C);
    retourner(F);
fin
finfonction

```

Exercice 3.6

Soit L une liste d'entiers strictement croissante et un entier N . Ecrire une fonction `compter` qui dénombre les couple $(x, x+N)$ avec x et $x+N$ présents dans la liste.

Exercice 3.7

Un palindrome est une chaîne de caractères qui se lit de la même manière de gauche à droite et de droite à gauche. En utilisant un nombre fixe de piles et de files, des primitives du type PILE et FILE, et un nombre fixe de variables de type INT et CHAR, écrire un algorithme qui détermine si une chaîne de caractère est un palindrome. On suppose que la chaîne de caractère est passée en paramètre dans un tableau qui ne sera parcouru qu'une fois. L'algorithme doit donner en sortie TRUE ou FALSE selon le cas.