Université Bordeaux 1 Licence Informatique 2007-2008

Algorithmique 1

Contrôle continu - Pile et file

1 Primitives de Pile

Les primitives du type abstrait `Pile` sont :

```
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): booleen;
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 ;
```

2 Primitives de Files

Les primitives du type abstrait `File` sont :

```
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(ref F: file de type_predefini): booleen;
fonction creerFile(ref 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 3.1 *Implémentation d'une File avec deux Piles*
Dans cette exercice, on veut implémenter les primitives du type abstrait `File` en utilisant deux piles :

1. Donner l'implementation du type *type_file.*
2. Ecrire l'ensemble des primitives du type abstrait `File` en considérant l'implémentation avec deux `Piles`
3. Si la File contient N éléments, expliquer et donner la compléxité des primitives `enfiler` et `défiler`.

Exercice 3.2 *Evaluation d"une expression post-fixée*
Utiliser une pile pour

1. Evaluer une expression arithmétique postfixée codée sur un tableau de caractères , en supposant pour simplifier que
   - tous les opérateurs sont binaires et limités à `+,-,*,/` ,
   - on utilise uniquement des nombres sur un caractère,