

TD 2, Analyse amortie

Exercice 1. Analyse amortie pour des structures à croissance exponentielle

Une suite de n opérations est effectuée sur une structure de données. La i -ème opération coûte i si i est une puissance exacte de 2, et 1 sinon. Déterminer le coût amorti de l'opération par la méthode de l'agrégat.

Exercice 2. Analyse amortie dans un ABR

Soit A un arbre binaire de recherche, contenant n éléments et de profondeur maximum k . Soient les fonctions:

- **Fonction** Premier($\underline{E} A : \text{ABR}$) : Noeud qui retourne le Noeud de l'arbre dont la valeur est la plus faible.
 - **Fonction** Suivant($\underline{E} A : \text{ABR}, \underline{E} N : \text{Noeud}$) : Noeud qui retourne le Noeud de l'arbre dont la valeur est juste supérieure à la valeur du noeud N ou Null sinon.
1. Ecrire le pseudo-code des fonctions *Premier* et *Suivant*, en utilisant les fonctions classiques des ABR: *Racine*, *Gauche*, *Droite*, *Pere*.
 2. Soit le programme suivant. Que fait-il ? Quelle est la complexité en pire cas de l'opération *Suivant* ? En déduire une borne supérieure sur la complexité en pire cas du programme *Parcours*.

```
1 Action Parcours( $\underline{E} A : \text{Arbre}$  ) ;  
  Var :  $N : \text{Noeud}$  ;  
2 début  
3    $N \leftarrow \text{Premier}( A )$ ;  
4   Tant Que  $N \neq \text{Null}$  Faire  
5      $\text{Affiche}( \text{Valeur}( A, N ) )$  ;  
6      $N \leftarrow \text{Suivant}( A, N )$ ;  
7 fin
```

3. Quelle est maintenant le coût amorti de chaque appel à *Suivant* ? Utilisez la méthode des agrégats.

Exercice 3. Analyse amortie d'une file implémentée à l'aide de deux piles

Montrer que l'on peut implémenter une file avec deux piles ordinaires, de telle manière que le coût amorti de chaque opération *Enfiler* et *Défiler* soit $O(1)$.