

	ANNÉE UNIVERSITAIRE 2007/2008 DEUXIÈME SESSION DU PRINTEMPS	
	Parcours : CSB6 Code UE : INF352 Épreuve : Algorithmique 3 Date : Le juin 2008 Heure : Durée : 3h Documents : Tous documents obtenus de manière légale autorisés. Votre voisin n'est pas un document. Un téléphone portable non plus. Épreuve de M. Robert Strandh	

Le barème est donné à titre **indicatif**.

Le sujet comporte 2 pages.

Si vous pouvez vous servir directement d'une fonction traitée en cours, vous n'avez pas besoin de répéter son implémentation ici.

Rappel : soyez bref et clair. Personne n'est impressionné par la quantité.

Exercice 1 (10pts)

On considère un type abstrait appelé *poche* utilisé pour stocker des objets arbitraires. Il est défini par les opérations suivantes :

Opération	Description
insérer	rajouter un objet à la poche
membre ?	déterminer si un objet quelconque est dans la poche
supprimer	renvoyer et supprimer un objet arbitraire de la poche
vide ?	vérifier si la poche est vide

L'opération **insérer** a comme précondition que l'objet ne doit pas déjà être dans la poche (chose que l'on peut vérifier grâce à l'opération **membre ?**). L'opération **membre ?** détermine si l'objet *identique* (que l'on peut tester avec l'opérateur $=$) à celui passé en paramètre est dans la poche. L'opération **supprimer** renvoie un objet et le supprime de la poche. Pour l'implémentation de l'opération **supprimer**, on tiendra compte de la structure de données de la poche afin de choisir de façon la plus adéquate possible l'objet à renvoyer.

On souhaite implémenter la poche par une liste simplement chaînée.

a. Donner l'implémentation (en utilisant un langage algorithmique proche de ce qui a été utilisé en cours) de l'ensemble des opérations du tableau ci-dessus, lorsque la poche est implémentée par une liste simplement chaînée.

b. Pour chaque opération, donner la complexité asymptotique au pire. Justifier la réponse.

Pour implémenter les opérations de la poche, on dispose également de l'implémentation d'un autre type abstrait : *dictionnaire* avec les opérations **insérer** pour insérer un objet dans le dictionnaire, **membre ?** pour déterminer si un objet est dans le dictionnaire et **supprimer** pour supprimer un objet du dictionnaire.

c. Montrer (en utilisant un langage algorithmique proche de ce qui a été utilisé en cours) comment implémenter les opérations de la poche avec une combinaison d'une liste simplement chaînée et un dictionnaire.

d. Quelle est la complexité des opérations de la poche si l'on suppose que les opérations sur le dictionnaire sont de complexité $O(1)$? Justifier la réponse.

Exercice 2 (10pts)

On utilise un arbre 2-3 pour implémenter une *séquence éditable*.

a. Donner un algorithme (en utilisant un langage algorithmique proche de ce qui a été utilisé en cours) pour concaténer deux séquences éditables représentées par des arbres 2-3 de *même hauteur*.

b. Donner un algorithme (en utilisant un langage algorithmique proche de ce qui a été utilisé en cours) pour concaténer deux séquences éditables représentées par des arbres 2-3 de *hauteur arbitraire*.

c. Quelle est la complexité de ce dernier algorithme ? Justifier la réponse.

FIN