

T.D. Algorithmique n° 14

Séquences chaînées – Pointeurs

Exercice 1 : File d'attente chez un médecin

Réécrire les algorithmes de gestion de la file d'attente d'un médecin, en représentant la file d'attente par une liste chaînée. Chaque doublet de la liste sera formé d'un champ de type Patient et d'un pointeur sur le doublet suivant. Pour rappel, le type Patient composé d'un nom et d'un numéro de sécurité sociale :

Patient type agrégat **nom** : chaîne, **numéroSS** : chaîne agrégat

a) Définir le type Doublet

b) Ecrire la classe FileDAttente, c'est-à-dire les variables de la partie privée qui permettent de définir une file d'attente, un constructeur qui crée une file vide et les méthodes suivantes :

action arrivéePatient(consulté **x** : Patient, élaboré ok: booléen)

// Effet : ajoute le patient x à la file. Le booléen est à faux en cas d'erreur ;

action appelPatient(élaboré ok : booléen)

// Effet : supprime le premier patient de la file d'attente et affiche son nom ok est à faux en cas d'erreur

action abandonPatient(consulté **n** : chaîne, élaboré ok : booléen)

// Effet : supprime le patient de nom n de la file ; ok est à faux en cas d'erreur

fonction nombreDePatients() → entier ≥ 0

// renvoie le nombre de patients présents dans la file d'attente

fonction estPresent(num : chaîne) → booléen

// renvoie vrai si le patient de numéro num est présent dans la file d'attente

Exercice 2 : Concaténation de séquences chaînées

Écrire un algorithme qui crée la séquence S, concaténation de deux séquences d'entiers données S1 et S2. Les séquences sont représentées sous forme chaînée. Étudier le problème selon les deux hypothèses suivantes :

1) les éléments des séquences données sont dupliqués : S1 et S2 ne sont pas modifiées par la concaténation.

2) les éléments des séquences données ne sont pas dupliqués : S1 et S2 sont perdues à l'issue de la concaténation qui est réalisée uniquement en termes de modifications de pointeurs.

Exercice 3 : Union, intersection, interclassement

On considère deux séquences chaînées d'entiers, ordonnées selon l'ordre croissant. Ecrire les algorithmes qui réalisent l'union, l'intersection et l'interclassement des deux séquences. Étudier ces problèmes selon les deux hypothèses suivantes :

1) les éléments des séquences données sont dupliqués pour former la séquence résultat : S1 et S2 ne sont pas modifiées par l'opération d'union, d'intersection ou d'interclassement.

2) les éléments des séquences données ne sont pas dupliqués : S1 et S2 sont perdues à l'issue l'opération d'union, d'intersection ou d'interclassement qui est réalisée uniquement en termes de modifications de pointeurs et d'éventuelles destruction de doublets.

