

DS2 - Algorithmes et Structures de Données

Mardi 11 Janvier 2022

Durée 3H – Documents non autorisés (autres que le résumé distribué)

1. Tableaux – 3 pts

Écrire en pseudo-langage une procédure `diff` qui prend en entrée deux tableaux d'entiers `t1` et `t2` (non triés et sans doublon) de taille respective `n1` et `n2` et qui renvoie le tableau `t3` de taille `n3` représentant la **différence symétrique** de `t1` et `t2` (ensemble des éléments appartenant soit à `t1`, soit à `t2`, **mais pas au 2**). Cette procédure peut appeler d'autres fonctions ou procédures que vous écrirez.

Const max = 30

Type tab = tableau[1..max] d'entier

Procédure diff(E t1,t2 : tab, E n1, n2 : entier ; S t3 : tab, S n3 : entier)

Exemple : t1=(6,20,12,1000,8) ; t2=(2,8,6,7,12) ; t3=(20,1000,2,7)

2. Récursivité – 4 pts

Ecrire en pseudo-langage une **fonction récursive** qui renvoie le plus petit entier d'un tableau `t` non trié entre les indices `i` et `j` par la méthode de dichotomie : on sépare le tableau en 2 sous-tableaux à chaque appel de la fonction, et le plus petit entier du tableau est le min des plus petits entiers des 2 sous-tableaux.

Const max = 30

Type tab = tableau[1..max] d'entier

Fonction ppentier(t : tab ; i,j : entier) : entier

3. File de priorités – 4 pts

On souhaite gérer la file d'attente des documents à traiter par un service. Cette file est représentée par une liste chaînée de documents dont on connaît le nom du fichier `nomdoc`, sa priorité `prio` (de 1 à 3 selon l'urgence du traitement du document, 1 étant le plus urgent), le type de traitement à effectuer `type` et le nom de l'utilisateur `util` qui demande ce traitement. Cette liste chaînée est toujours triée dans l'ordre croissant des priorités et, pour une même priorité, dans l'ordre des arrivées. Le premier document de la file (en tête) est le premier document arrivé avec la priorité 1 et le dernier (en queue) est le dernier arrivé avec une priorité 3.

```
Type doc = Enregistrement
    nomdoc, util, type : chaîne
    prio : entier
    suiv : ^doc
FinEnregistrement
file = Enregistrement
    tete, queue : ^doc
FinEnregistrement
```

Ecrire en pseudo-langage une procédure `ajoute` qui ajoute un document de nom `n`, de l'utilisateur `u`, de type `t` et de priorité `p`, dans la file `f`. **Attention aux cas particuliers et à l'optimisation de l'algorithme.**

Procédure ajoute(E n, u, t : chaîne, E p : entier ; E/S f : file)

4. Pointeurs : recettes de cuisine – 9 pts

On souhaite représenter un livre de recettes de cuisine par une liste chaînée de recettes avec pour chacune la liste chaînée des ingrédients qu'elle utilise. Un ingrédient est donné par son nom `nomi` avec la quantité `qté` et l'unité `unité` utilisées (un ingrédient sera toujours donné dans la même unité). Une recette est donnée par son nom `nomr` et son type de plat `type` (« entrée », « plat » ou « dessert »).

```
Type  ingre = Enregistrement
        nomi, unité : chaîne
        qté : réel
        suiv : ^ingre
        FinEnregistrement
recette = Enregistrement
        nomr, type : chaîne
        pingre: ^ingre
        suiv : ^recette
        FinEnregistrement
livre = ^recette
Var l : livre
```

4.1. Faire un dessin de la structure de données `l` avec les recettes suivantes (type et ingrédients entre parenthèses) : `u` = unité

- crêpes (dessert, farine 500 g, œuf 4 u, beurre 20 g, lait 1 l)
- marbré-chocolat (dessert, farine 250 g, beurre 100 g, œuf 3 u, sucre 150 g, levure 10 g, chocolat-noir 200 g, lait 20 cl)
- lasagnes (plat, pâtes-lasagnes 500 g, bœuf-haché 1 kg, sauce-tomate 1 l, oignon 2 u, ail 2 u, huile-olive 3 cl, lait 0.5 l, farine 100 g, beurre 10 g, sel 1 g)
- tzatziki (entrée, concombre 1 u, fromage-blanc 1 kg, huile-olive 5 cl, ail 2 u, sel 1 g)
- brownie-chocolat (dessert, farine 100 g, beurre 150 g, œuf 3 u, sucre 150 g, noix 100 g, chocolat-noir 200 g).

4.2. Ecrire en pseudo-langage une fonction qui crée la liste de toutes les recettes du livre `l` de type « dessert » qui contiennent du chocolat-noir.

Fonction `donne-choco(l : livre) : livre`

4.3. Ecrire en pseudo-langage une fonction qui crée la liste des ingrédients et, pour chacun d'eux, les recettes qui le contiennent. **Définir un nouveau type de liste et faire un dessin.**

4.4. Un menu est décrit dans un fichier texte dont chaque ligne correspond au nom d'une recette. Ecrire en pseudo-langage une procédure qui crée, à partir du fichier menu en entrée, un nouveau fichier texte avec la liste des courses à faire pour réaliser toutes ces recettes : chaque ligne est un ingrédient avec sa quantité et son unité (chaque ingrédient n'apparaît qu'une fois dans le fichier). L'espace est le séparateur de mot.

Exemple : **fichier en entrée**

```
tzatziki
lasagnes
crêpes
```

fichier en sortie

```
concombre 1 u
fromage-blanc 1 kg
huile-olive 8 cl
ail 4 u
sel 1 g
pâtes-lasagne 500 g
bœuf-haché 1 kg
sauce-tomate 1 l
oignon 2 u
lait 1.5 l
farine 600 g
beurre 30 g
œuf 4 u
```