

Algorithmique et structure de données 1

L2 2022-2023
Devoir Surveillé 1A-X

- Notes de cours et de TD/TP autorisées - Calculatrice autorisée - Durée : 1h20 - si vous répondez sur le sujet, insérer-le dans votre copie avec nom et prénom ou/et num étudiant - par défaut aucune justification n'est demandée - chaque exercice réussi rapporte le même nombre de points

Pour les exercices impliquant une structure en liste chaînée on utilisera pour chaque cellule la structure suivante :

```
1 typedef struct s_cellule_t
2 {
3     int val; // valeur contenue dans la cellule
4     struct s_cellule_t* suiv; // adresse de la cellule suivante
5 } cellule_t;
6 // la liste vide sera representee par la valeur NULL
```

Exercice 1. Analyse

Voici une fonction `f`

```
1 int f(int n){
2     int i, j;
3     int r=0;
4     for (i=0; i<n; i++)
5         for (j=0; j<n; j++)
6             r++;
7     return r;
8 }
```

1. La fonction se termine-t-elle pour n'importe quelle entrée ? Justifier
2. Que renvoie la fonction ?
3. Donner sa complexité en temps et en espace dans le pire des cas.

Exercice 2. Croissant

Classez les suites suivantes de la plus dominée à la plus dominante. Indiquer les suites qui sont dans la même classe.

— n	— $(\log n)^{345}$	— $\sqrt{n} \times \log n$	— $\frac{n^2+n}{n+1}$
— $\sqrt{n} \times \sqrt{n}$	— $n^{0,001 \cdot n}$	— n^{45}	

Exercice 3. Recherche liste

Écrire une fonction `int index(cellule_t* liste, int val)` qui renvoie le numéro de la première cellule de la liste chaînée `liste` (voir structure en début de sujet) dans laquelle se trouve la valeur `val`.

Détail : `index` doit renvoyer 0 si `val` se trouve dans la première cellule, 1 si `val` se trouve dans la seconde mais pas dans la première, ..., et -1 si `val` n'est pas présente pas dans la liste `liste`.

Exercice 4. *Shuffle liste*

Écrire une fonction `void shuffle(cellule_t** listeD, cellule_t** listeS)` qui << *entrelace* >> (voir exemple) les deux listes chaînées `listeD` `listeS` en une nouvelle liste. La méthode `shuffle` n'a pas besoin d'allouer de la mémoire pour de nouvelles cellules, elle se contente de modifier les cellules existantes.

La structure de `cellule_t` est donnée en début de sujet.

Exemple : si avant `listeD`: 1 3 5 7 9 et `listeS`: -2 -4 -6 alors après appel à `shuffle` on aura `listeD`: 1 -2 3 -4 5 -6 7 9 et `listeS`: vide

Exercice 5. *Caisse*

Un caissier, amateur de programmation, se demande toujours comment rendre la monnaie en utilisant le moins de pièces/billets possibles parmi les pièces de sa caisse.

Il souhaite écrire la fonction `int renduMonnaie(int n, int* tab, int R)` qui renvoie le minimum du nombres de pièces/billets pour faire la somme `R`. Les valeurs de chaque pièce/billet disponible sont stockées dans le tableau `tab` de taille `n`. Si on ne peut faire la somme `R` avec les pièces disponibles la fonction doit renvoyer `-1`.

Écrivez le corps de la fonction `renduMonnaie` pour lui.