

Langage C (LC4)
Examen de rattrapage du vendredi 27 juin 2014
Durée : 2 heures 30 minutes

Les exercices sont indépendants et peuvent être traités dans n'importe quel ordre.

Exercice 1. *Quelques questions simples*

1.1. Écrire la définition d'un tableau `t` de cinq entiers dont les trois premiers éléments ont comme valeur respective 3, 5 et -1.

1.2. Écrire la définition d'une structure `st` contenant les champs suivants :

- deux champs entiers `a` et `b` ;
- un champ réel `r` ;
- un tableau de 16 caractères.

Comment s'écrit la définition de deux variables `x` et `y` ayant cette structure.

1.3. Écrire le prototype d'une fonction `f` prenant en paramètre un entier et un pointeur sur un flotant et retournant un pointeur sur caractère.

1.4. Écrire le code d'une fonction `plusGrand` ayant en paramètres quatre entiers et renvoyant une structure `couple` de deux entiers dont le premier champ est le plus petit des 4 nombres donnés en paramètres et le second le plus grand.

1.5. Écrire le code d'une fonction `copier1` ayant en paramètres deux tableaux `t1` et `t2` à une dimension et la taille `n` de ces tableaux et qui effectue la copie de `t1` dans `t2`.

1.6. Écrire le code d'une fonction `copier2` ayant en paramètres deux tableaux `t1` et `t2` à deux dimensions et les entiers `n1` et `n2` correspondant aux dimensions de ces tableaux et qui effectue la copie de `t1` dans `t2`.

Exercice 2. *Exercice sur les listes chaînées.*

2.1. Donner la définition d'un type `noeud_t` correspondant à un noeud d'une liste chaînée de nombres entiers et celle d'un type `liste_t` correspondant à un pointeur sur un tel noeud.

2.2. Écrire une fonction de prototype

```
liste_t ajouterEnTete(liste_t p, int n);
```

qui insère le nombre `n` au début de la liste dont la tête est pointée par `p`.

2.3. Écrire une fonction de prototype

```
liste_t ajouterEnFin(liste_t p, int n);
```

qui insère le nombre `n` à la fin de la liste dont la tête est pointée par `p`.
La fonction renverra le pointeur sur la tête de la liste après cette insertion.

2.4. Écrire une fonction de prototype

```
void afficherListe(liste_t p);
```

qui affiche à l'écran la suite des entiers dans la liste chaînée dont la tête est pointée par `p`.

Pour répondre à cette question, on utilisera les fonctions `sorties` précédemment.

2.5. Écrire un programme (fonction `main`) qui

- lit un entier `n` ;
- crée la liste chaînée correspondant à la suite d'entiers

`1 2 3 ... (n-1) n`

- affiche le contenu de la liste chaînée construite.

Exercice 3. On considère les définitions suivantes :

```
int t[3][5] = {{1,2,3,4,5}, {6,7,8,9,10}, {11,12,13,14,15}};  
int *p = &t[2][3], *q = &t[1][2];  
int *x = t[1], *y = t[2];  
int (*a)[5] = &t[1], (*b)[5] = &t[2];
```

et on suppose que, après leur réalisation, l'instruction

```
printf("%ld %ld %p\n", sizeof(void *), sizeof(int), t);
```

affiche 4 4 0x3740

Pour chacune des expressions suivantes, dire si elle est correcte et si c'est le cas quels sont son type et