

Contrôle Continu

1h00 - Documents autorisés

Nota : vous sauvez vos fichiers **.c**, **.h** et **Makefile** dans un répertoire dédié à l'examen (par exemple **CC2**)

Exercice 1 : une suite récurrente (5 points)

On considère la suite suivante :

$$U_1 = 1$$

$$U_2 = 2$$

$$U_n = (2U_{n-1} + U_{n-2})/2 \text{ définie pour } n > 2$$

1. Ecrire une fonction en langage C **récurive** qui retourne le $n^{\text{ième}}$ élément de la suite (**3 points**)
2. Ecrire un programme qui calcule et affiche les termes inférieur ou égaux à un nombre entré par l'utilisateur. (**2 points**)

Exercice 2 : et des arbres (14 points)

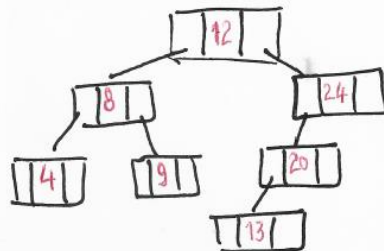
On considère des **arbres composés** de cellules. Chaque cellule de l'arbre possède trois champs :

1. un champ **val**, correspondant à la valeur pour stocker une valeur entière,
2. un champ **fg**, « fils gauche » pour stocker l'adresse de la cellule suivante
3. et un champ **fd**, « fils droit » pour stocker l'adresse de la cellule suivante.

- Définir les structures **s_arbre** et **s_cellule** (**2 points**)

On s'intéresse ici à stocker puis à retrouver des données entrées dans un arbre dans leur ordre d'apparition.

Par exemple, l'arbre correspondant aux données entrées suivantes est correct : 12, 8, 4, 9, 24, 20, 13



- Créer le fichier **arbre.h** correspondant en prenant soin de cacher les implémentations des structures utilisées (**2 points**)
- Dans le fichier **arbre.c**, écrire la fonction **ajouter(entier x, arbre A)** qui ajoute une cellule (au bon endroit) dans A, et qui renvoie le nouvel arbre (**3 points**)
- Dans le fichier **arbre.c**, écrire une fonction **trouver(entier x, arbre A)** qui renvoie vrai si il existe une cellule dans A contenant la valeur x, et faux sinon. (**3 points**)
- Dans le fichier **arbre.c**, écrire une fonction **supprime(arbre A)** qui supprime toutes les données contenues dans l'arbre (**2 points**)
- Etant donné un tableau d'entiers T contenant n valeurs entières non triées, écrire une fonction **main** dans le fichier **exercice2_cc2.c** qui construit un arbre avec toutes les valeurs de T (**2 points**).

Nota : vous pouvez ajouter autant de fonctions que vous souhaitez.

Exercice 3 : écrire un Makefile (1 point)

- Ecrire le fichier **Makefile** associé à l'exercice 2

A l'issue du contrôle continu, vous enverrez vos fichiers à l'adresse [mél Philippe.Truillet@irit.fr](mailto:Philippe.Truillet@irit.fr) avec comme sujet : **[CC2 L2 CUPGE] – nom/prénom**