UNIVERSITÉ
CÔTE D'AZUR

Université Côte d'Azur
Programmation C

L2 Informatique

Nom
Prénom
Né(e) à
Le

Ne surtout pas rabattre le triangle grisé. Il est là simplement pour faire sérieux. Dans tous les cas, votre copie ne sera pas anonyme car c'est le même enseignant qui corrige et qui rentre les notes.

Partiel : Programmation C

20 mars 2024

Durée : 1 heure 15 minutes
Aucun documents autorisés. Il est interdit d'accéder à internet.

Note

Toutes les questions sont indépendantes.
Tous les codes devront être écrits en **Langage C ANSI**. La notation est donnée à titre indicatif.
Nombre de pages : 6

Exercices divers (7 points)

1. Écrire une fonction `int puissance(int a, int b)` qui calcule et renvoie la valeur de a^b.

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

2. Écrire une fonction `int update_max(int *a, int b)` qui prend en argument deux entiers : le premier, « `a` », sous forme de pointeur, et le second, « `b` », en tant que valeur. La fonction calculera le maximum des entiers « `a` » et « `b` » et le stockera à l'adresse pointée par « `a` ».

On renverra `-1` si la valeur pointée par « `a` » était déjà la plus grande ; on renverra `0` si les deux valeurs étaient égales ; enfin, on renverra `1` si la valeur de `b`, étant plus grande, « `a` » s'est vu affecter la valeur de « `b` ».

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

3. **En utilisant la fonction précédente**, écrire une fonction `int calcul_max(int tab[], int taille)` qui calcule et renvoie le maximum d'un tableau de notes (comprises entre 0 et 20), la taille du tableau étant donnée en paramètre.

4. Écrire une fonction `char * concatener(char *a, char *b)` qui renvoie une nouvelle chaîne obtenue en concaténant les deux chaînes données en paramètre. On pourra utiliser la fonction `strlen` de la bibliothèque standard.

Problème (13 points)

Afin d'améliorer le bien-être de la promotion, on cherche à recenser et étudier les différentes relations d'ordre sentimentales qui se nouent entre les étudiant·e·s.

Première partie : modélisation des étudiants et de leurs amours

1. Créez un type `enum cursus` qui code les différents parcours des étudiants suivant l'UE de programmation C. Les cinq parcours étant : `INFO`, `IA`, `DLMI`, `MIAGE` et `ST`.

.....

.....

.....

2. Écrire une fonction `int licence_info(enum cursus c)` qui renvoie Vraie si le cursus mène au diplôme de la licence Informatique (c'est-à-dire les parcours Info et DL MI) et Faux sinon. Rappel, il faudra renvoyer les valeurs correspondant aux booléens en C.

.....

.....

.....

.....

.....

3. Définir un type `etudiant` comme étant une structure contenant quatre attributs :
   - un attribut `parcours` de type `enum cursus`
   - un attribut `nom` de type chaîne de caractères
   - un attribut `prénom` de type chaîne de caractères
   - un attribut `couple` du type pointeur vers un autre étudiant.

.....

.....

.....

.....

.....

.....

.....

4. Écrire une fonction `etudiant inscription(char *prenom, char *nom, enum cursus parcours)` qui renvoie un nouvel étudiant. Par défaut le champ `couple` sera initialisé à `NULL` pour représenter le célibat.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

5. Écrire une fonction `void coup_de_foudre(etudiant *alice, etudiant *bob)` qui met en couple les deux étudiants donnés en paramètre. La fonction ne renverra rien et se contentera de mettre à jour les champs correspondants.

.....

.....

.....

.....

.....

Seconde partie : stockage de la promotion

On va représenter la promotion par un tableau avec une capacité maximale et un indice de fin de tableau représentant l'indice de la première case disponible.

6. Écrire une structure `promo` contenant trois champ :
   - un champ `tab` représentant un tableau (alloué sur le tas) contenant des pointeurs vers le type `etudiant`.
   - un champ `taille` représentant la capacité maximale du tableau.
   - un champ `haut` représentant le plus grand indice libre (par défaut à 0).

.....

.....

.....

.....

.....

.....

7. Écrire une fonction `promo new_promo(int n)` qui renvoie une nouvelle promotion vide mais dont la taille (c'est-à-dire la capacité maximale) vaut `n`.

.....

.....

.....

.....

.....

.....

.....

.....

8. Écrire une fonction `int push(promo *p, etudiant *e)` qui ajoute un étudiant à la promo p. La fonction renverra `-1` s'il le tableau est déjà rempli et `0` sinon.

.....

.....

.....

.....

.....

.....

.....

.....

Troisième partie : statistiques et exclusion des étudiants au comportement immoral

9. Afin d'encourager une politique de diversité, écrire une fonction `int nombre_couples_mixtes(promo p)` qui renvoie le nombre de couples mixtes dans la promotion. On appelle couple mixte, un couple entre des étudiants de deux parcours (cursus) différents. Par exemple, si Alice est en MIAGE et Bob en Info, leur couple est un couple mixte. Par contre deux étudiant·e·s de DLMI ne forment pas un couple mixte.

10. Écrire une fonction `int rechercher_etudiant(promo p, etudiant *e)` qui donne l'indice du tableau auquel se trouve l'étudiant « `e` » dans la structure « `p` » de type `promo`. On renverra -1 si l'étudiants n'est pas présent.

11. On souhaite exclure les étudiants au comportement problématique. On dit que l'étudiant Bob est infidèle à Alice si Alice est en couple avec Bob, mais que Bob est en couple avec un·e autre étudiant·e qu'Alice.

Écrire une fonction `int desinscrire_partenaire_infidele(promo *p, etudiant *alice)` qui enlève de la promo la personne en couple avec Alice, si cette personne est infidèle. On reverra les codes d'erreurs suivants :

- `-1` si Alice n'a pas de partenaire
- `-2` si le partenaire d'Alice est fidèle
- `-3` si autre problème
- `1` si le partenaire infidèle a été désinscrit avec succès !

Pour avoir tous les points, il faut réussir à enlever l'étudiant du tableau sans faire de boucle (on peut par contre changer l'ordre des étudiants dans le tableau et utiliser la fonction `rechercher_etudiant`).

Conclusion : l'heure de la libération !

Après discussion avec le comité d'éthique de la Faculté des Sciences, il semblerait que de telles statistiques posent des problèmes légaux (RGPD) et éthique et qu'en tant qu'enseignants, les romances des étudiants ne rentreraient pas dans nos champs de compétences.

12. Écrire une fonction `void liberer(promo p)` qui libère toute la mémoire associée à la promo afin de ne laisser aucune trace des données allouées sur le tas.