

Sujet d'examen 1
Pratique du C

Novembre 2011

Introduction

Écrivez lisiblement et n'hésitez pas à commenter votre code en langage C. Vous ne pouvez utiliser que les fonctions C dont le prototype est donné dans l'énoncé et celles dont vous donnez la définition dans vos copies.

Les sections sont indépendantes ; lisez l'énoncé complet avant de commencer à le résoudre.

1 Quizz

1. À quoi ressemblera le programme suivant après traitement par le préprocesseur :

```
/* #include<stdio.h> */
#define N 100
int main(void){
#ifdef N
    printf("N vaut %d\n", N);
#else
    printf ("N n'est pas défini\n");
#endif
return 0 ;
}
```

Pour répondre, donnez le code C résultant de ce traitement.

2. Donnez en justifiant votre réponse l'affichage produit par le code suivant :

```
#include<stdio.h>
union compteenbanque {
    double solde ;
    int numero ; } ;
struct etudiant {
    char **nom;
    union compteenbanque potdevin;
    enum e {male, femelle, indetermine} genre;
    unsigned short int numiden;
    struct etudiant *acopiersur; };
typedef struct etudiant *PDC[100];

int main(void){
    printf("%d\n", sizeof(struct etudiant));
    printf("%d\n", sizeof(PDC));
    return 0 ;
}
```

Vous pouvez utiliser `sizeof(type)` avec `type` un type scalaire au lieu des chiffres renvoyés par cette expression. L'exercice consiste à expliciter la constitution en mémoire d'un objet construit sur le modèle `struct etudiant`.

2 Validité d'un code ISBN

La plupart des livres sont publiés avec un code les identifiant : il s'agit du code ISBN pour International Standard Book Number. Ce code est composé d'entiers compris entre 0 et 9. On utilise la lettre *X* pour représenter l'entier 10. De plus, des tirets sont introduit dans le code afin d'en faciliter la lecture sans pour autant avoir d'autre signification.

Seul les 9 premiers chiffres d'un code ISBN sont utilisés pour identifier le livre. Le 10ième caractère sert à contrôler la validité du code (comme la clef d'un RIB ou les deux derniers chiffres de votre numéro de sécurité sociale).

L'algorithme pour tester la validité du code ISBN est simple. On calcule à partir de ce dernier deux sommes s_1 et s_2 . Le code ISBN est correct si la valeur finale de s_2 est divisible par 11.

On expose l'algorithme au travers de l'exemple du code ISBN 0-13-162959-X. Considérons tout d'abord le calcul de s_1 .

chiffres du code ISBN	0	1	3	1	6	2	9	5	9	10(X)
s_1	0	1	4	5	11	13	22	27	36	46

Le calcul de s_2 est fait en sommant les sommes partielles de s_1

chiffres du code ISBN	0	1	3	1	6	2	9	5	9	10(X)
s_1	0	1	4	5	11	13	22	27	36	46
s_2	0	1	5	10	21	34	56	83	119	165

Pour finir, on constate que 165 est le produit de 15 par 11. Notre code ISBN est donc valide.

Question. Construisez une fonction C d'identificateur `IsISBNValid` qui prend en argument un tableau de caractères représentant le code ISBN (pouvant contenir des tirets) et retourne 0 si, et seulement si, ce code est correct.

3 Un programme récursif

Écrire une fonction **récursive** qui récupère les caractères de l'entrée standard et les reproduit sur la sortie standard en ordre inverse. Par exemple, si on a en entrée la suite de caractères 1234A alors la fonction retourne sur la sortie standard la suite de caractères A4321.

Remarque : vous disposez des fonctions `getchar` et `putchar`.

4 Conversion récursive d'une chaîne de caractères en entier

Donnez la définition de la fonction récursive de prototype `int convertirString(char ch[MAX])` qui convertit la chaîne de caractères `ch` en un entier `nb` renvoyé par la fonction. Remarquez que l'entier `nb` peut être négatif.

5 Recherche par fonction de hachage

Pour commencer, fixons arbitrairement deux proportions :

```
/* on choisit un nombre premier */
#define n 131
/* on choisit un multiple */
#define m 4
```

On se propose de coder un carnet d'adresse par une table (d'au plus $m \times n$ entrées) et pour ce faire on désire construire un tableau dont chaque élément (aussi appelé cellule) donne accès à trois champs :

- un nom (d'au plus 30 caractères) que l'on considère être la clef de la table ;
- un numéro de téléphone (codé par un entier non signé) ;
- un entier initialisé à -2 et dont l'usage sera explicité par la suite.

Fonction de hachage Le hachage est une méthode de recherche dans une table. On utilise une fonction h de l'ensemble des clefs dans un intervalle d'indice. Pour une clef x , l'entier $h(x)$ est l'indice de la cellule qui contient x dans la table.

Pour construire h , on choisit un nombre premier n (diviseur de la taille de la table) et un entier positif b (puissance de 2 pour des raisons arithmétiques) ; dans cet exercice $n = 131$ et $b = 128$. Ces entiers permettent de construire une fonction h qui à partir d'une chaîne x de longueur ℓ retourne un entier :

$$h : x \rightarrow (x[0] + x[1] \cdot b + \dots + x[\ell - 1] \cdot b^{\ell-1}) \bmod n. \quad (1)$$

(mod désigne le reste de la division euclidienne e.g. $10 = 3 \bmod 7$.)

Recherche dans la table Pour toute clef x , la fonction h donne une entrée possible dans les n premières cellules la table. Deux cas se présentent alors :

- soit la clef contenue dans la cellule d'indice $h(x)$ est bien x ;
- soit ce n'est pas le cas. Cette situation est une *collision* : il existe une autre clef x' telle que $h(x') = h(x)$. Dans ce cas, on utilise le troisième champs pour indiquer l'indice i_1 de la cellule candidate à contenir la clef x . Si on ne trouve pas la valeur cherchée à cet indice, alors une nouvelle collision est rencontrée et on peut utiliser le troisième champs de la cellule d'indice i_1 pour pointer sur une autre cellule, etc.

La figure suivante donne un exemple de cette structure de donnée :

indice	nom	tél	collision
0	Sandrine	2345	-1
1			-2
2	Marianne	3556	-1
3			-2
4	Béatrice	4234	131
5			-2
	...		
131	Anne	4333	430
	...		
430	Laëtitia	3205	-1

La valeur -1 indique qu'il n'y a pas de collision et la valeur -2 que la cellule est vide.

Questions.

1. Donnez la déclaration d'un type permettant de représenter une cellule et la définition du tableau global codant la table.

-
2. Donnez le code d'une fonction de hachage h qui prend en entrée une chaîne de caractères et qui retourne un entier donné par la formule (1).
 3. Donnez le code d'une fonction de recherche qui prend en argument une chaîne de caractères et qui retourne un numéro de téléphone.
 4. Donnez le code d'une fonction d'insertion qui prend en argument une chaîne de caractères et un numéro de téléphone et qui insère ces informations dans la table. On peut supposer disposer d'une variable globale :

```
static int emptycell = n ;
```

indiquant la première cellule vide en dehors des indices dans l'image de la fonction de hachage.