



EXAMEN DE RATRAPAGE

Matière : Atelier Programmation avancée **Date:** juin 2015
Groupes : LA1 RI **Documents et calculatrices:** Non autorisés
Durée : 1h30 **Nbre Pages** : 2
Enseignants : M^{me} A. Ben Ali, M^r H. Eletaif

NB : Les réponses illisibles seront lourdement pénalisées

Exercice 1

8 points

Une opération est définie par ses deux opérandes ainsi qu'un opérateur (addition (+), soustraction (-), multiplication (*), division entière (/) et le reste de la division entière (%)).

- Définir la structure **Operation** décrite par les champs suivants : (1,5 points)
 - **Operande1, Operande2** : entiers
 - **Operateur** : caractère
 - **Resultat** : un entier qui représente le résultat de l'opération.
- Ecrire la fonction **Saisir_Operation(Operation *op)** qui saisit les deux opérandes et l'opérateur de l'opération *op*. (1,5 points)
- Ecrire la fonction **Afficher_Operation(Operation op)** qui affiche l'opération *op* sous la forme *<opérande1><opérateur><opérande2> = <Résultat>*. (1 point)
Par exemple 5 + 3 = 8.
- Ecrire la fonction **Calculer_Operation(Operation *op)** qui effectue l'opération *op* et met à jour le champ *Resultat*. (2 points)
- Ecrire la fonction **Calculer(int N)** qui saisit, calcule et affiche *N* opérations allouées dynamiquement dans un tableau. (2 points)

Exercice 2

12 points

Un numéro de compte bancaire se décompose en quatre parties :

- Code banque : cinq caractères
- Code guichet : cinq caractères
- Numéro de client : onze caractères
- Clef RIB : deux caractères

Pour vérifier si un numéro de compte soit valide, on réalise sa conversion en un nombre de 23 chiffres et si ce dernier est divisible par 97, alors le numéro de compte est valide. La conversion d'un numéro de compte se fait de la manière suivante :

Caractères	Chiffre
0	0
1, A, J	1
2, B, K, S	2
3, C, L, T	3
4, D, M, U	4
5, E, N, V	5
6, F, O, W	6
7, G, P, X	7
8, H, Q, Y	8
9, I, R, Z	9

1. Définir la structure de données *Numero_Compte* qui représente les données d'un numéro de compte. (1,5 points)
2. Ecrire la fonction *Saisir_Compte(Numero_Compte *nc)* qui assure la saisie du numéro de compte bancaire *nc* passé en paramètre. (1,5 point)
3. Ecrire la fonction *Convertir_Caractere(char c)* qui convertit le caractère *c* passé en paramètre selon le tableau ci-dessus et retourne le chiffre correspondant. Ce chiffre sera de type caractère. (2 points)
4. Ecrire la fonction *Convertir_Compte (Numero_Compte nc, char cc[])* qui convertit les données du numéro de compte bancaire *nc* passé en paramètre en une chaîne de caractères qui contient le nombre de 23 chiffres. Utilisez la chaîne *cc* passée en paramètre pour stocker la chaîne résultat. (2,5 points)
5. Ecrire la fonction *Valider_Compte(Numero_Compte *nc)* qui saisit le numéro de compte bancaire *nc*, contrôle sa validité et affiche le message « Numéro de compte valide » ou « Numéro de compte non valide » selon le cas. Utilisez la fonction *atoi(...)* pour convertir une chaîne de caractères en un entier long. (2,5 points)
6. Ecrire la fonction *main ()* dans laquelle vous déclarez un numéro de compte *N* et vous le validez. (2 points)