

LIFAP1 – Partie B – Langage C/C++

Contrôle Continu Terminal (Durée : 2h)

Lundi 18 décembre 2017

Recommandations : Les documents, calculatrice, téléphone portable sont interdits. La qualité de l'écriture et de la présentation sera prise en compte dans la note finale. Vous veillerez à **respecter** les notations et les règles d'écriture des algorithmes vues en cours et en TD.

Compactage de chaînes de caractères...

On souhaite écrire un programme permettant de compresser les chaînes de caractères. Pour ce faire, on comptera le nombre de répétitions de chaque caractère et on fera précéder le caractère par ce nombre d'occurrences : par exemple "aaaa" sera transformé en "4a".

Note : on supposera qu'aucun des caractères de la chaîne de départ n'apparaît plus de 9 fois successives et on pourra utiliser tous les sous-programmes de la bibliothèque string.h.

Ecrire en C/C++ une fonction `compresse_chaine` qui, à partir de la chaîne passée en paramètre, "renvoie" au programme principal la chaîne résultat construite (sans l'afficher dans le sous-programme) et retourne le taux de compression obtenu : (longueur de la chaîne finale / longueur de la chaîne initiale) * 100. Exemple : avec `ch_initiale = "nnnnnoooooooooonnnnnnnnn"` on construira `ch_finale = "5n9o8n"` et le taux de compression sera de $(6/22)*100 = 27,27\%$.

NOM :

.....

PRENOM :

.....

Numéro Etudiant :

.....

On souhaite créer une application de gestion d'un club de natation.

Un nageur est identifié par son nom, son prénom ainsi qu'un tableau de performances pour les 4 distances suivantes : 50m, 100m, 200m et 400m pour chacun des mois de l'année. On utilisera un tableau 2D de réels (secondes) pour représenter les performances.

Laure												
Wanadoo												
27												
55												
112												
237												

Un club est identifié par le nombre de nageurs et un tableau de nageurs.

- 1- Définir en C/C++ deux constantes `CHMAX` (qui sera utilisée pour toutes les chaînes de caractères) et `MAXPLOUF` (le nombre maximum de nageurs dans le club) ayant pour valeurs respectives 64 et 32.

- 2- Définir en C/C++ les structures `nageur` (contenant les champs `nom`, `prenom`, et `tab_performance`), et `club` (contenant les champs `nb_nageurs` et `tab_nageurs`).

- 3- Ecrire en C/C++ la **fonction** `remplir_nageur` qui permet de saisir toutes les informations relatives à un `nageur`. Attention, on veillera à recommencer la saisie des temps, tant que la valeur saisie n'est pas strictement positive.

- 4- Ecrire en C/C++ la procédure `ajoute_nageur` qui permet, **s'il reste de la place** dans le tableau de nageurs, d'ajouter un `nageur`.

- 5- Ecrire en C/C++ un sous-programme `min_max_moy` qui à partir d'un tableau de 12 réels "retourne" la valeur minimale, la valeur maximale **et** la valeur moyenne de ces 12 valeurs.

- 6- En utilisant le sous-programme écrit à la question précédente, écrire le sous-programme `meilleures_perf` qui "retourne" un tableau de 4 réels contenant les meilleures performances d'un nageur (pour chacune des distances : 50m, 100m, 200m, et 400m) sur une année.

- 7- Ecrire en C/C++ un sous-programme `affiche_nageur` qui affiche les meilleures performances annuelles (pour chaque distance) d'un nageur dont le nom est passé en paramètres.

- 8- Ecrire en C/C++ le programme principal permettant d'ajouter autant de nageurs à un club que l'utilisateur le souhaitera (on lui demandera après chaque ajout s'il veut ajouter ou non ajouter un nouveau nageur), et affichera (pour chaque distance) les meilleures performances pour un nageur dont le nom sera choisi par l'utilisateur.