

LIFAPI – Partie B – Langage C/C++

Contrôle Continu Terminal (Durée totale : 2h)

Lundi 9 janvier 2023

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

Nous allons développer une application permettant la gestion des **réservations de chambres** dans un **hôtel**.

Une `Chambre` est identifiée par son numéro `Num_Chambre` (un entier), le nombre de lits `Simple`s et de lits `Double`s (deux entiers) et un tableau `T_Prix` de 3 prix (entiers) correspondants aux tarifs appliqués en basse saison, moyenne saison et haute saison.

Une `Reservation` contient un numéro de chambre `Num_Chambre` (entier), le nombre de nuits `Nb_nuits` réservées (entier) et le `Tarif` utilisé dans les calculs (entier entre 0 et 2 correspondant à l'indice dans le tableau de prix de la `Chambre`).

Enfin un `Hotel` contiendra un tableau `T_Chambre` de `MAXCHAMBRE` `Chambre`, le nombre de `Chambre` `Nb_Chambre`, un tableau `T_Resa` de `MAXNUITS` `Reservation` et le nombre de `Reservation` `Nb_Resa`.

1. Définir en C/C++ deux constantes `MAXCHAMBRES` et `MAXNUITS` ayant pour valeurs respectives 32 et 128.

NOM :
PRENOM :
Numéro Etudiant :

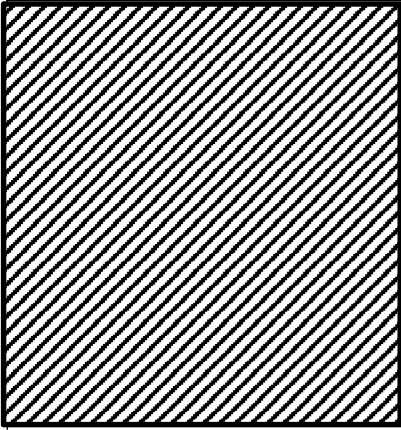
2. Définir en C/C++ les 3 structures précédentes.

```
struct Chambre {
```

```
struct Reservation {
```

```
struct Hotel {
```

3. Ecrire en C/C++ une **fonction** `Remplir_Chambre` qui crée et retourne une `Chambre`. Le numéro de la `Chambre` sera passé en paramètre. Les valeurs de tous les autres champs seront demandées à l'utilisateur mais on recommencera la saisie tant que :
 - a. les prix ne sont pas strictement positifs et de plus en plus élevés entre deux cases successives du tableau,
 - b. et le nombre total de lits (`Simple`s + `Double`s) dans une `Chambre` n'est pas strictement positif et inférieur ou égal à 5.



4. Ecrire en C/C++ une **procédure** `Remplir_Reservation` qui crée et "retourne" une `Reservation` en demandant à l'utilisateur le nombre de nuits `Nb_Nuits`. On recommencera la saisie tant que la valeur n'est pas strictement positive. Le `Tarif` et le numéro de la `Chambre` seront passés en paramètres.

5. Ecrire en C/C++ un sous-programme `Remplir_Hotel` qui demande à l'utilisateur combien de `Chambre` il y a dans l'`Hotel` (entier compris entre 1 et `MAXCHAMBRE`) et ajoute autant de `Chambre` que souhaité dans l'`Hotel`. On utilisera `Remplir_Chambre` avec comme numéro de `Chambre` (`1 + indice de la Chambre`) dans le tableau `T_Chambre`. Les saisies seront recommencées tant que les contraintes ne sont pas respectées. On veillera à initialiser le nombre de `Reservation` à 0.

6. Ecrire en C/C++ un sous-programme `Ajoute_Une_Reservation` qui ajoute une réservation à l'`Hotel` s'il reste de la place dans le tableau de `Reservation`. On utilisera `Remplir_Reservation`. Le `Tarif` sera une valeur aléatoire entre 0 et 2, et le numéro de la chambre sera tiré aléatoirement parmi les numéros de `Chambre` existants. Une même `Chambre` pourra être réservée plusieurs fois.

7. Ecrire en C/C++ un sous-programme `Chiffre_Affaire` qui calcule et "retourne" le gain total de l'Hotel (`Gain_Tot`) ainsi que le nombre de client hébergés (`Nb_Clients`). Le **gain** sera obtenu en multipliant le nombre de nuits pour chaque réservation par le `Tarif` de la `Chambre` réservée. Le **nombre de clients** sera quant à lui évalué en sommant le nombre de lits de chaque `Chambre` réservée. Si le client séjourne plusieurs nuits (dans une même `Reservation`) il ne sera comptabilisé qu'une seule fois.

8. Ecrire le programme principal qui
- Remplit un `Hotel`,
 - Ajoute autant de `Reservation` que l'utilisateur le voudra
 - Calcule et affiche le gain de l'Hotel ainsi que le nombre de clients qu'il a reçus.