LIFAP1 – TD 12 : Révisions

Objectifs : Réviser toutes les notions abordées dans l'UE.

Terenui est un véritable Globetrotter … Il parcourt le monde à la recherche des paysages les plus idylliques.
Aussi afin de conserver une trace de ses aventures, nous avons décidé de conserver une trace de ses différents périples.

Pour chaque voyage, on renseignera les informations suivantes :

- Destination
- Continent (Asie, Afrique, Europe, Amérique, Océanie)
- Durée du séjour (en nombre de jours)
- Appréciation (note sur 20)

Sachant qu'il effectue **au plus** 20 voyages par an, nous allons définir une **structure** contenant le nombre de voyages déjà effectués ainsi qu'un tableau de voyages (caractéristiques de chacun des voyages effectués).

1. Définir en C/C++ les structures de données nécessaires à la mise en œuvre du programme : structure voyage et structure tous_les_voyages.
2. Écrire en C/C++ un sous-programme remplir_voyage permettant de saisir **toutes** les informations relatives à un voyage.
3. Écrire en C/C++ un sous-programme ajoute_voyage permettant d'ajouter les caractéristiques d'un voyage au tableau de voyages. **On utilisera pour cela le sous-programme écrit en 2.**
4. Écrire en C/C++ un sous-programme moyenne_par_continent qui calcule et "renvoie" au programme principal la moyenne des appréciations des voyages effectués **pour chacun** des 5 continents. On utilisera un tableau tab_moyenne pour stocker ces 5 valeurs (indice 0 : Afrique, indice 1 : Amérique, indice 2 : Asie, indice 3 : Europe, indice 4 : Océanie).
   On supposera que chaque continent aura été visité au moins une fois.
5. Écrire en C/C++ un sous-programme le_mieux_et_le_pire permettant (à partir du tableau des moyennes défini en 4) de renvoyer l'indice du continent ayant la meilleure moyenne des appréciations, **et** l'indice du continent ayant la plus mauvaise moyenne.
6. Écrire en C/C++ le programme principal permettant d'ajouter dans le tableau de **voyages autant de caractéristiques de voyages que l'utilisateur le voudra** et d'afficher le **nom** du continent préféré de Terenui et celui qui l'enthousiasme le moins.

```
/* soit version LIF1 on définit des constantes,
   soit + fainéant, on fait un enum */

enum Continent {
 Afrique = 0,
 Amerique = 1,
 Asie = 2,
 Europe = 3,
 Oceanie = 4,
 Antartique = 5 /* même si on n'y va pas souvent */
} ;
```

```
static
char *libelle[] = {
  "Afrique", "Asie", "Amerique", "Europe", "Oceanie", "Antartique"
    } ;

static
const int NB_CONTINENTS = 6 ;

/* pour LIF1, ils peuvent faire de même avec des constantes, et en
   réécrivant libelle en beau tableau 2D */

/* question 1 */

/* les constantes */

/* 2 */

static
Continent saisie_continent()
{
  int i ;
  int lu ;
  cout << "Choix du continent ?" << endl ;
  for (i = 0 ; i < NB_CONTINENTS; i++ )
  {
    cout << i << ")" << libelle [i] <<endl;
  }
  do {
    cout << "Votre choix (entre 0 et "<< NB_CONTINENTS - 1 << ") ?" << endl;
    cin >> lu ;
  } while ( (lu < 0) || (lu >  NB_CONTINENTS - 1)) ;
  return (Continent) lu ;
}

void saisie_voyage ( voyage &v )
{
  float lu ;
  cout << "Destination ?" << endl ;
  cin >> v.destination ;
  cout << "Quel continent :" ;
  v.continent = saisie_continent() ;
  cout << "Duree ? (nombre de jour)" << endl ;
  cin >> v.duree_jour ;
  do {
    cout << "Votre avis (notes entre 0 et 20) ?" << endl ;
    cin >> lu ;
  } while ((lu <0) || (lu >20)) ;
  v.note_20 = lu ;
}

/* 3 */

void init_tous_les_voyages (tous_les_voyages &tv)
{
  tv.nb_voyages = 0 ;
}

void ajoute_voyage  (tous_les_voyages &tv)
{
  if ( tv .nb_voyages >= MAX_VOYAGES - 1 ) {
```

```
    cout << "Plus possible d'ajouter un voyage-quota de l'annee epuise" <<endl;
    return ;
  }
  saisie_voyage ( tv.liste [ tv.nb_voyages ]) ;
  tv.nb_voyages = tv.nb_voyages + 1;
}

/* 4 */

void moyenne_par_continent (tous_les_voyages tv,
            float tab_moyenne[NB_CONTINENTS]) {
  int i ;
  int c ;
  float tab_cumul[NB_CONTINENTS] ;
  int   nb_voyages[NB_CONTINENTS] ;
  /* init */
  for ( c = 0; c < NB_CONTINENTS; c++ ) {
    tab_cumul[c] = 0.0;
    nb_voyages[c] = 0 ;
  }
  /* boucle */
  for (i = 0 ; i < tv.nb_voyages; i++ ) {
    c = tv.liste[i].continent ;
    tab_cumul[c] = tab_cumul[c] + tv.liste[i].note_20 ;
    nb_voyages[c] = nb_voyages[c] + 1 ;
  }
  /* resultat */
  for ( c = 0; c < NB_CONTINENTS; c++ ) {
    if (  nb_voyages[c] == 0 ) {
      tab_moyenne[c] = -1 ; /* je ne tiens pas compte de la remarque*/
    } else {
      tab_moyenne[c] = tab_cumul[c] /  nb_voyages[c] ;
    }
  }
}

/* 5 */

/* input non pecisee dans sujet, j'ai fait au plus simple */
void le_mieux_et_le_pire ( float tab_moyenne[NB_CONTINENTS],
            int &imeilleur, int &ipire )
{
  int c ;
  imeilleur = 0;
  ipire = 0 ;
  for (c = 1; c < NB_CONTINENTS; c++ ) {
    if ( tab_moyenne[c] >  tab_moyenne[imeilleur] ) {
      imeilleur = c;
    }
    if  ( ( (tab_moyenne[c] <  tab_moyenne[ipire] ) &&
          (tab_moyenne[c] >= 0)) || (tab_moyenne[ipire] < 0))
    { /* comme la remarque a ete mangee - empeche optimisation un cas bug*/
      ipire = c ;
    }
  }
}

/* 6 */

int main()
```

```
{
  tous_les_voyages recueil ;
  float tab_moyenne[NB_CONTINENTS] ;
  int bien, pas_bien ;
  char lu ;
  /* init */
  init_tous_les_voyages (recueil) ;
  do {
    ajoute_voyage ( recueil ) ;
    cout << "voulez-vous continuer ? (o ou n) ?" << endl ;
    cin >> lu ;
  } while ((lu == 'o') && (recueil.nb_voyages < MAX_VOYAGES ) ) ;
  moyenne_par_continent ( recueil, tab_moyenne ) ;
  le_mieux_et_le_pire ( tab_moyenne, bien, pas_bien ) ;
  cout << "Le continent prefere de Terenui est :" << libelle[bien] << endl ;
  cout << "et le moins apprecie est :" << libelle[pas_bien] << endl ;
  cout << "That's all folk" << endl ;
  return 0 ;
}
```