

IFB2 – Examen final

Durée : 1h00

Seule une feuille A4 manuscrite nominative recto-verso est autorisée comme document.

Tous les autres documents ou tout autre dispositif électronique sont interdits.

Barème donné à titre indicatif (±1)

Exercice 1 – Structures et tris (8 points)

Dans un jeu vidéo, un personnage possède un inventaire d'objets, chaque objet est caractérisé par son nom, son poids, son niveau de rareté (1 = commun, 2 = rare, 3 = légendaire) et son prix de revente.

Un exemple de description d'objet est :

- Hache à deux mains
- Rare
- 3,5 kg
- 1500 pièces d'or

Ecrire le code C répondant aux énoncés suivants :

1. Définir un type structure représentant un objet, choisissez judicieusement les types de données appropriés
2. Ecrire une fonction prenant en paramètre un tableau d'objets de taille fixe (représentant l'inventaire) et triant ce tableau par poids décroissant ; le choix de l'algorithme de tri est libre

Exercice 2 – Chaînes de caractères et fichiers (12 points)

Un fichier CSV est un fichier contenant des lignes de valeurs séparées par des virgules. Dans cet exercice nous travaillerons sur un tel fichier « inventaire.txt », dont le contenu se présente de la façon suivante :

```
Hache à deux mains,2,3.5,1500  
Epée simple,2,4.5,1000  
Bouclier large,1,8.2,500
```

Ecrire le code C répondant aux énoncés suivants :

1. Ecrire une fonction qui étant donnée une chaîne de caractère contenant une ligne du fichier (par exemple « Hache à deux mains,2,3.5,1500 »), retourne un entier représentant le prix de l'objet (le prix étant le nombre se trouvant après la 3^{ème} virgule)
2. Ecrire le code qui calcule la somme du prix des objets contenus dans le fichier « inventaire.txt ». Vous appellerez la fonction écrite à la question précédente.

Aide-mémoire (prototypes de fonctions usuelles, tels que définis dans la référence du langage C) :

```
int atoi( const char * theString );  
FILE * fopen( const char * filename, const char * accessMode );  
int fclose( FILE * stream );  
int fprintf( FILE * stream, const char *format, ... );  
int fscanf( FILE * stream, const char *format, ... );  
char * strcpy( char * destination, const char * source );  
unsigned int strlen( const char * theString );  
char * fgets( char * string, int maxLength, FILE * stream );  
char * gets( char * string );
```