

Fonctions

TRAITER UNE SERIE DE PIECES

Les mesures d'une série de 500 pièces sont stockées dans un tableau nommé `cote[]` sous la forme d'un réel, de type `double`. Une pièce est déclarée valide si la valeur de sa cote est comprise entre la cote minimum et la cote maximum. Au tableau `cote[]` on associe un tableau `pieceValide[]`. Ce tableau contient l'index des pièces dont la cote est valide.

1.1 *Éléments du logiciel*

On définit les valeurs suivantes :

```
const double COTE_MAX = 155.5 ;
const double COTE_MIN = 152 ;
```

On dispose de 3 fonctions :

```
void afficherPiece (double tab[], int taille) ;
// Affiche la cote des Pièces.

int nbrePieceValide(int *ptrValide, int taille) ;
// Compte le nombre de pièce valides

void pieceValide(double *ptrTab, int *ptrValide, int taille)
// Renseigne le tableau pieceValide.
```

Le tableau `pieceValide[]` est initialisé à la valeur 2000.

1.2 *Questions*

- En langage C++, établissez le fragment de code pour :
 - Déclarer des 2 tableaux
 - Initialiser le tableau `pieceValide` à la valeur 2000.
- En utilisant les 3 méthodes à votre disposition, proposez le codage, en langage C++, de la séquence suivante :

```
// Afficher la cote des pièces
// Identifier les pièces dont la cote est valide
// Compter le nombre de pièces valides
```
- Etablir le corps des fonctions :
 - `affichePiece()`
 - `nbrePieceValide()`
 - `pieceValide()`
- Proposez la déclaration et le corps de la fonction `nbrePieceGrande()`. La fonction retourne le nombre de pièces dont la cote est trop grande. Le passage du tableau `cote[]` est réalisé par référence.
- Proposez la déclaration et le corps de la fonction `afficheCoteValide()`. La fonction affiche les cotes des pièces valides. Le passage des tableaux `cote[]` et `pieceValide[]` sont réalisés par référence.
- Proposez la déclaration d'une structure de données qui regroupe les informations : cote des pièces, index des pièces valides, nombre de pièces valides.
- Proposez la déclaration de 3 fonctions : `affichePiece()`, `nbrePieceValide()`, `pieceValide()` qui permettent d'opérer à partir de la structure de données.