

Correction de l'évaluation C++ 1 du 27/11/23

1 QCM [6 pts]

Voir sur Moodle la correction dédiée.

2 Questions de cours [6 pts]

- Q_1 En python, on exécute directement le code à l'aide d'un environnement de développement (*spyder*, *jupyter*, *ipython*, ...), expliquer en quoi C++ est différent sur cet aspect-là.

Réponse: En C++, on compile le code écrit dans un ou plusieurs fichiers source, à l'aide d'un compilateur (g++ sous Linux par ex), afin de créer un fichier exécutable. Cette étape n'existe pas en python, qui est un langage interprété.

- Q_2 Citer 3 structures de contrôle de C++, les décrire brièvement. Donner 1 exemple d'utilisation

Réponse: Structures de contrôle pouvant être citées (il fallait en citer 3):

- *for*: boucle avec compteur
- *while*: permet de répéter une ou plusieurs instructions tant qu'une condition est vraie
- *do while*: assez proche de *while*, mais l'instruction est exécutée au moins une fois
- *if*: pour choisir si une partie du code sera exécutée ou pas
- *switch*: pour choisir quelle partie de code sera exécutée selon la valeur d'une variable

Exemple d'utilisation de la boucle *for*:

```
for (int i=0; i<10; i++)  
{  
    cout << "i = " << i << endl;  
}
```

- Q_3 Comment crée-t-on une variable en C++? Comparer à python, donner deux types d'entiers, deux types de réels et un autre type courant de C++

Réponse: En C++, une variable doit avoir un type, au contraire de python. On peut déclarer une variable en lui donnant un type et lui attribuer une valeur en même temps.

- Types entiers: *int*, *unsigned int*, *long int* (ou *long*), *short int* (ou *short*), ...
- Types réels: *float*, *double*, *long double*
- Autre type: *char*, *string*, *bool*, un type pointeur était accepté aussi

- *Q₄* Expliquer la surcharge de fonction en C++. Donner un exemple.

Réponse: En C++, sous un même nom, une fonction peut avoir des versions différentes. Elle peut avoir des paramètres d'entrée de type différent d'une version à une autre et le nombre de paramètres d'entrée peut changer aussi; enfin, le type de sortie peut changer. C'est le compilateur qui fait la différence et qui choisit la bonne version de la fonction à utiliser. Ce mécanisme n'existe pas en python.

Exemple:

```
void printMe(int a)
{
    cout << "Hello ! I'm an integer ! : " << a << endl;
}

void printMe(double a)
{
    cout << "Hello ! I'm a double ! : " << a << endl;
}
```

- *Q₅* Qu'est-ce un pointeur? Citer une utilisation concrète d'un pointeur. Indiquer une commande, utilisée de nombreuses fois dans le cours, liée à la manipulation des pointeurs et expliquer son intérêt.

Réponse: Un pointeur est une adresse mémoire. Il peut servir à définir de manière dynamique un tableau d'éléments d'un même type ou travailler sur une variable en mémoire.

Les commandes courantes associées à un pointeur sont:

new, *delete*, *&* ou ***

- *Q₆* Décrire, de manière assez détaillée, par quoi est composée une classe en C++. Indiquer les étapes nécessaires pour utiliser des objets d'une classe en C++. Indiquer la différence entre classe et structure.

Réponse: Une classe est composée de variables, de fonctions membres (par exemple des getters (ou accesseurs) ou des setters (mutateurs)), de constructeurs et d'un destructeur. Les variables et fonctions membres peuvent avoir le statut privé (par défaut) ou le statut public (en utilisant le mot-clé *public*).

On utilise une classe en définissant un objet de la classe (dans la fonction *main*, par ex.), en faisant appel à un des constructeurs de la classe. La différence entre *class* et *struct* concerne le statut par défaut des attributs: il est privé pour la classe, public pour la structure.

3 Code à compléter/commenter [4 pts]

3.1 Code à compléter

Voir sur Moodle les fichiers *Animal.hh* et *Animal.cpp*

3.2 Questions sur le code

- Q_1 Pourquoi a-t-on défini les fonctions membres *get_poids* et *get_nom* dans la classe *Animal*?

Réponse: *_poids* et *_nom* sont des attributs privés de la classe *Animal*. Pour avoir accès à leur valeur en dehors de la classe, dans les fonctions *getMaxPoids*, *poidsTotal* et dans le *main*, on doit donc définir les fonctions membres publiques *get_poids* et *get_nom* dans *Animal* (qui sont des getters (accesseurs)).

- Q_2 Dans la fonction *main*, qu'est-ce que la variable *troupeau*? Combien a-t-elle d'éléments et de quel type?

Réponse: La variable *troupeau* est un tableau contenant 3 objets *Animal*.

- Q_3 Quel est le résultat attendu pour *PoidsTotal* (1.70 et 71 dans la fonction *main*)?

Réponse: *PoidsTotal* doit valoir 1460 (soit la somme des poids des 3 animaux)

- Q_4 Quel est l'affichage attendu dans le *main* pour les lignes 69 et 73-74?

Réponse: On doit obtenir

```
Animal vache, de 1000 kgs
Animal mouton, de 400 kgs
Animal chevre, de 60 kgs
L'animal de poids le plus eleve est vache
```

4 Codes à écrire [4 pts]

Voir sur Moodle les fichiers *theSum.cpp* et *myMax.cpp*