**CPP OBJET**

**1. Question de cours**

1.1 Quel est le rôle d’un constructeur, exécuté quand ?

Le rôle d’un constructeur est de « construire » un objet : Déclarer un objet en définissant ses variables par des valeurs par défaut, ou, si le constructeur accepte des arguments, alors les variables de l’objet seront définies en fonction de ses arguments. Il n’est pas obligatoire d’avoir autant d’argument que de variable. En effet, certaines peuvent être « set » par défaut dans le constructeur, tandis que d’autre, nous laissons le choix.

1.2 Quel est le rôle d’un destructeur, exécuté quand, indispensable quand ?

Le rôle d’un destructeur est de détruire l’objet et ses variables, afin de libérer l’espace mémoire qui lui avait été alloué. L’objet ne sera donc plus utilisable. Il est indispensable à la fin d’un programme, au moins, pour éviter les fuites de mémoire.

1.3 Comment repère-t-on un constructeur d’une classe ?Peut-il y en avoir plusieurs ? Pourquoi ?

On repère un constructeur d’une classe en voyant le nom de la classe dans le programme. C’est une méthode qui porte exactement le même nom que la classe, tandis que le destructeur aura le signe « ~ » devant le nom de la classe.

*Example :*

*Classe Utilisateur*

*Constructeur :*

*Utilisateur() ;*

*Destructeur :*

*~Utilisateur() ;*

En effet, il peut y avoir plusieurs constructeur mais qu’un seul destructeur. On peut déclarer plusieurs constructeur pour, comme dit précédemment question 1.1, une classe peut avoir un constructeur par défaut, ou un constructeur prenant en compte des paramètres. Il est donc possible d’avoir plusieurs constructeur.

*Example :*

*Class Utilisateur*

*Constructeur par défaut :*

*Utilisateur() ;*

*Constructeur avec paramètre(s) :*

*Utilisateur(String nom) ;*

*Constructeur avec 2 paramètres :*

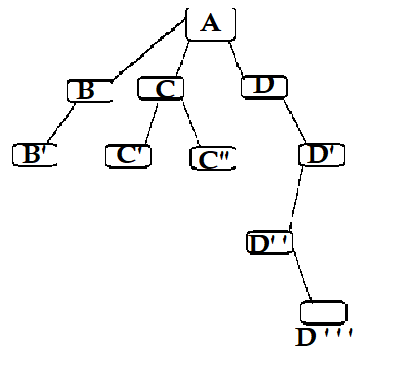
*Utilisateur(String nom, int age) ;*

*Unique Destructeur :*

*~Utilisateur() ;*

1.4 Qu’est-ce que la surdéfinition de fonctions ?

La surdéfinition de fonction prend son importance quand on parle d’héritage. C’est-à-dire qu’une classe peut être enfant d’une autre classe, et une classe parent peut avoir plusieurs enfants. De plus, un enfant peut aussi être parent d’une autre classe, etc…



1.5 Quel est la signification de public et de private dans une classe ?

Public :

Dans une classe, le mot clé « public » signifie que la méthode, variable, [..], peut être accessible depuis « partout ».

Private :

Dans une classe, le mot clé « private » signifie que la méthode, variable, [..], ne peut être accédé(e) qu’à partir de la classe même.

*(Protected) :*

*Dans une classe, le mot clé « protected » signifie que la méthode, variable, [..], ne peut être accédé(e) depuis la classe même, ainsi que ses enfants.*