



Corrigé-Examen ST 15 : Programmation Orientée Objet C++ Durée : 1h30 Année
2017-2018

Question de cours (08 points)

1. Donnez la définition d'un constructeur et d'un destructeur ?

Le constructeur est une méthode ayant le même nom que la classe et qui n'a pas de type de retour. Il peut prendre des paramètres. **01**

Le destructeur est une méthode appelée lorsque l'objet est détruit. Le destructeur doit être impérativement écrit pour libérer la mémoire allouée de façon dynamique pendant la vie d'un objet. Le destructeur est désigné par le nom de la classe précédé de ~, il n'a pas de paramètre, ni de type de retour. **01**

Que signifient les mots : surcharge, polymorphisme, héritage ?

Surcharge : définition de plusieurs fonctions avec des paramètres différents. **01**

L'héritage entre classes symbolise les relations hiérarchiques entre les concepts. Si une classe B hérite d'une classe A, on dit A est la classe *mère* ou appelé aussi classe de *base* et B la classe *filie*, on dit aussi que B *dérive* de A ou encore que B est *dérivée* de A. La relation d'héritage est une relation de généralisation-spécialisation : B est une *spécialisation* de A et A est une *généralisation* de B. Si la classe B hérite de la classe A, elle hérite des attributs et des méthodes de la classe A. Ainsi **un objet de la classe B est aussi un objet de la classe A** (mais l'inverse n'est sûrement pas vrai !!). **01**

Le polymorphisme est une fonctionnalité de l'héritage : la capacité d'appeler une méthode en fonction du type réel d'un objet (sa classe). **01**

2. Quelle est le rôle de l'instruction : **try ... catch ()** dans un programme ?

La gestion des exceptions. **0.5**

3. Quel est la différence entre les trois mots clés de droit d'accès : **public**, **private** et **protected** ? **1.5**

Le mot clé **public** et **private** indiquent la visibilité des membres. Les membres privés ne sont accessibles que par les objets de la classe. Par convention, les attributs sont toujours à accès privé.

Private : restreint l'accès aux seuls corps des méthodes de cette classe.

protected : comme private sauf que l'accès est aussi autorisé aux corps des méthodes des classes qui héritent (directement ou indirectement) de cette classe.

4. Quelle est le rôle de la bibliothèque **iostream** dans un programme C++ ?

Bibliothèque des entrée/Sortie **Cin** et **Cout**. **0.5**

5. Si une classe D dérive de la classe B, que présente la classe D par apport à la classe B ? **classe fille 0.5**

Exercice 1 (06 points)

Ecrire un programme qui permet de calculer le factorielle d'un nombre entier n, en utilisant les deux boucles :

1. For (...) **03**

```
#include<iostream>
using namespace std;
int factorielle (int n) { // déclaration et définition de la
fonction factorielle
    int i,fact=1;
    for(i=n;i>=1;i--)
    fact=fact*i;
    cout<<fact;
}
main() {
    int n,k;
    cout<<"Entrez la valeur de n :=";
    cin>>n;
    k=factorielle(n); // appel de la fonction
factorielle
}
```

1. Do ...while () **03**

```
#include<iostream>
using namespace std;
int factorielle (int n) { // déclaration et définition de la
fonction factorielle
    int i,fact=1;
    i=n;
    do
    fact=fact*i;
    i--;
    while (i>=1);
    cout<<fact;
}
main() {
    int n,k;
    cout<<"Entrez la valeur de n :=";
    cin>>n;
    k=factorielle(n); // appel de la fonction
factorielle
}
```

Exercice 2 : (06 points)

```
#include<iostream>
using namespace std;
class calcule { // (03 points)
public:
float volume (float r, float h) {
float v;

v=3.14*r*r*h;
cout<<h;

}
};
class calcule1: public calcule { // (03 points)
public :

float surface (float a, float b) {
float s;

s=3.14*a*b;
cout<<s;

}
};
int main() {
    calcule k;
    calcule1 n;
    int n; float r,h,a,b ;

    cout<<"Entrez la valeur de r et de h :=";
    cin>>r; cin>>h; cin>>a; cin>>b;
    k.volume(r,h) ;
    n.surface(a,b) ;
}
```