

C++ - M1 - Session 2 - Juin 2011

Exercice 1

Soit le programme suivant :

```
#include <iostream>
using namespace std;
class Premiere {
public:
    Premiere()          { cout << "ctor premiere" << endl; }
    virtual ~Premiere() { cout << "dtor premiere" << endl; }
    void salut()        { cout << "salut" << endl; }
    virtual void bonjour() { cout << "bonjour" << endl; }
};
class Seconde : public Premiere {
public:
    Seconde()          { cout << "ctor seconde" << endl; }
    virtual ~Seconde() { cout << "dtor seconde" << endl; }
    void salut()        { cout << "salutations" << endl; }
    virtual void bonjour() { cout << "bien le bonjour" << endl; }
};
int main() {
    Premiere p;
    Seconde *s = new Seconde;
    p.salut(); p.bonjour();
    (*s).salut(); (*s).bonjour();

    Seconde *ps = s;
    ps->salut(); ps->bonjour();
    Premiere *pp = ps;
    pp->salut(); pp->bonjour();
    delete s;
    return 0;
}
```

Indiquez, en argumentant, ce qu'il afficherait à son exécution.

Exercice 2

Soit le programme suivant :

```
#include <iostream>
using namespace std;

class Objet {
public:
    Objet() { cout << "ctor Objet" << endl; }
    ~Objet() { cout << "dtor Objet" << endl; }
};

Objet *f1(Objet o) {
    Objet *po = new Objet(o);
    return po;
}
```

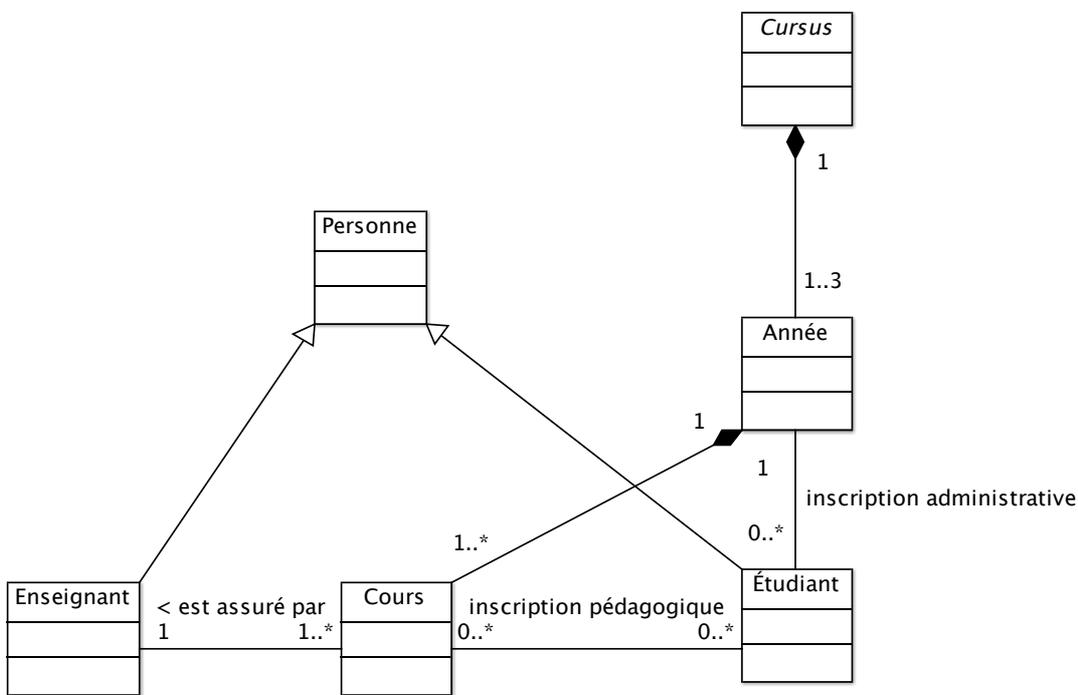
```
Objet f2(Objet &o) {
    Objet oo;
    return o;
};
```

```
int main() {
    Objet o;
    Objet *po = f1(o);
    delete po;
    o = f2(o);
}
```

Indiquez ce qu'il affiche et expliquez le comportement en apparence surprenant de son exécution.

Exercice 3

Soit le diagramme UML suivant :



Écrivez le code C++ correspondant (classes, relations et méthodes/constructeurs/destructeurs annexes si nécessaire)

Exercice 4

Soit les deux classes suivantes:

```
class Texte {
private:
    string laChaine;
public:
    Texte(string s) : laChaine(s) {}
};
```

```
class ObjetGraphique {
private:
    int color;
public:
    ObjetGraphique(int c) { color = c; }
};
```

Comment définir une classe `TexteGraphique` qui permettrait d'afficher une chaîne de caractère en utilisant une police (Fonte) particulière et une couleur donnée, *i.e.* qui pourrait être utilisée de la façon suivante:

```
int main() {
    TexteGraphique t("Bonjour tout le monde", ROUGE, "Helvetica",
12);
}
```

En supposant que les deux classes `ObjetGraphique` et `Texte` dérivent toutes deux d'une même classe (disons `Objet`), que cela changerait-il ?

Exercice 5

Soit le code suivant :

```
template <class X>
class MaClasse {
private:
    X attribut;
public:
    MaClasse(const X &v) : attribut(v) { attribut.setValue(12); }
    X uneMethode(X *x) { return *x+attribut; }
};
```

À quelles conditions une classe `A` peut-elle être utilisée pour instancier le template ? Écrire son code.

Exercice 6

Quelle différence faites-vous entre surcharge et redéfinition ? Donner un exemple de chaque.

À quoi sert la spécialisation partielle ? En donner un exemple pertinent (qui ne serait pas dans le cours...)