

Examen session 2

Lundi 24 Juin 2013

Tous documents autorisés (sauf copie du voisin, appareillage électronique, etc). Les téléphones portables, comme tout autre moyen de communication vers l'extérieur, doivent être éteints. Le temps à disposition est de trois heures. Motivez bien vos réponses.

1 Exercice

prem2.cpp

```
1 #include <iostream>
2 using namespace std;
3 class P {
4 public:
5     P() { cout << "ctor_P" << endl; }
6     P(int i) { cout << "ctor_P(" << i << ")" << endl; }
7     virtual ~P() { cout << "dctor_P" << endl; }
8     virtual string coucou() { return "coucou"; }
9 };
10 class S : public P {
11 public:
12     S() { P p(5); cout << "ctor_S" << endl; }
13     virtual ~S() { cout << "dctor_S" << endl; }
14     virtual string coucou() { return "cuicui"; }
15 };
16 void f(P p) { cout << "allo_ici_f()" << endl; }
17 void f(P *p) { cout << p->coucou() << "_ici_g()" << endl; }
18 int main() {
19     P p;
20     S *s = new S;
21     S *ps = s;
22     P *pp = ps;
23     f(p);
24     f(s);
25     f(pp);
26     f(ps);
27     delete s;
28     return 0;
29 }
```

1. Indiquer ce que ce programme affiche lors de son exécution, en justifiant chaque ligne d'affichage.

2. À quoi sert le mot-clé `virtual` devant la définition des destructeurs ? Donner un exemple permettant d'illustrer sa nécessité.

2 Exercice

On souhaite pouvoir effectuer des calculs arithmétiques en précision arbitraire en autorisant la syntaxe habituelle de l'arithmétique. Autrement dit, on souhaite pouvoir exécuter le code suivant :

Soit l'extrait de programme suivant :

```
BigNum fonction(BigNum a, const BigNum b) {  
    BigNum r = a, q = 0;  
    while (r >= b) {  
        r = r - b;  
        q++;  
    }  
    return q;  
}
```

1. Écrire la classe `BigNum` et toutes les méthodes nécessaires au bon fonctionnement du code précédent. Il n'est pas utile d'implémenter de *vrais* entiers en précision arbitraire, on peut se contenter d'utiliser des `int` en première approximation.
2. Si l'on utilise de vrais `BigNums`, quel problème le code précédent pose-t-il ? Et comment corriger ?

3 Exercice

1. Écrire un type `Array<T, N>` permettant de représenter un type de tableau de N éléments de type T . Ce type devra contenir au minimum la surcharge de l'opérateur d'accès `[]` ainsi qu'un contrôle de la validité de l'indice (en cas d'erreur une exception devra être levée contenant les informations utiles).
2. Puisqu'il s'agit d'un modèle de classe, préciser les contraintes imposées sur le type pour que l'instanciation du modèle fonctionne.

4 Problème

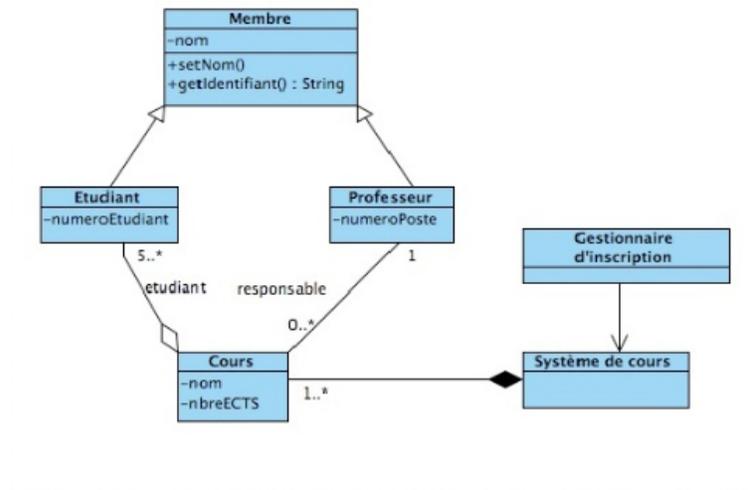


Diagramme emprunté à Mme Blay-Fornarino (Université de Nice).

1. Quels sont les types des différentes relations de ce diagramme ?
2. Traduire en C++ la spécification précédente.