

Ejercicio 2- CC3002

Prof: Nancy Hitschfeld Kahler

9 Noviembre 2011

1. Considere la definición de la siguiente jerarquía de clases:

```
class A{
    ...
public:
    A(){ cout <<"A\n";}
    A(int x){
        cout <<"A:"<<x<<"\n";
    }
    virtual ~A(){
        cout <<"~A\n";
    }
    virtual int m()=0;

    virtual A* g(A* a){
        return new B(a->m());
    }
    virtual Ra* f(Pa*){
        ...
    }
};

class B: public A {
    ...
public:
    B(){ cout <<"B\n";}
    B(int x){
        cout <<"B:"<<x<<"\n";
    }
    ~B(){
        cout <<"~B\n";
    }
    virtual int m(){
        return -1;
    }
    virtual B* g(A* a){
        return new C(a->m());
    }
    virtual Rb* f(Pb*){
        ...
    }
};

class C: public B{
    ...
public:
    C(){ cout <<"C\n";}
    C(int x){
        cout <<"C:"<<x<<"\n";
    }
    ~C(){
        cout <<"~C\n";
    }
    int m(){
        return 1;
    }
    virtual C* g(A* a){
        return new C(a->m());
    }
    virtual Rc* f(Pc*){
        ...
    }
};
```

(a) Para el Código B, diga qué se chequea en tiempo de compilación y qué imprime cada instrucción en tiempo de ejecución. Si hay errores diga por qué se producen. Donde se usa enlace dinámico? Nota: Hágalo para las instrucciones marcadas con (*).

(b) Suponiendo que c++ implementa la redefinición de métodos bajo la ley de la contravarianza diga cuál es la relación más general entre los tipos Pa, Pb, y Pc, y entre Ra, Rb y Rc, (quién puede ser subtipo de quién) para que el código C compile y ejecute como se esperaría.

Código B

```
A* a = new C();
A* aa = a->g(new C()); (*)
B* bb = a->g(new B()); (*)
B* b = new B();
aa = b->g(new C()); (*)
bb = b->g(new A()); (*)
```

Código C

```
Pa* pa; Pb* pb;

A* a = new C();
Rb* rb = a->f(pb);
B* b = new C();
Ra* a = b->f(pa);
```