

## CS 3723: Programming Languages

Apr 23, 2012

You Name:

- For each of the following situations, indicate whether it uses subtype polymorphism or implementation inheritance, or both.
  - A Java class implementing an interface.
  - A Java class extends another class.
  - A C++ class extends another class using public inheritance.
  - A C++ class extends another class using private inheritance.
  - A C++ class extends another class using protected inheritance.
- Which of the following C++ class hierarchy is a correct subtype translation of the ML datatype  $edges = EmptyEdge \mid EdgeList$  of  $string * string * edges$ ?
  - ```
class edges { public: virtual void print() = 0; };
class EmptyEdge : public edges { public: virtual void print() {} };
class EdgeList : public edges
{
    private: const char *from, *to;    edges* cdr;
    public: virtual void print() { printf("%s->%s",from,to); cdr->print(); }
        EdgeList(const char* f, const char* t, edges* c) { from=f; to = t; cdr = c; }
};
```
  - ```
class EmptyEdge { public: virtual void print() {} };
class EdgeList : public EmptyEdge
{
    private: const char *from, *to;    edges* cdr;
    public: virtual void print() { printf("%s->%s",from,to); cdr.print(); }
        EdgeList(const char* f, const char* t, edges* c) { from=f; to = t; cdr = c; }
};
```
  - ```
class edges { public: virtual void print() {} };
class EmptyEdge : public edges {};
class EdgeList : public edges
{
    private: const char *from, *to;    edges cdr;
    public: virtual void print() { printf("%s->%s",from,to); cdr.print(); }
        EdgeList(const char* f, const char* t, edges c) { from=f; to = t; cdr = c; }
};
```