

Inheritance: Polymorphism III

Another approach, same output:

animal* A[2]; A[0] = &c; A[1] = &d; for (int j = 0; j < 2; j++) A[j]->print();

Note, how this is different:

```
animal a = d; // default copy constructor called!
a.print();
Output is:
```

My name is Rex.

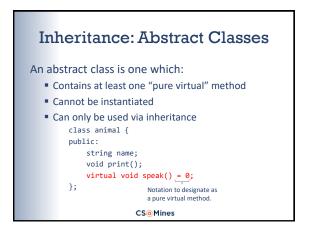
CS@Mines

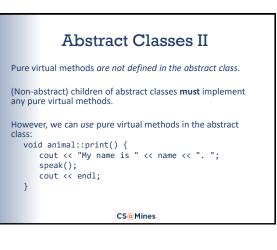
Polymorphism

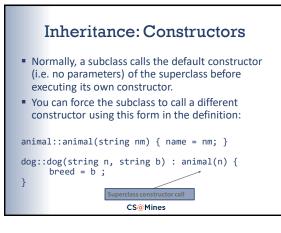
- The word polymorphism means having many forms. Typically, polymorphism occurs when there is a hierarchy of classes and they are related by inheritance.
- C++ polymorphism means that a call to a member function will cause a different function to be executed depending on the type of object that invokes the function.

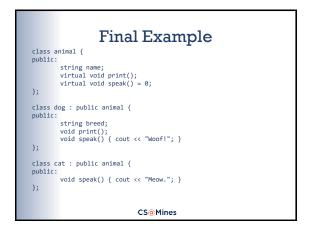
Polymorphism in C++ - tutorialspoint.com. (n.d.). Retrieved October 25, 2016, from https://www.tutorialspoint.com/cplusplus/cpp_polymorphism.htm

CS@Mines









Final Example II	Final Example III
<pre>void animal::print() { cout << "My name is " << name << ". "; speak(); cout << endl; } void dog::print() { animal::print(); cout << "I am a " << breed << "." << endl; }</pre>	<pre>int main() { dog d; cat c; d.name = "Rex"; d.breed = "Dachshund"; c.name = "Fluffy"; print_animal(c); print_animal(d); return 0;</pre>
<pre>void print_animal(animal& a) { a.print(); }</pre>	}
CS@Mines	CS@Mines

