CS 444/544 | Ozyegin University | Baris Aktemur

1. Given the following class definitions, what would be printed on screen?

```
class Cat extends Mammal {
                                            public void print() {
                                              System.out.print("Cat ");
class Animal {
                                              sound();
 public void print() {
    System.out.print("Animal ");
                                          }
                                          class Dog extends Mammal {
                                            public void sound() {
 public void sound() {
                                              System.out.print("Hav ");
   System.out.print("A ");
                                          }
}
                                          public class Zoo {
class Mammal extends Animal {
                                           public static void main(String[] args)
 public void print() {
   System.out.print("Mammal");
                                              Cat cat = new Cat();
    super.print();
                                              Dog dog = new Dog();
                                              Mammal ma1 = new Cat();
                                              Mammal ma2 = new Dog();
 public void sound() {
                                              Mammal ma3 = new Mammal();
    System.out.print("M ");
                                              cat.print(); System.out.println();
}
                                              dog.print(); System.out.println();
                                              mal.print(); System.out.println();
                                              ma2.print(); System.out.println();
                                              ma3.print(); System.out.println();
                                            }
```

```
C++
                                             class Cat : public Mammal {
                                             public:
                                               virtual void print() {
#include <iostream>
                                                 cout << "Cat ";
using namespace std;
                                                 sound();
class Animal {
                                             };
public:
  virtual void print() {
                                            class Dog : public Mammal {
    cout << "Animal ";</pre>
                                             public:
    sound();
                                               virtual void sound() {
                                                 cout << "Hav ";
  virtual void sound() {
                                             };
   cout << "A ";
                                            int main() {
};
                                               Cat *cat = new Cat();
                                               Dog *dog = new Dog();
class Mammal : public Animal {
                                             Mammal *ma1 = new Cat();
public:
                                               Mammal *ma2 = new Dog();
  virtual void print() {
                                               Mammal *ma3 = new Mammal();
    cout << "Mammal ";</pre>
    Animal:: print();
                                              cat->print(); cout << "\n";</pre>
                                               dog->print(); cout << "\n";</pre>
                                              mal->print(); cout << "\n";</pre>
  virtual void sound() {
                                              ma2->print(); cout << "\n";</pre>
    cout << "M ";
                                              ma3->print(); cout << "\n";</pre>
  }
                                               return 0:
};
                                             }
```

- **2.** What if we had static binding instead of dynamic binding? That is, what would the output be if the methods were all non-virtual?
- 3. What would the output be if print() were virtual but sound() were not?
- **4.** What would the output be if print() were non-virtual but sound() were virtual?