605.629: Programming Languages
Assignment 10
Sabbir Ahmed

November 10, 2021

1. [40 pts, dynamic binding]

Dynamic binding or late binding is the concept when the actual subroutine cannot be bound to the method symbol until the run-time is reached. Thus the compiler generates appropriate code in an object oriented program to perform dynamic binding when necessary.

Write a program to demonstrate dynamic binding (such as using more than one class, inheritance and printing a string in the methods of the objects) in a language of your choice (Python, Java, etc.).

Answer

Dynamic binding can be demonstrated in Python through polymorphism.

[1]:

```python
class Cat:
    def make_noise(self):
        print("Meow")

class Dog:
    def make_noise(self):
        print("Bark")

cat_obj = Cat()
dog_obj = Dog()
for obj in (cat_obj, dog_obj):
    obj.make_noise()
```

```
Meow
Bark
```

---

2. [20 pts, subtyping]

Give an example of two classes from existing libraries (Python, Java, etc.) that are subtypes but are not inherited from one another.

Answer

In the C++ standard template library (STL) the containers `std::stack` and `std::queue` are both derived from `Container()`. The deque (double ended queue)

In Python's standard library, the `collections` module contains several specialized container datatypes. `Counter` and `OrderedDict` are classes implemented in the library. `Counter` is used as a counter for hashable objects while `OrderedDict` maintains the order of insertion of the hashable objects. Both of the classes are derived from the built-in `dict` datatype, but provide different implementations.

---

3. [20 pts, inheritance]

Give one example of a disadvantage of inheritance.

Answer

One major disadvantage of inheritance is the introduction of tight coupling between the base and derived classes. Once a derived class is declared, expanding its capabilities becomes very limited due to the strong dependence on its base class. Maintainance of both of the classes becomes tedious as well, often requiring both the base and derived classes to receive concurrent and interdependent updates.

---

4. [20 pts, inheritance]

What can be the primary reason about Java having all objects derived from the Object class?

Answer

The Object class is the base class of all classes in a Java system. The class defines the basic state and behavior that all objects must have. Deriving the Object class gives every objects these basic functionalities and avoids having to redefine basic variables, methods, and classes.
