Handout 3

Inheritance (a. k. a. Generalization-Specialization Structure)

Inheritance (a.k.a. Gen-Spec structure) is organization of classes into super- and sub-classes that captures properties and behavior shared by these classes.

Examples:
1. **Superclass** (or generalization) – Animal
   **Subclasses** (or specializations) – Cat, Dog, Squirrel, Horse

2. **Superclass** – Person
   **Subclasses** – Student, Employee

Relation between subclass and superclass can be described by “is a” relation, e.g. Each Cat is an Animal, Dog is an Animal, etc…
Each Student is a Person, Each Employee is a Person

All properties (attributes and services) described in the superclass (generalization) are **inherited** by all subclasses (specializations).

Practice Problems.

1. Establish gen-spec structure of the following classes:
   ( a ) Game, Bridge, Chess, Go, BoardGame, CardGame, Poker
   ( b ) Polygon, Circle, Triangle, Rectangle, Geometric Figure.

2. Identify Classes&Objects necessary to characterize the following description of campus map. Then establish inheritance between the classes.

Bentley College campus consists of residential, administrative and academic buildings, library, parking lots, student centers and various fields. Each campus objects has a unique identification number and is depicted on the campus map. Residential buildings are undergraduate or graduate dorms, and the President’s residence. Academic buildings house classrooms, labs, and auditoriums, sports facilities, and academic departments. Administrative buildings house various offices. Parking lots are restricted for student use only, staff only and mixed (students, staff, and visitors).
Inheritance can form
- hierarchical structure, if each class has at most one superclass
- lattice, when a class inherits from more than one superclass (more commonly referred to as multiple inheritance).

**Practice problem**

What modification of the Campus Map domain would require multiple inheritance?

Guidelines from Coad&Yourdon:

Consider each class as a **generalization**. For its potential specializations ask
- Is it in the problem domain?
- Is it within system’s responsibilities?
- Will there be inheritance?
- Will the specializations meet the Class&Object criteria?

Consider each class as a **specialization**. For its potential generalization ask
- Is it in the problem domain?
- Is it within system’s responsibilities?
- Will there be inheritance?
- Will the generalization meet the Class&Object criteria?