CS2704

Topic 11
Behavioral Design

Outline

• Dimensions of Design
  – Procedural Dimensions
  – Object-Oriented Dimensions
• Scenarios
• Scenario Design Notations

Dimensions of Design

• Can view design in multiple ways
  – Structural
  – Behavioral
  – Information
• Different views for object-oriented versus procedural programming

Procedural Dimensions

• Information – (abstract) data types
• Structural – procedures and their call structure (modular structure chart)
• Behavioral – definition of tasks performed

Object-Oriented Dimensions

• Information – classes and implementing data structures
• Structural – relationships between classes (Class Diagrams)
• Behavioral – descriptions of scenarios in system

Scenarios

• “A sketch of a plot”
• Description of behavior of a method
• Want to document intended behavior for major scenarios (scenarios for constructors usually uninformative)
• Similar to pseudo-code for a method
**Design Notations**

- Show call sequence between objects
- Like diagrammed pseudo-code
- We will learn: Object message diagrams

**Diagramming Objects**

- Employee
- Employee joe
- Employee joe
- double _salary
- Date _hiredate

**Object-Message Diagram**

```
Object-Message Diagram

1. read_input
2a. Receive_message
2b.1. Locate_mailbox
2b.2. Receive_message
MailSystem
InputReader
AdminMailbox
Mailbox
```

**Parameter Objects**

- GasStation is Parameter to method

```
void Car::add_gas(GasStation& station) {
    station.buy_gas(...);
}
```

**Global Objects**

- GasStation is a global variable

```
void Car::add_gas() {
    the_gas_station.buy_gas(...);
}
```

**Field Objects**

- GasStation is field of Car object:

```
void Car::add_gas() {
    my_gas_station.buy_gas(...);
}
```
Local Objects

- GasStation is local variable of method

```
void Car::add_gas() {
    GasStation* station = new GasStation();
    station->buy_gas(...);
    delete station;
}
```

Self-References

- Method is in Car class

```
void Car::add_gas() {
    GasStation* station = find_gas_station();
    station->buy_gas(...);
}
```

Object-Message Diagram

```
process_dialing
  MailSystem 1. read_input 2. Receive_message
  InputReader 2a. Receive_message
  AdminMailbox 2b1. Locate_mailBox 2b2. Receive_message
  Mailbox
```