



# SS ZG514

## Object Oriented Analysis and Design



**BITS Pilani**

Pilani | Dubai | Goa | Hyderabad

Dr. Ritu Arora  
[rituarora@pilani.bits-pilani.ac.in](mailto:rituarora@pilani.bits-pilani.ac.in)

Content s of these slides are adapted from Applying UML and Patterns, Craig Larman, 3<sup>rd</sup> edition



# System Sequence Diagrams

# System Sequence Diagram



- Sequence Diagrams illustrate interactions of the actors with the system and the operations initiated by them.
- A system sequence diagram, shows, for one particular scenario of a use case, the events that external actors generate, their order, and inter-system events.
- The entire system is treated like a black box.
- Emphasis of the diagram is events that cross the system boundary from actors to systems.

# System Sequence Diagram

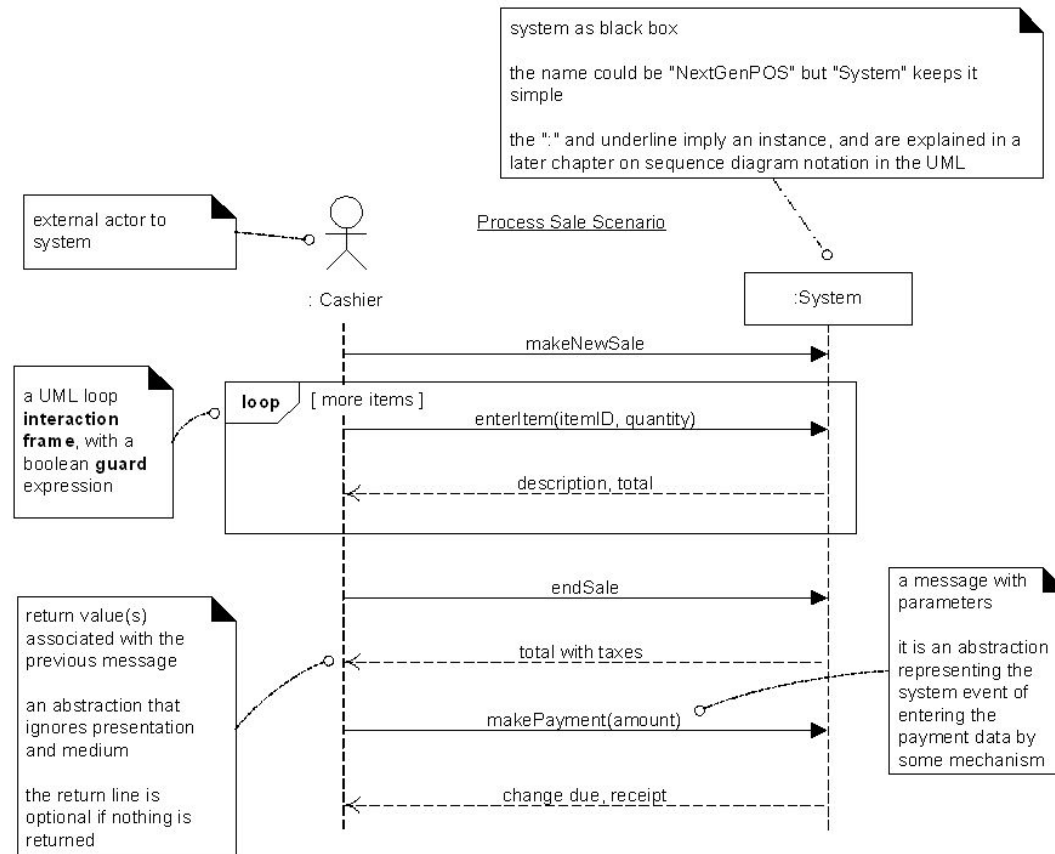


Figure representing SSD for Process Sale Scenarios for NextGen POS

Courtesy: Adapted from Applying UML and Patterns, Craig Larman, 3<sup>rd</sup> edition

# System Sequence Diagram: Notations



- :className- represents anonymous object of the class
- System is represented as a black box
- Dotted line is called the life-line of the object
- A cross on the life-line shows end of life of the object
- All requests / messages / events from actors to System are drawn as solid arrows
- These are labeled as method calls or function invocation since they invoke some functions / code in the system
- Response from the System is represented using a dotted arrow
- It would be a data, report or message send from the System to the actor

# System Sequence Diagram: Notations

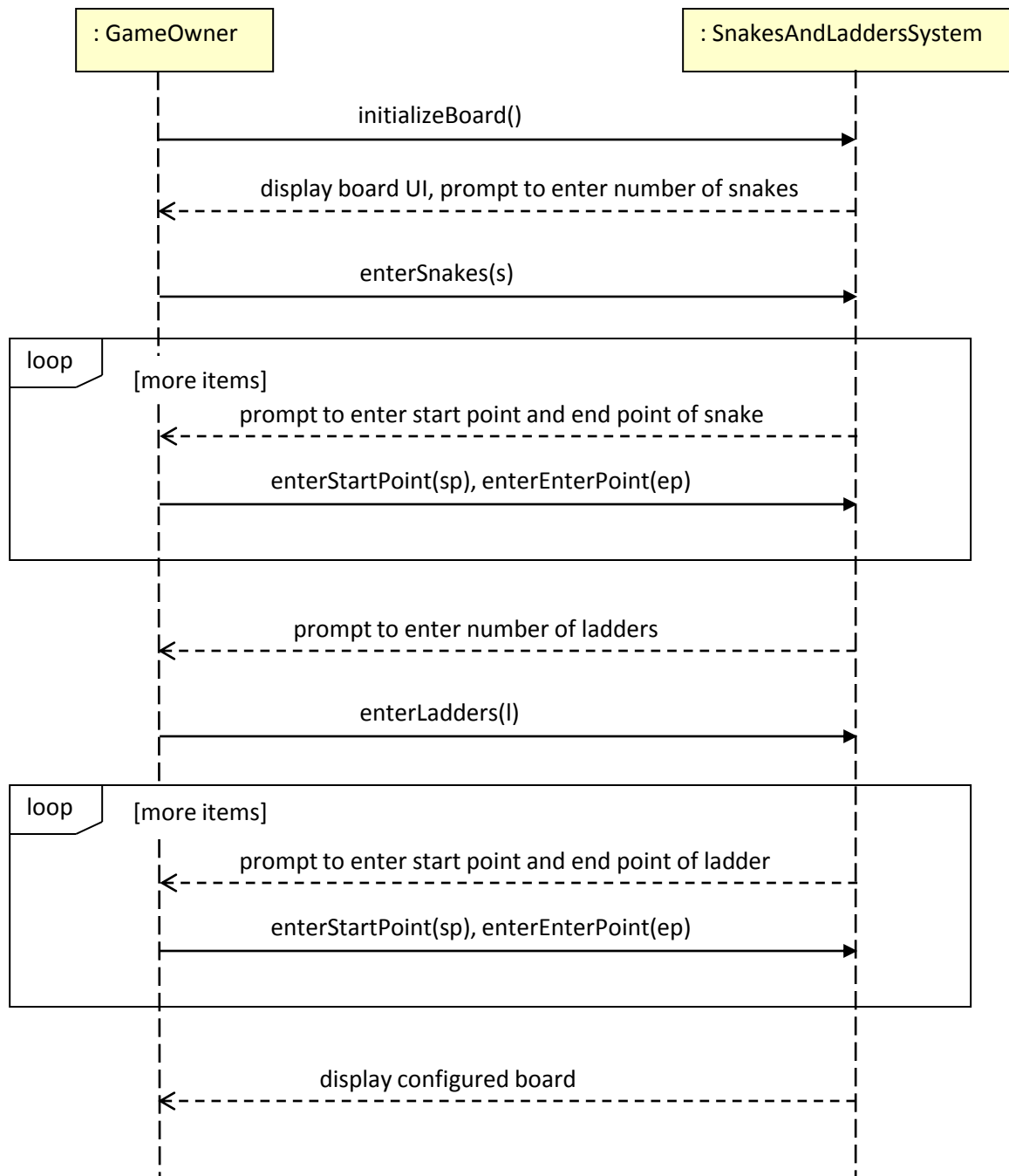


- Time proceeds downwards
- Ordering of events should follow their order in the scenario.
- Interaction frames are used to show loops in sequence diagrams.
- SSD is a pictorial representation of a use case text scenario

# Exercise: SSD



For the snakes and ladders case study, draw the system sequence diagram for the use case scenario of GameOwner configuring the game board with 's' snakes and 'l' ladders.

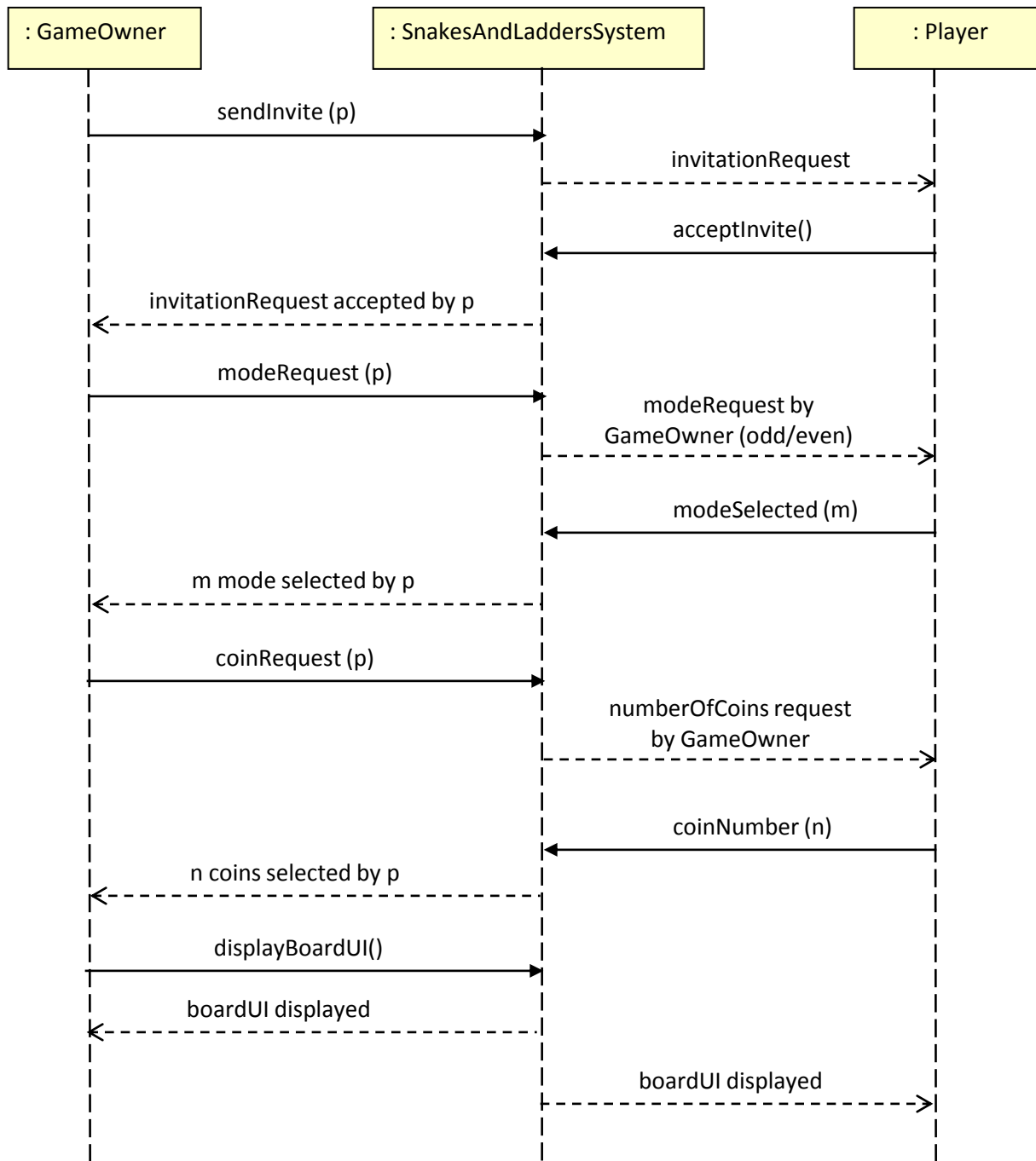




# Exercise: SSD



For the snakes and ladders case study, draw the system sequence diagram for the use case scenario of GameOwner inviting a player to join the game in specified mode, with specified number of coins.





# Activity Diagrams

# Activity Diagram

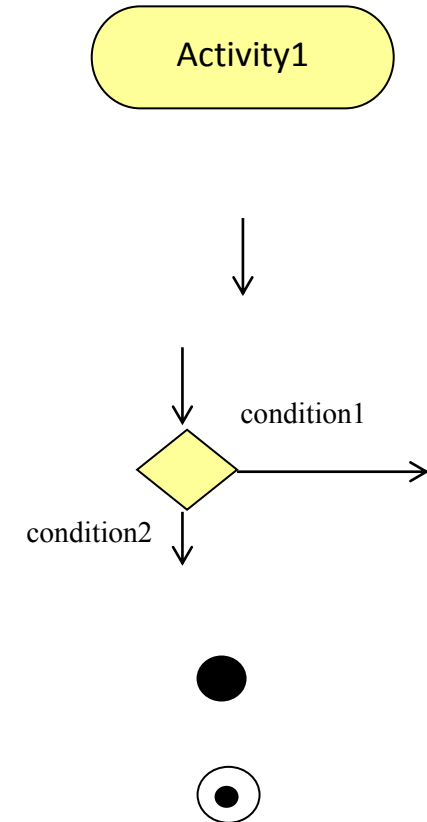


- UML Activity diagram shows sequential and parallel activities in a process
- It is useful for modeling:
  - Business processes
  - Workflows
  - Data flows
  - Use Case scenarios

# Activity Diagram: Notations



- An activity node represents execution of a statement in a procedure or the performance of a step in a workflow. It is shown as a box with rounded corners containing a description of the activity.
- Nodes are connected by control flows and data flows; depicted using arrows.
- Consists of branches, represented using decision nodes (diamond boxes), specified with guard conditions.
- Start node: represents the start of activity workflow
- Stop node: represents end of activity workflow



# Activity Diagram: Example



Consider the user interface for entering the number of snakes (as shown in figure 1 below). After entering the number of snakes, the user is prompted to enter the start and end cell number of each snake (figure 1 (b)). The starting and ending points for a snake should be between 1 and 100. Also, the starting point of the snake (mouth) should be larger than the ending point (tail) of the snake.

Draw the activity diagram for the above discussed scenario of configuring snakes on the game board.

Welcome to the game of Snakes And Ladders!

Enter the number of Snakes:

(between 1 and 10)

(a)

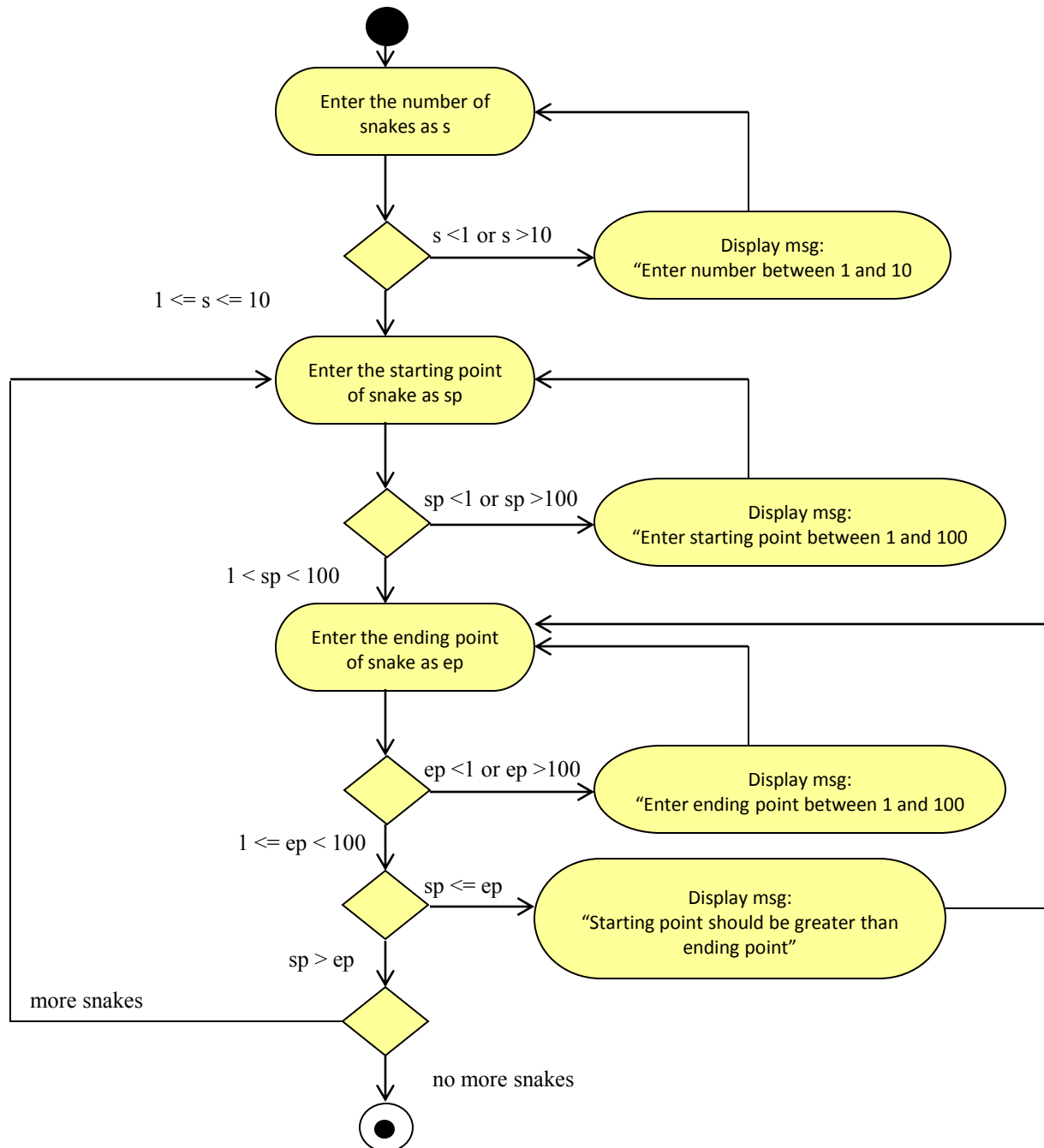
Welcome to the game of Snakes And Ladders!

Enter the starting and ending points for the Snakes:

Snake No.:	Starting Point	Ending Point
1	<input type="text"/>	<input type="text"/>
2	<input type="text"/>	<input type="text"/>
3	<input type="text"/>	<input type="text"/>
4	<input type="text"/>	<input type="text"/>
5	<input type="text"/>	<input type="text"/>
6	<input type="text"/>	<input type="text"/>
7	<input type="text"/>	<input type="text"/>

(b)

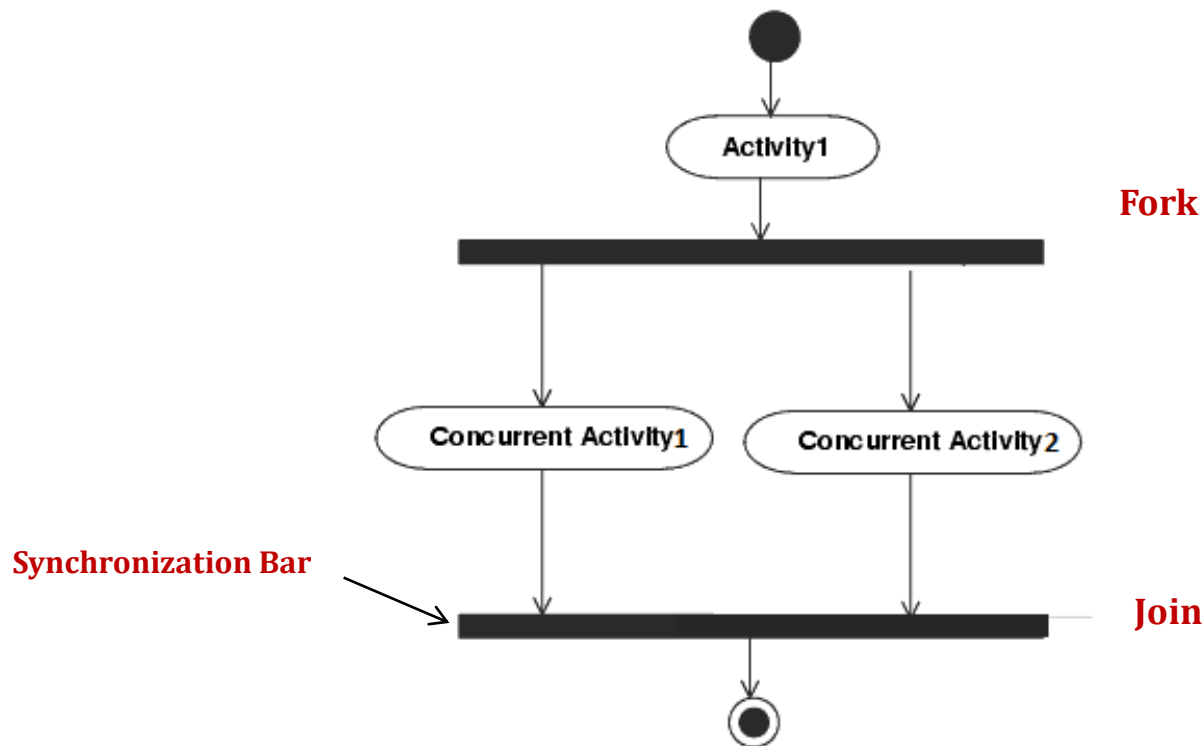
Figure 1 (a) and (b): User interface for entering the number of snakes and their end and starting points.



# Activity Diagram: Notations



- Concurrent execution is modeled using forks and joins, shown by multiple arrows entering or leaving a heavy synchronization bar.





# Example of fork and join

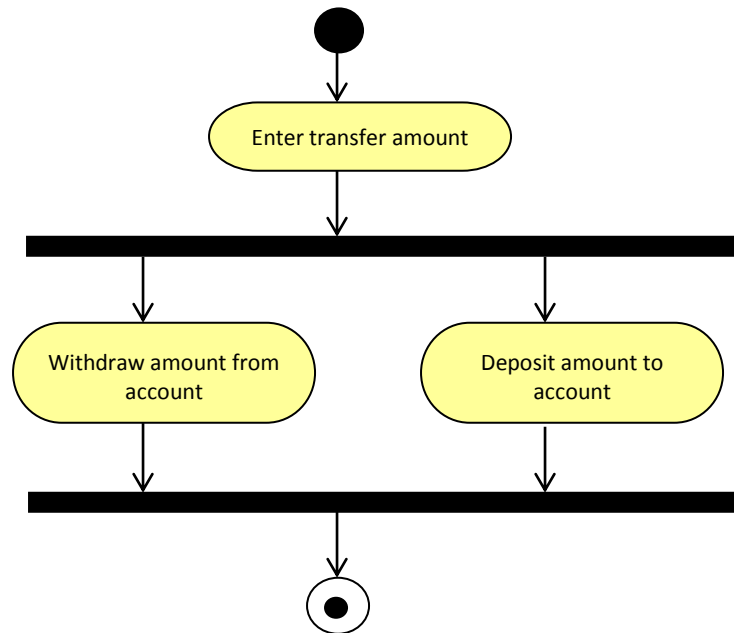
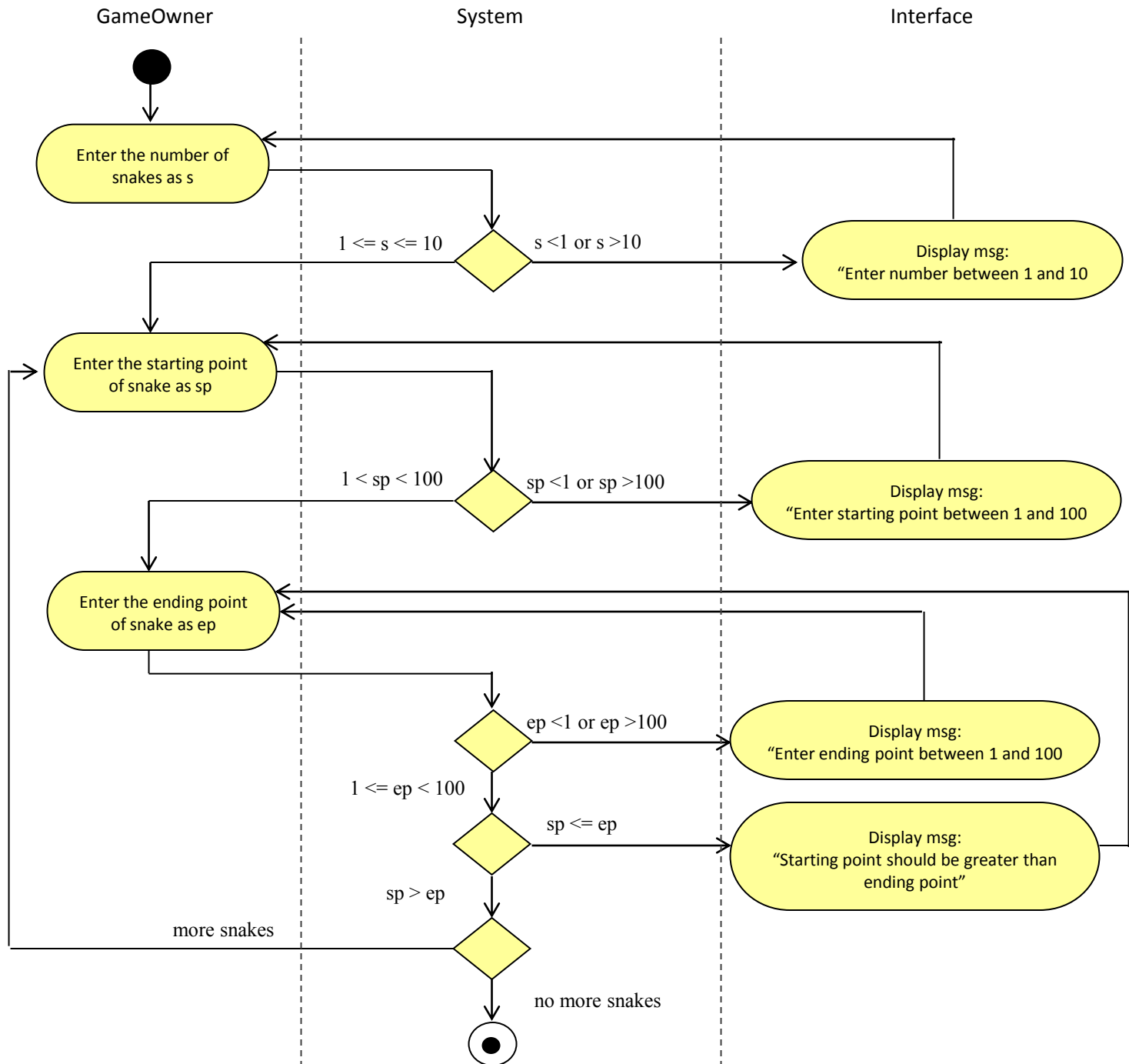


Figure gives an example parallel activities using fork and join

# Swimlane Diagram



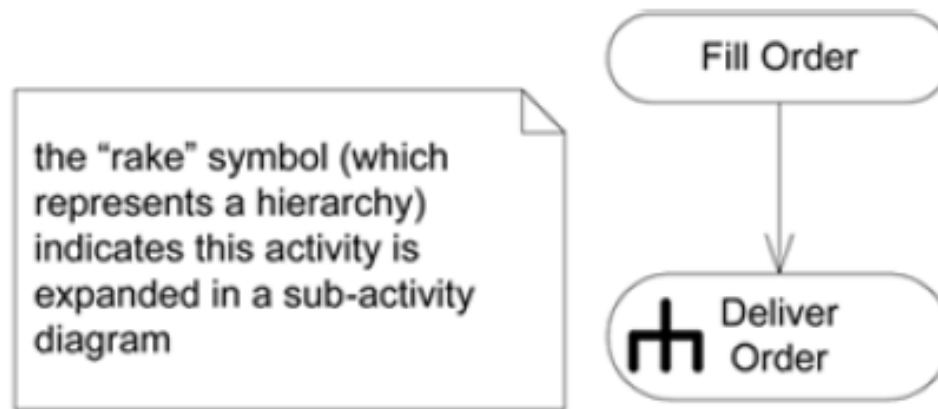
- Swimlane diagram is a useful variation of the activity diagram.
- It organizes activities according to the responsibility- indicate which actor or analysis class has the responsibility for the activity.
- Organized into distinct regions (called partitions or swimlanes) separated by lines in the diagram.



# Activity Diagram: Notations

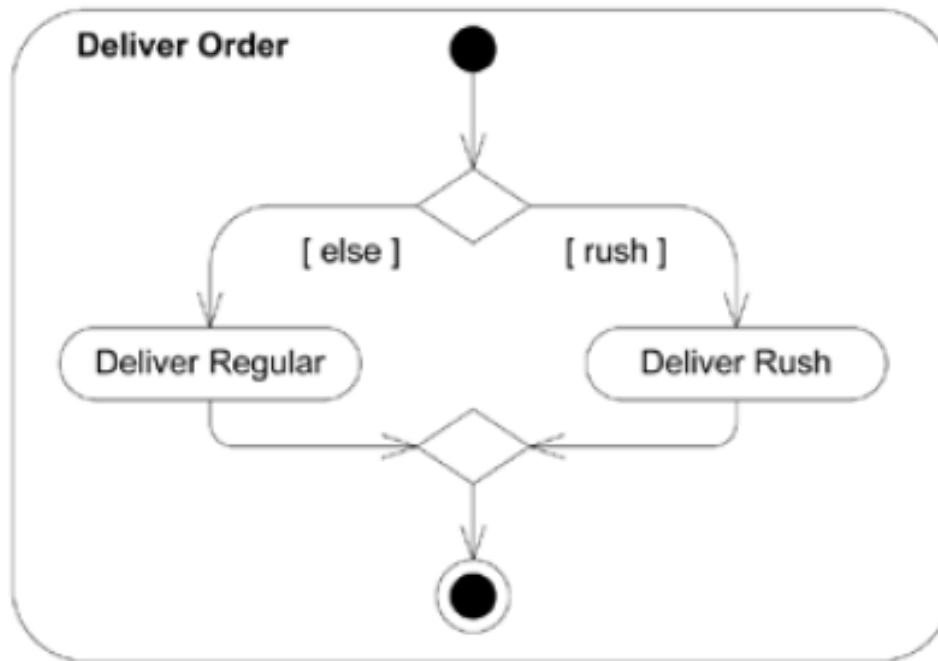


- An activity can be expanded into another activity, using the **rake** symbol.
- Use **rake** symbol to indicate a higher level activity



UML notations for rake symbol in Activity Diagram

# Activity Diagram: Notations



Expansion of an activity in Activity Diagram

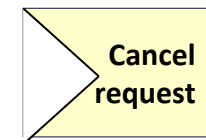
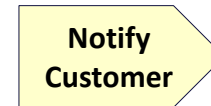
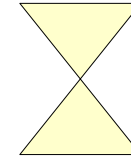
Courtesy: Adapted from Applying UML and Patterns, Craig Larman, 3<sup>rd</sup> edition

# Activity Diagram: Notations

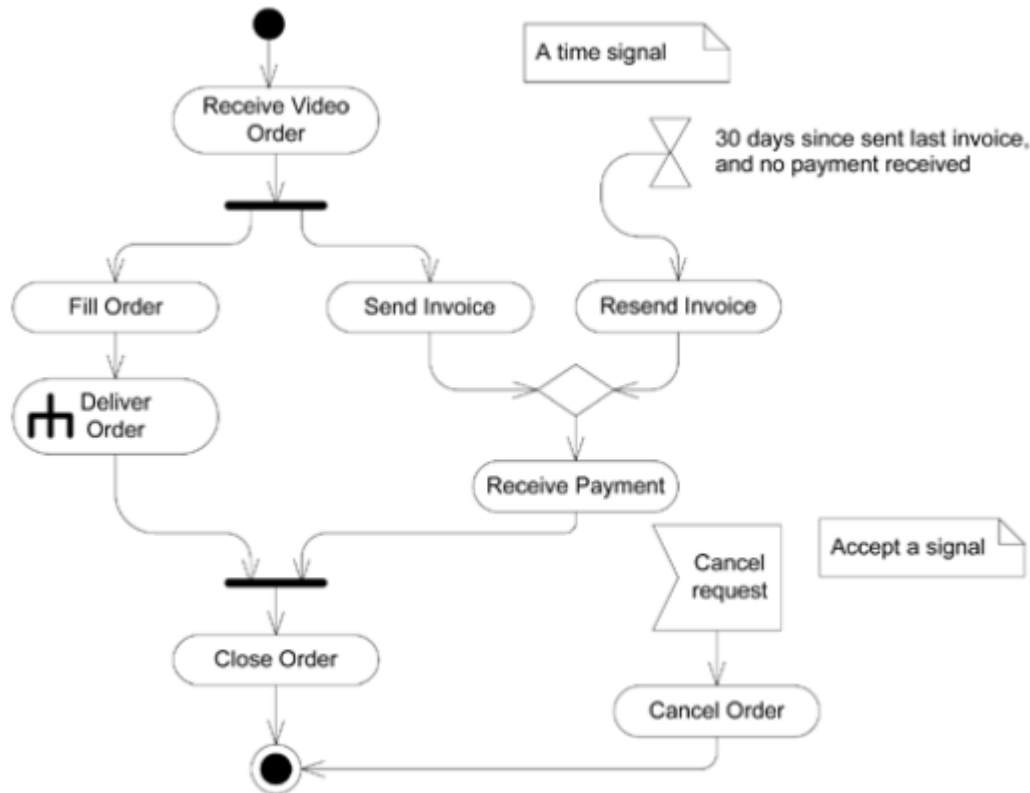


## Signals:

- Time signal: represents time event action or occurrence (denoted using an hour glass)
- Send signal action (represented with convex pentagon): is an invocation action that creates a signal from its inputs, and transmits it to the specified target object, leading to execution of an activity.
- Accept signal action (represented with a concave pentagon) : has no incoming edges, and is triggered by an accept signal event.



# Activity Diagram: Notations



Use of signals in Activity Diagram

Courtesy: Adapted from Applying UML and Patterns, Craig Larman, 3<sup>rd</sup> edition

# Plan ahead.....



Go through Lecture Slides:

- Module 4: Getting into Object Oriented Design :  
Refinements in Use Cases & Domain Model, Interaction Diagrams, State Transition Diagram, Activity Diagram

Agenda: Lecture 4

- Interaction Diagram
- State Transition Diagram