CSE 333 – Software Engineering

Pranta Sarker

Lecturer

Dept. of CSE, NEUB

Lecture 11

Outline

- System Modeling
- Unified modeling language
- UML types
- Activity diagram

System Modeling

- System modeling is the process of developing abstract models of a system, with each model presenting a different view or perspective of that system.
- System modeling has now come to mean representing a system using some kind of graphical notation, which is now almost always based on notations in the Unified Modeling Language (UML).
- System modelling helps the analyst to understand the functionality of the system and models are used to communicate with customers.

Unified Modeling Language

- Standardized modeling language consisting of an integrated set of diagrams.
- Developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems.

UML diagram types (5 essentials)

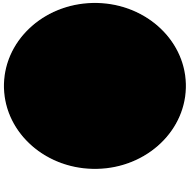
- Activity diagrams, which show the activities involved in a process or in data processing .
- Use case diagrams, which show the interactions between a system and its environment.
- Sequence diagrams, which show interactions between actors and the system and between system components.
- Class diagrams, which show the object classes in the system and the associations between these classes.
- State diagrams, which show how the system reacts to internal and external events.

Activity Diagram

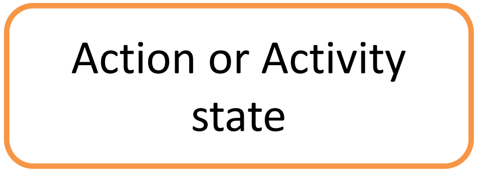
- Illustrate a series of actions or flow of control in a system
- Sequential and concurrent activities
- Similar to a flowchart or a data flow diagram
- It has both a beginning and a end state

Activity Diagram Notation

- Initial state – the starting state before an activity

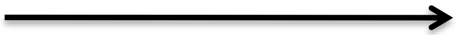


- Action or Activity state – activity represents execution of an action on objects or by objects. It is represented by rectangle with rounded corners.

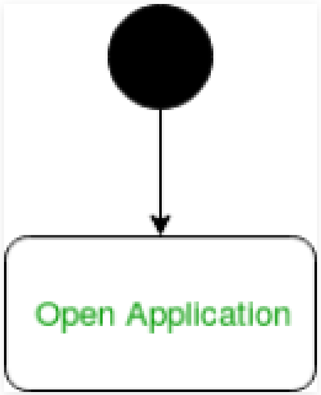


Activity Diagram Notation

- Action flow or control flow – it referred as an edge paths. They are used to show the transition from one activity state to another.

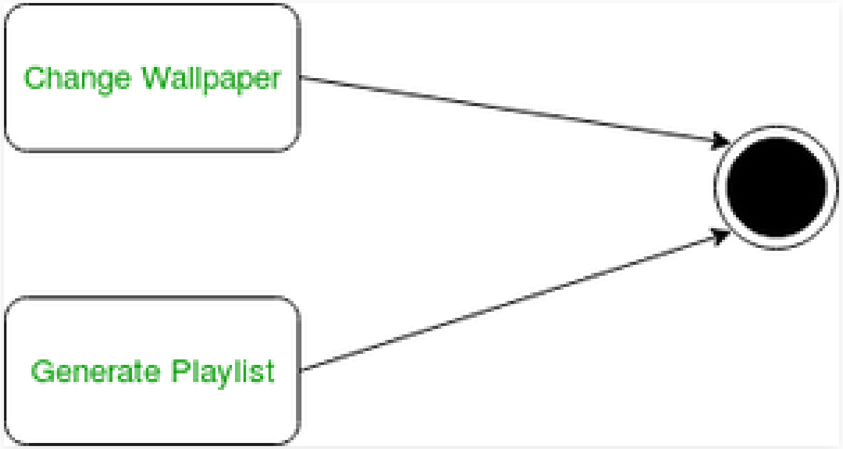


- For example: initial state + Activity state + action flow



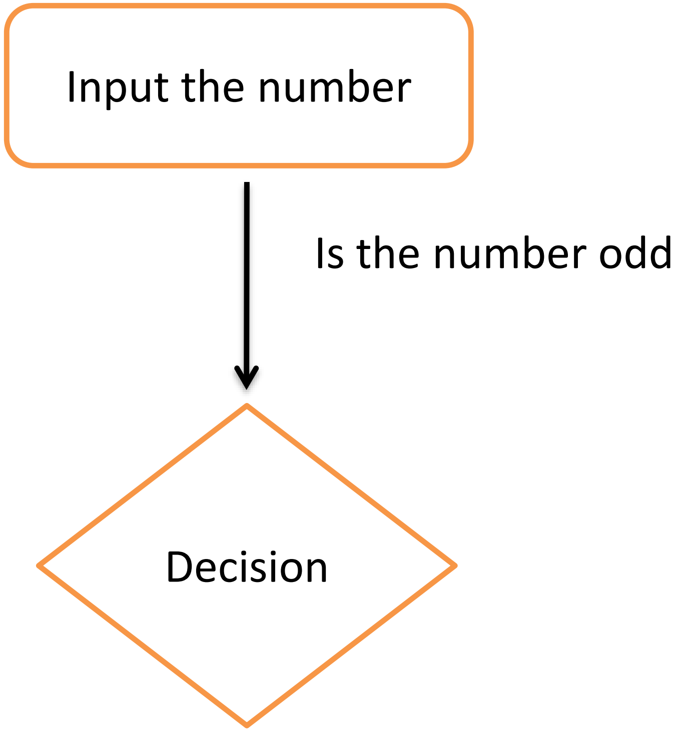
Activity Diagram Notation

- Any activity state can have one or multiple incoming and outgoing flows.



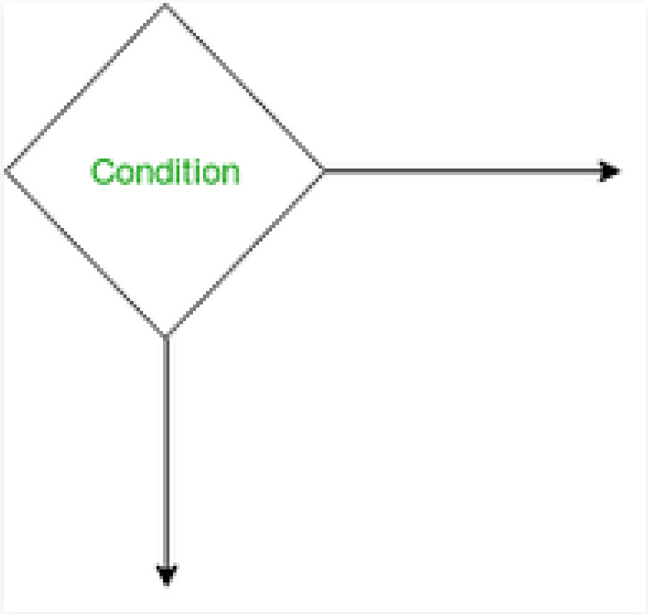
Activity Diagram Notation

- If there is a constraint to be adhered to while making the transition it is mentioned on the arrow.



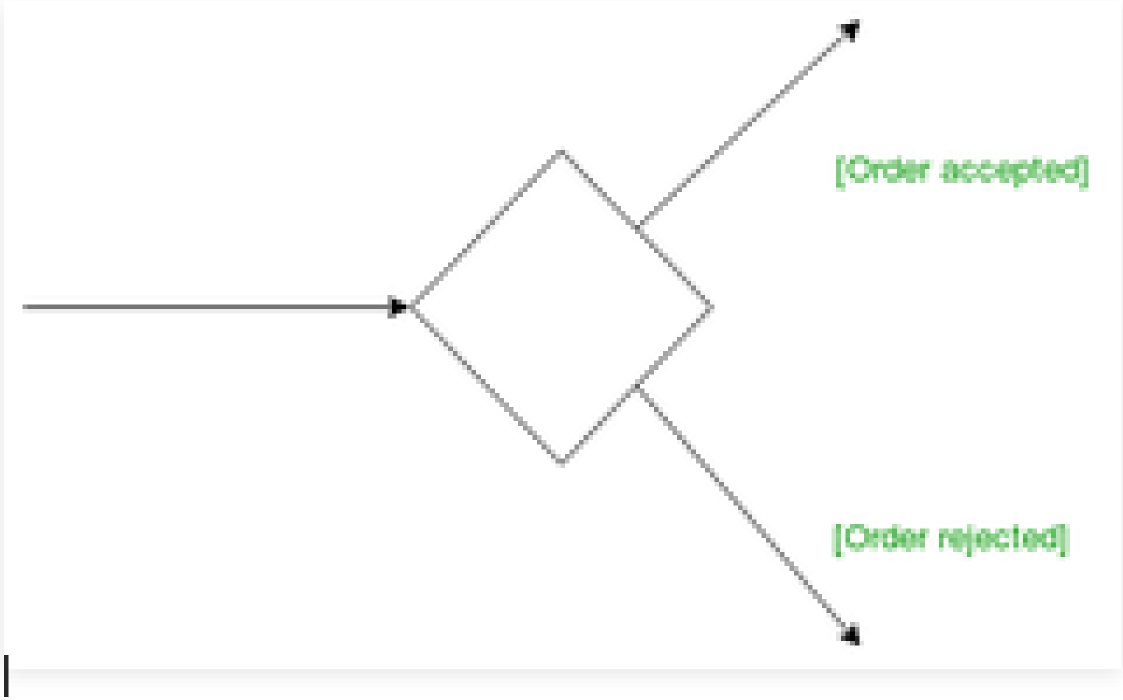
Activity Diagram Notation

- Decision node and Branching – when we need to make a decision before deciding the flow of control



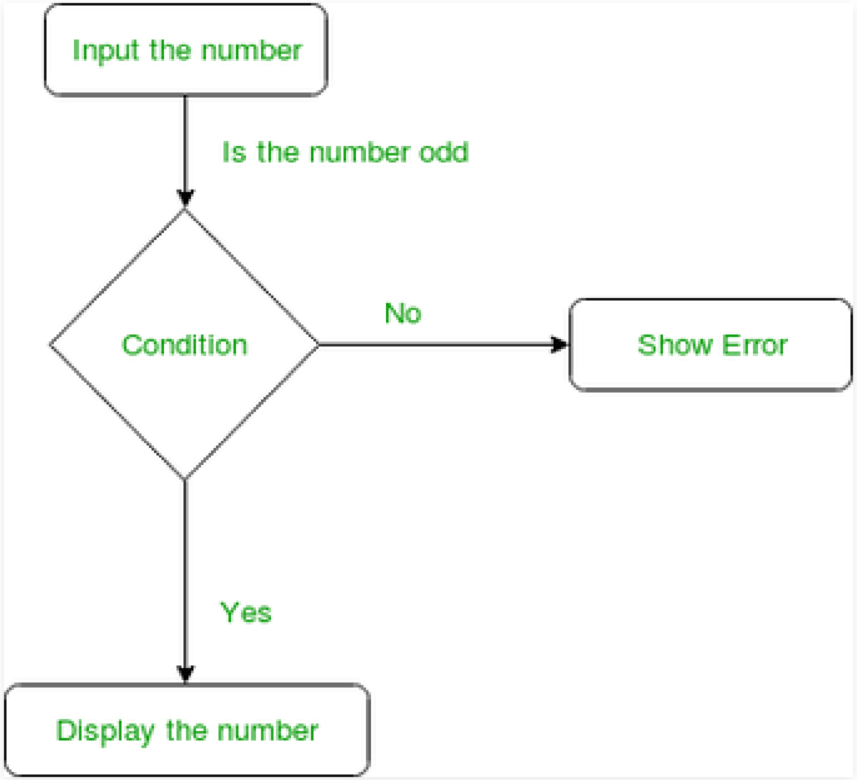
Activity Diagram Notation

- Guards – A Guard refers to a statement written next to a decision node on an arrow.



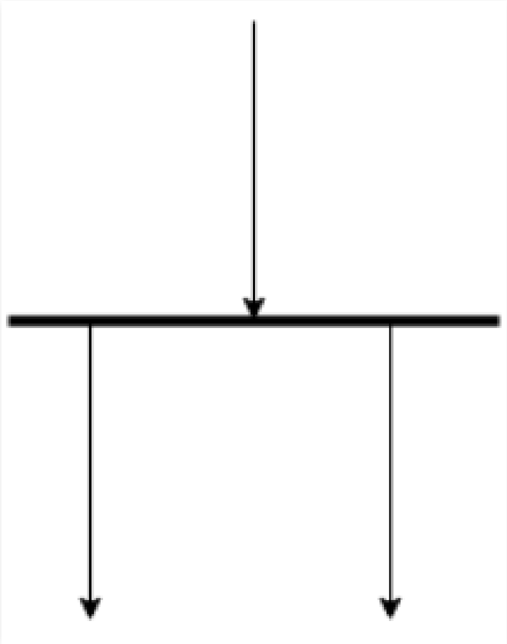
Activity Diagram Notation

- Decision node + Guards – It always includes two or more output arrows.



Activity Diagram Notation

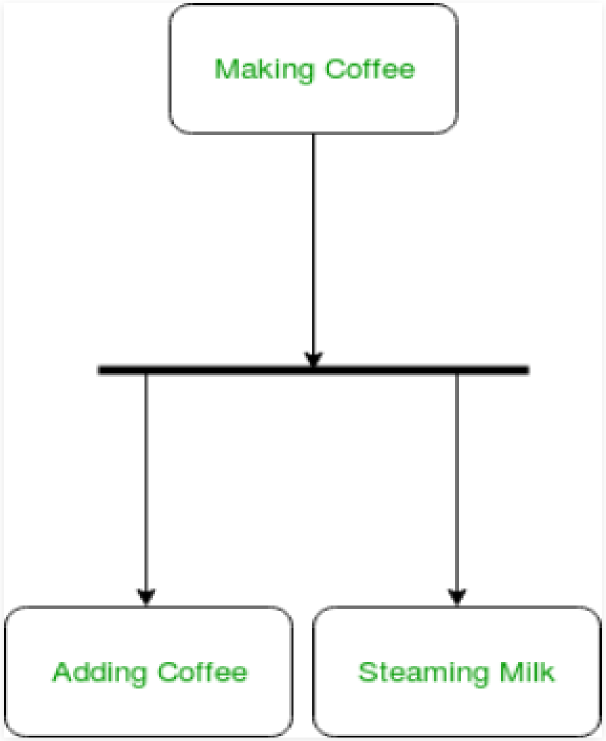
- Fork – fork nodes are used to support **concurrent activities** which means both activities will be executed concurrently.



Activity Diagram Notation: Example of Fork

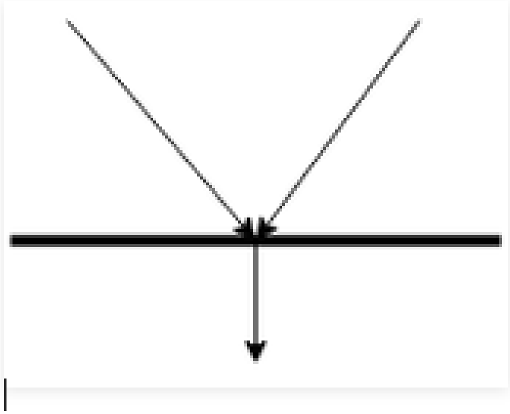
- Example: The activity of making coffee can be split into two concurrent activities and hence we use the fork notation.

{ Parent activity state }



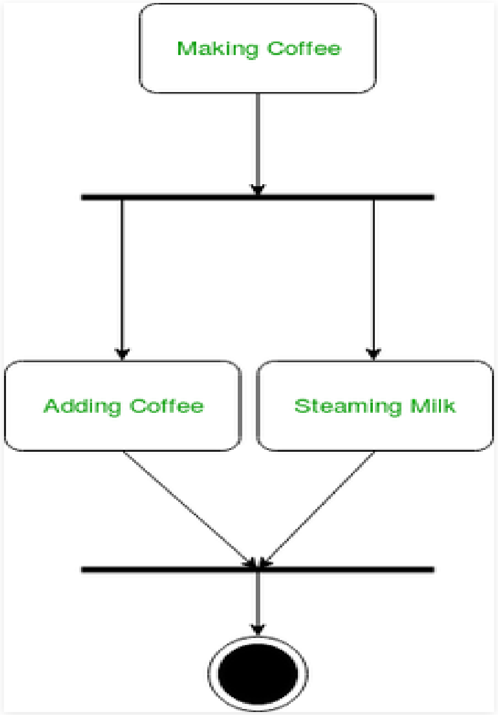
Activity Diagram Notation

- Join - Join nodes are used to support **concurrent activities converging into one**. For join notations we have **two or more incoming edges and one outgoing edge**. If your activity states are go into fork then you may need to do join.



Activity Diagram Notation: Join example

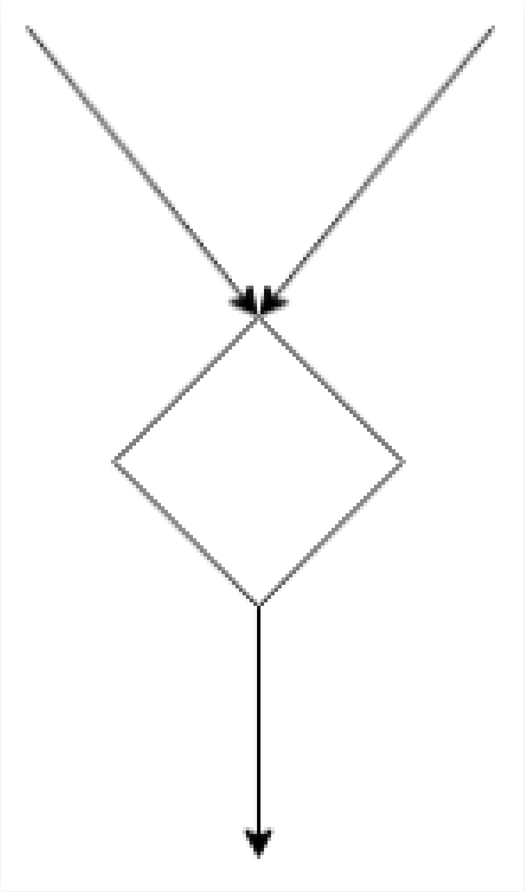
- For example – When both activities i.e. steaming the milk and adding coffee get completed, we converge them into one final activity.



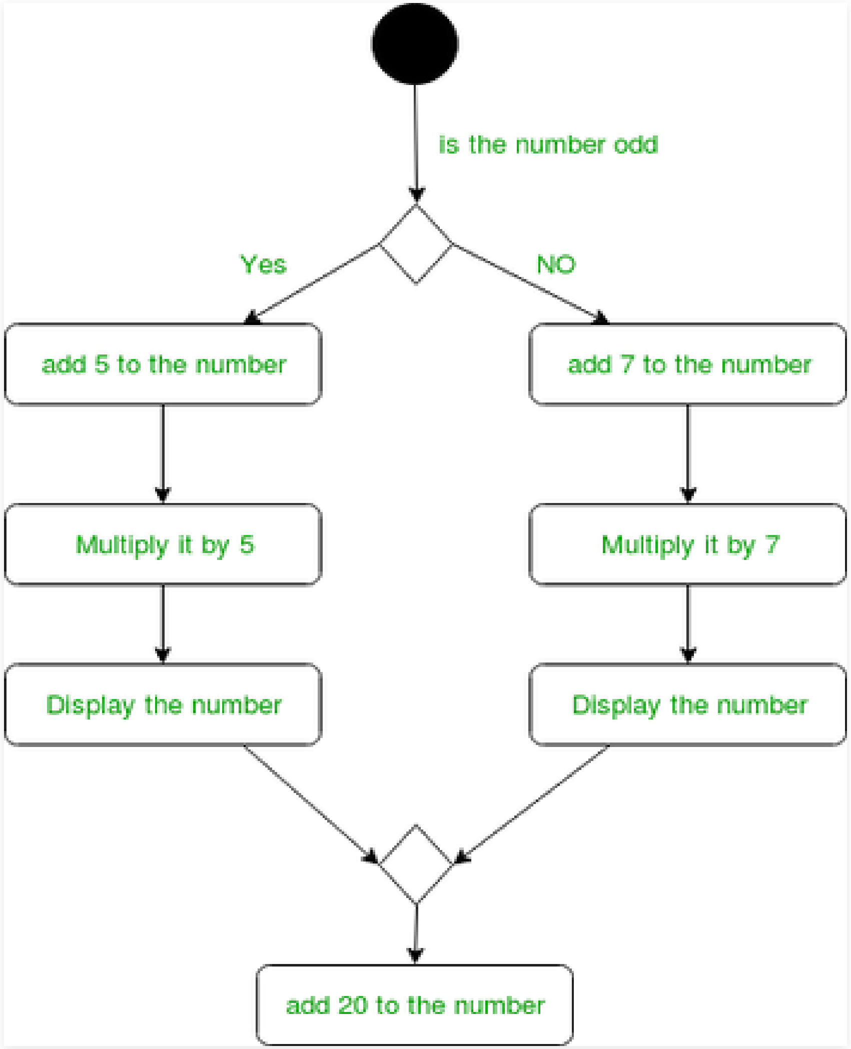
Activity Diagram Notation

- Merge or Merge Event – Scenarios arise when **activities which are not being executed concurrently have to be merged**. We use the merge notation for such scenarios. We can merge two or more activities into one if the control proceeds onto the next activity irrespective of the path chosen. **If your activity states are not go into fork** then you may need to do merge.

Activity Diagram Notation: Merge



Activity Diagram Notation: Merge Example

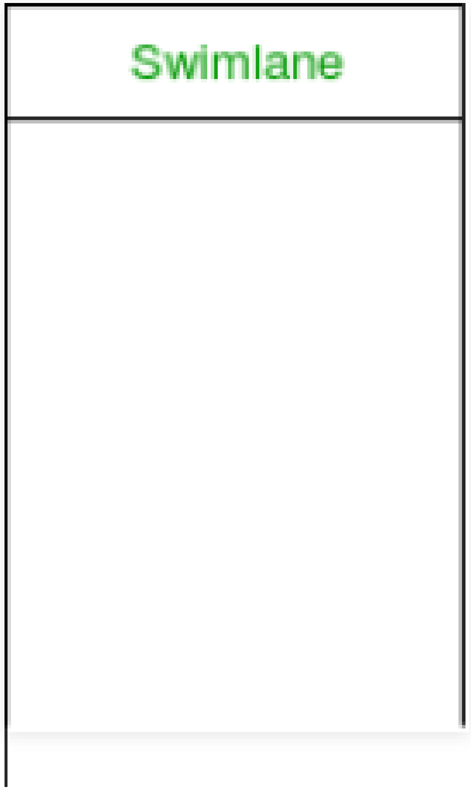


Activity Diagram Notation

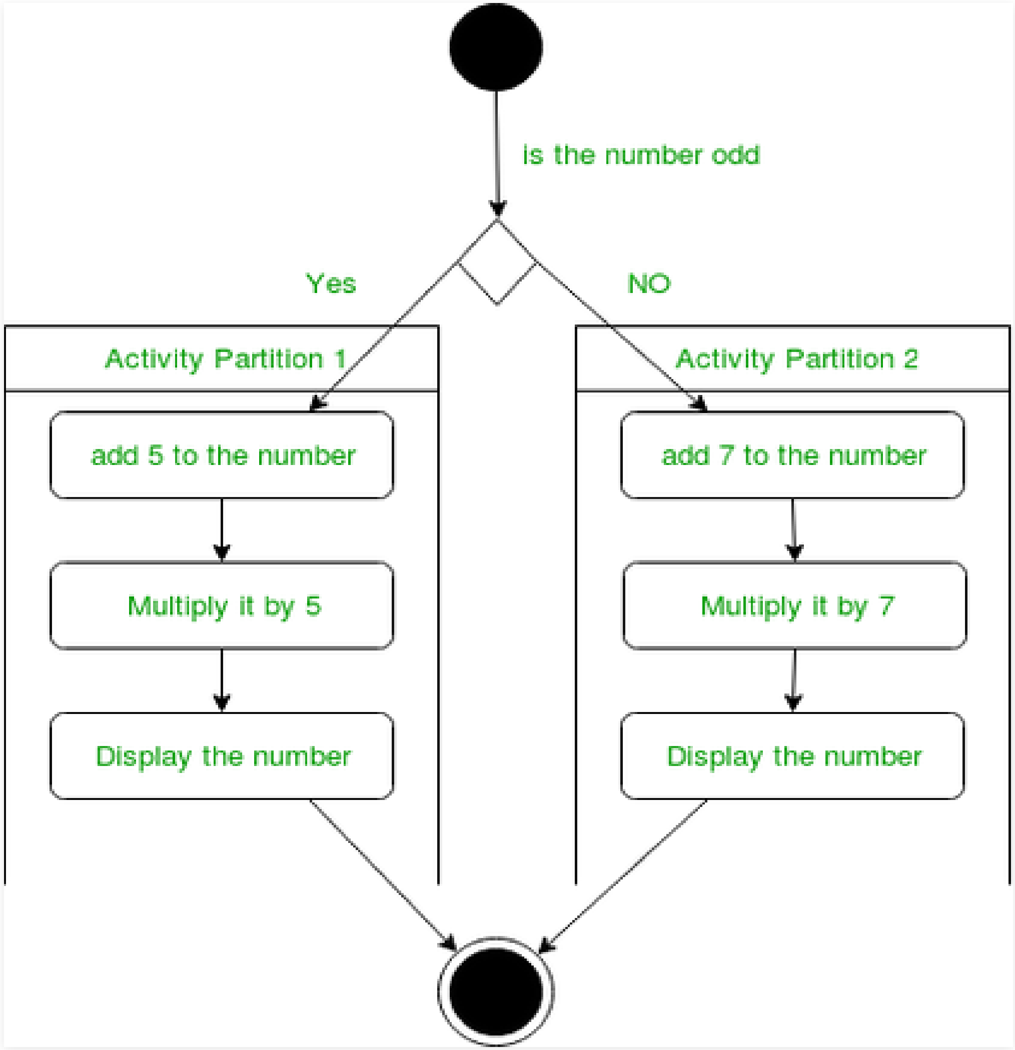
- Swimlanes – We use swimlanes for **grouping related activities** in one column. Swimlanes group related activities into one column or one row. Swimlanes can be **vertical and horizontal**. They usually give more clarity to the activity diagram. It’s similar to creating a function in a program. It’s not mandatory to do so, but, it is a recommended practice.

Activity Diagram Notation: Swimlane

[illegible]

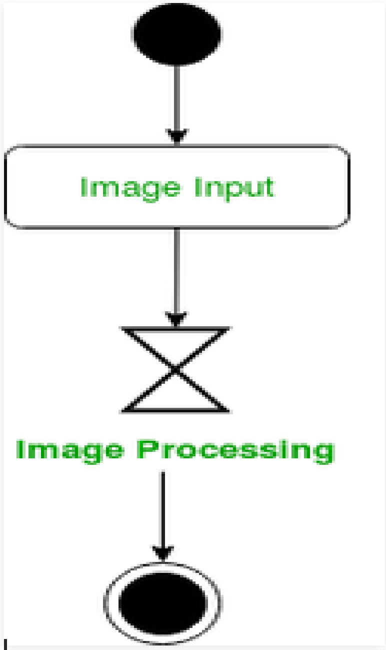


Activity Diagram Notation: Swimlane example



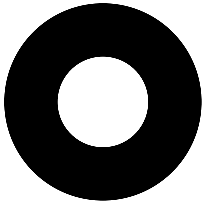
Activity Diagram Notation

- Time event – this is used when an **event needs some time to execute**. Let us assume that the processing of an image takes a lot of time. Then it can be represented as shown below.



Activity Diagram Notation

- Final State or End State – The state which the system reaches when a particular process or activity ends is known as a Final State or End State. We use a filled circle within a circle notation to represent the final state in a state machine diagram. A system or a process can have multiple final states.



Activity Diagram Notation: How to draw

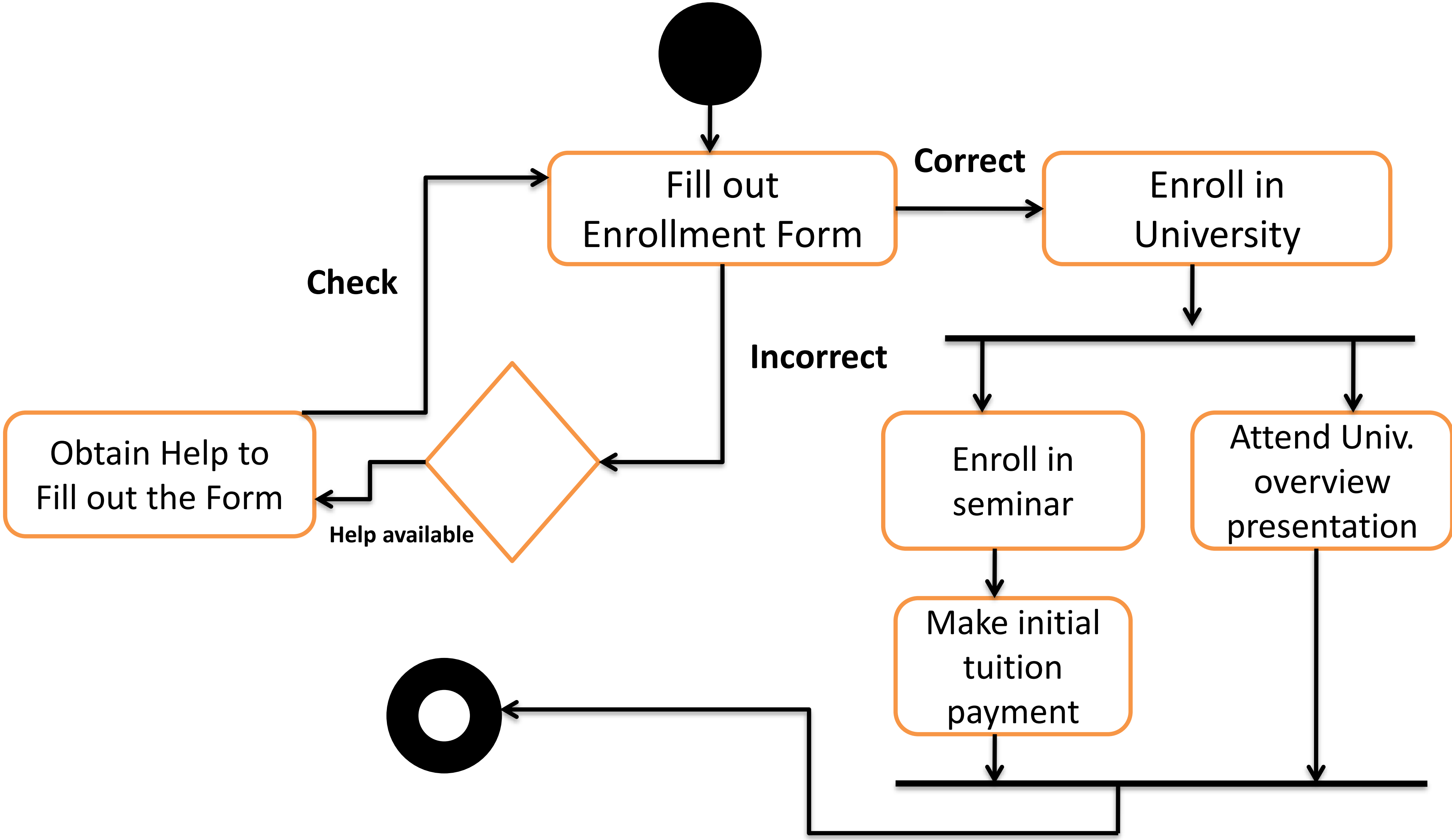
- Identify the initial and final state
- Identify the intermediate activities needed to reach the final state from initial state.
- Indentify the conditions and constraints which because it may change on the flow.
- Draw the diagram with appropriate notations.

Activity Diagram: Example

- This UML activity diagram example describes a process for student enrollment in a university as follows:

1. An applicant wants to enroll in the university.
2. The applicant hands a filled-out copy of Enrollment Form.
3. The registrar inspects the forms.
4. The registrar determines that the forms have been filled out properly.
5. The registrar informs student to attend in university overview presentation.
6. The registrar helps the student to enroll in seminars.
7. The registrar asks the student to pay for the initial tuition.

Activity Diagram: Example



Activity Diagram: Task

Q: Make an Activity Diagram with appropriate grouping for this example.

[***Hint:*** *use 'Swimlane' notation of Activity Diagram*]

Thank you!!