

# Повторение пройденного материала



Кто такой «актёр»?

# Кто такой «актёр»?

Актёр - тот, кто взаимодействует с системой.  
Благодаря которому и возникает прецедент.  
Обычно он изображен в виде человечка и назван  
каким-то существительным.

Каких типов они бывают?

# Каких типов они бывают?

## Primary Actors

Initiates the use of the system

## Secondary Actors

Reactionary

Что такое прецедент?

# Что такое прецедент?

Прецедент – системная функция, а именно процесс, который может быть автоматическим или мануальным.

Поясните разницу между включением  
(`<<include>>`) и расширением  
(`<<extend>>`)



Для каких целей нужны USE CASE DIAG

# Для каких целей нужны USE CASE DIAG?

Данный вид диаграмм описывает функциональные требования системы.

Что такое диаграмма классов и для чего она нужна?

# Что такое диаграмма классов и для чего она нужна?

**Диаграммы классов** – это способ представления внутренней структуры класса и интерфейса взаимодействия с ним.

**Целью** создания **диаграммы классов** является графическое представление статической структуры декларативных элементов системы

С каких точек зрения можно посмотреть  
на «модель»?

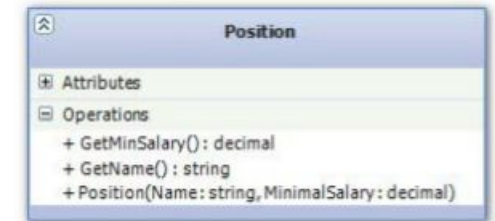
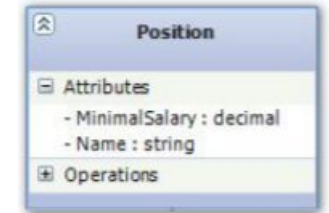
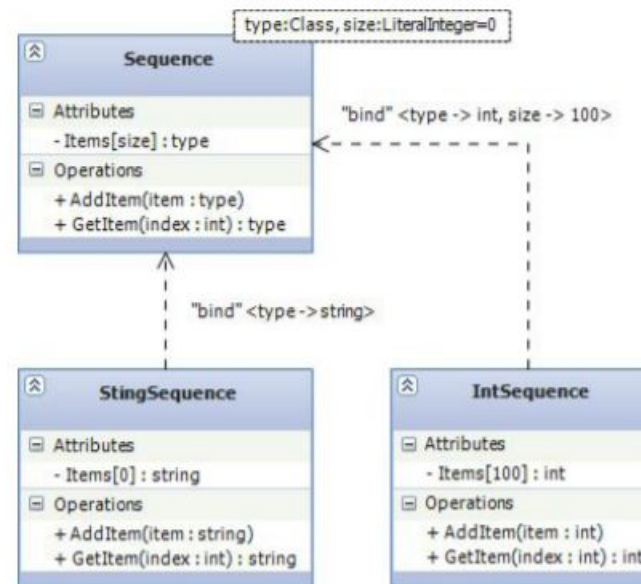
# С каких точек зрения можно смотреть на «модель»?

- Концептуальная точка зрения
- Точка зрения спецификации
- Точка зрения реализации

Перечислите основные понятия,  
касающиеся диаграммы классов.

# Перечислите основные понятия, касающиеся диаграммы классов.

- Атрибуты
- Операции
- Объекты
- Интерфейс
- Шаблоны (каких типов?)





Какие существуют отношения между  
классами?

# Какие существуют отношения между классами?

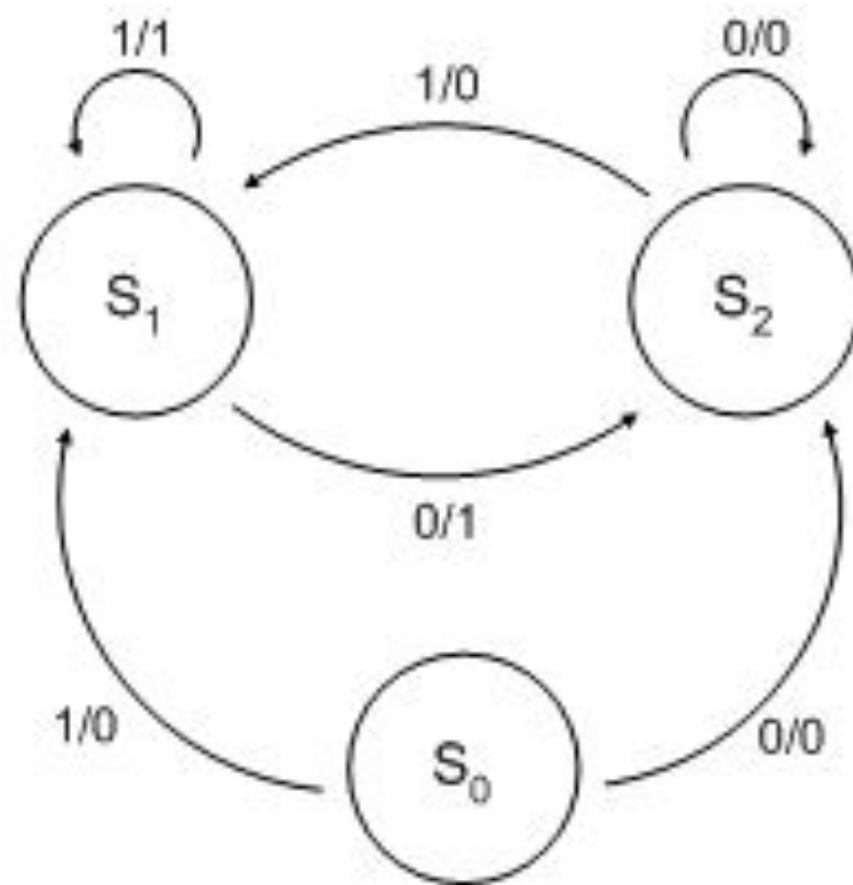
- **Отношение зависимости**
- **Отношения ассоциации**
- **Отношения реализации**
- **Отношения обобщения**

# Диаграмма состояний

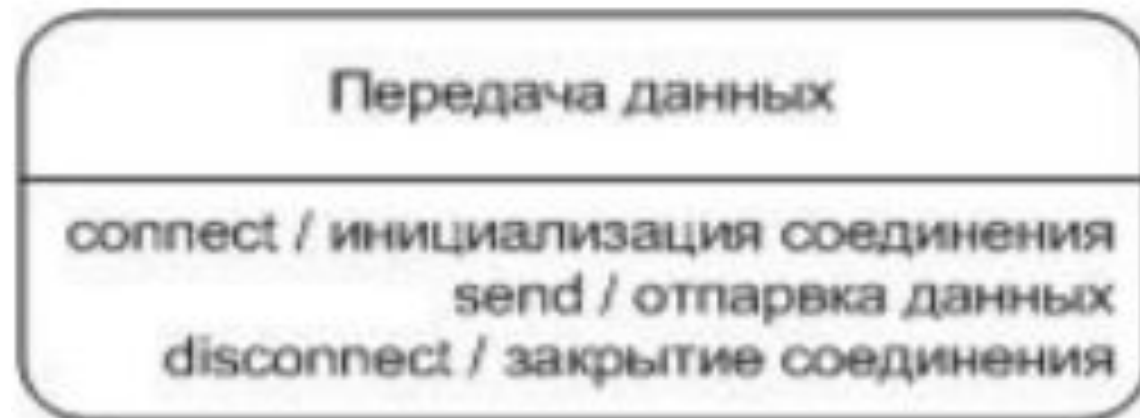
# Диаграмма состояний





# Диаграмма состояний. Автомат



# Диаграмма состояний. Действия



# Диаграмма состояний. Псевдосостояния

Начальное состояние	
Начальное состояние с переходом	

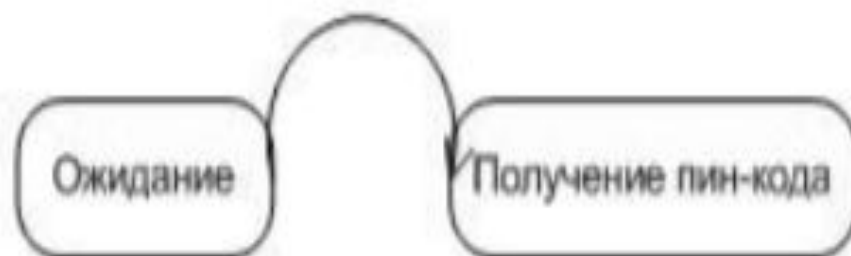
# Диаграмма состояний. Переход





# Диаграмма состояний. Событие

если: появление карточки/ получить пин-код



# Диаграмма состояний. Сторожевое условие



# Диаграмма состояний. Выражение действия

отсутствие ответа [превышен интервал ожидания] / разорвать соединение, освободить ресурсы



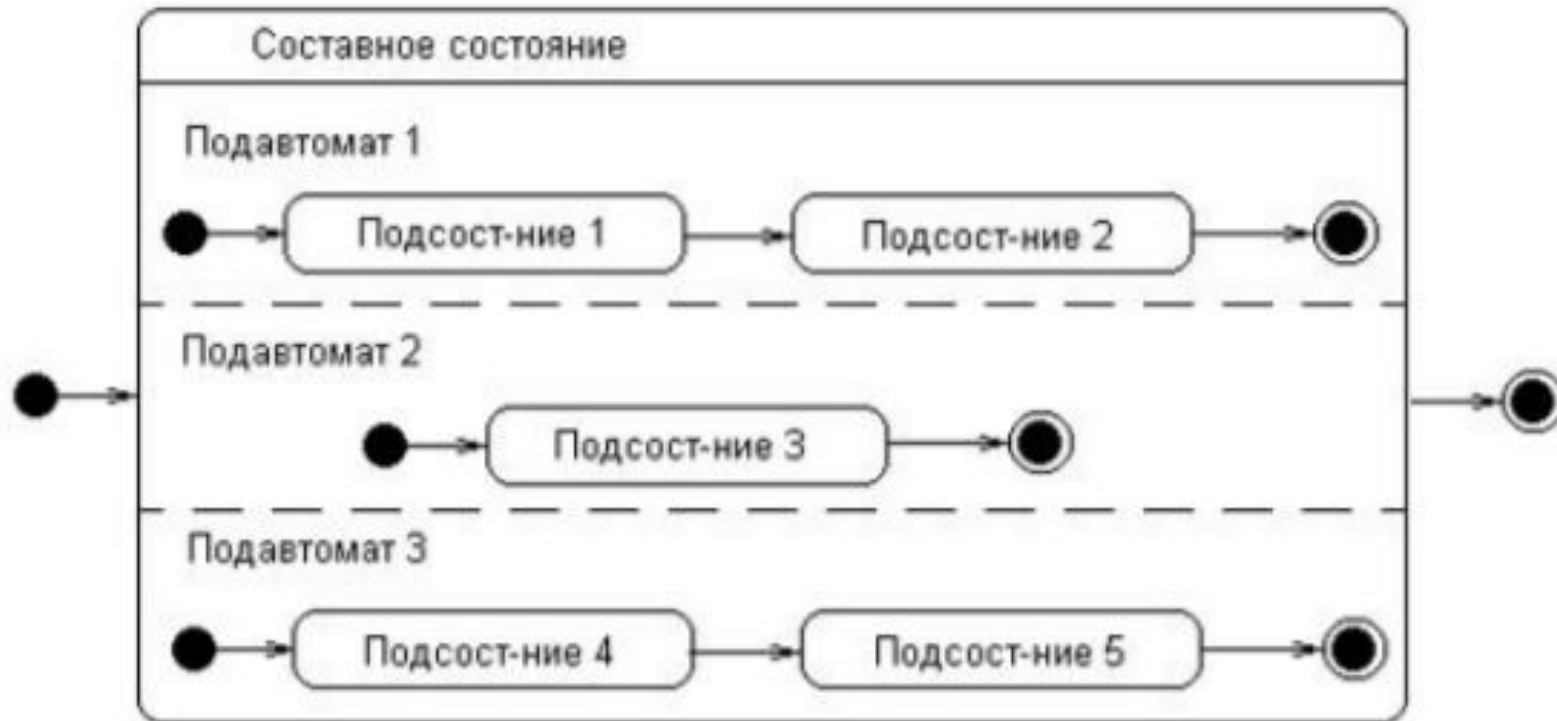
# Диаграмма состояний. Составное подсостояние



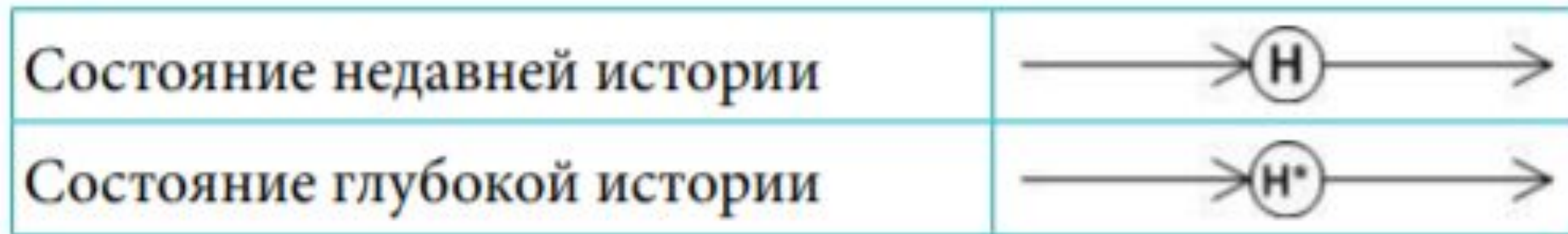
# Диаграмма состояний. Последовательные подсостояния



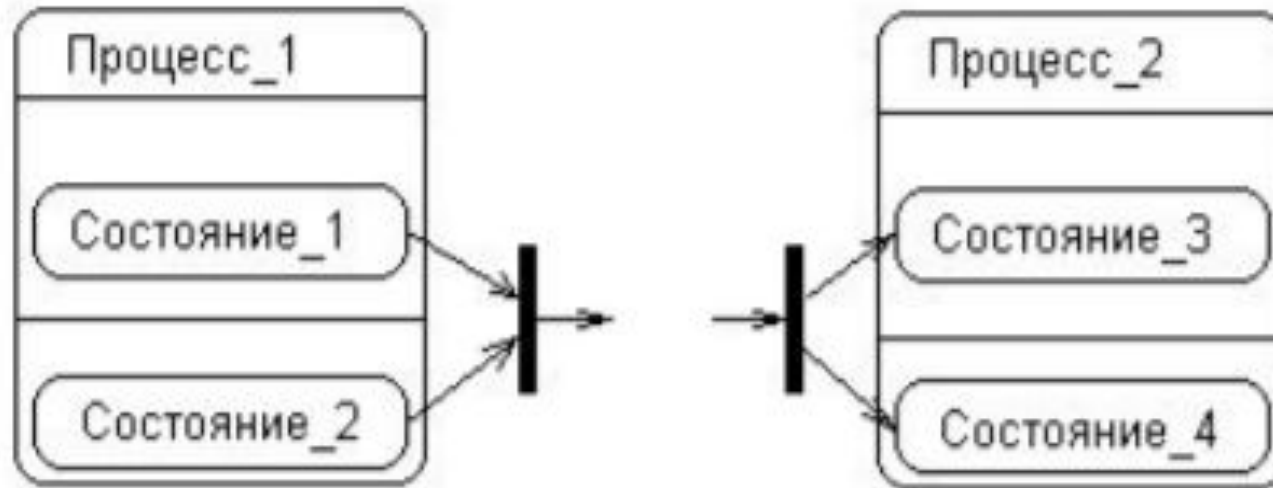
# Диаграмма состояний. Параллельные подсостояния



# Диаграмма состояний. Историческое состояние

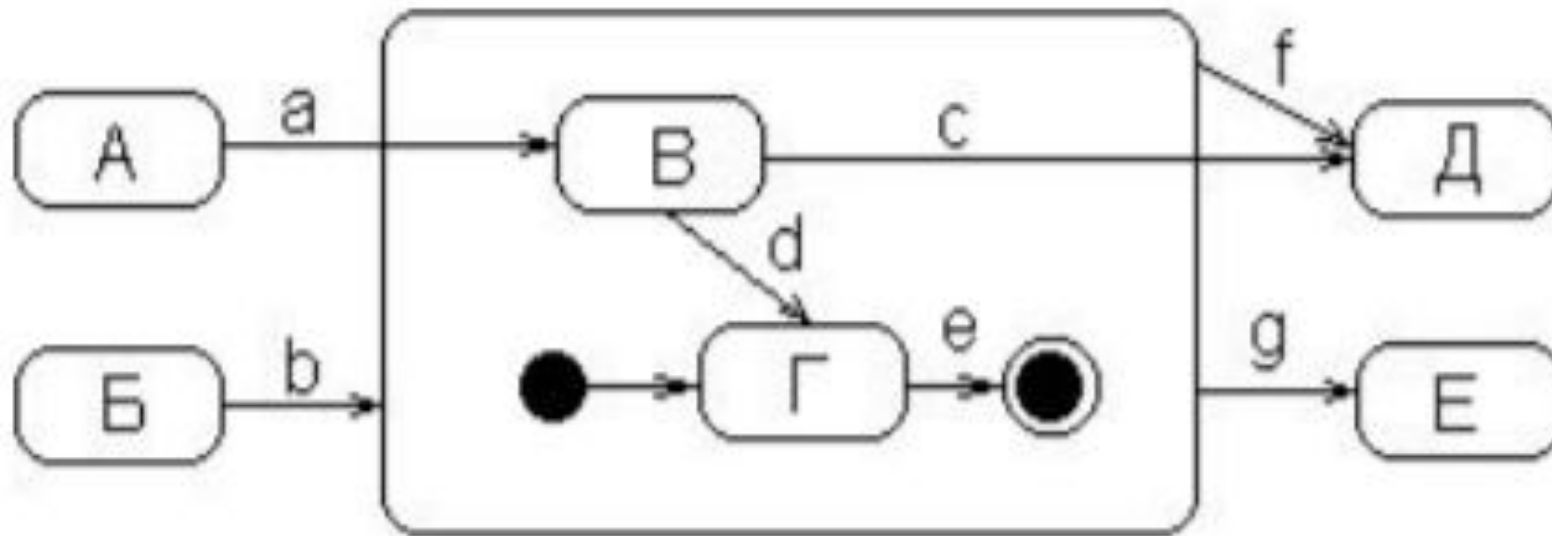


# Диаграмма состояний. Сложные переходы

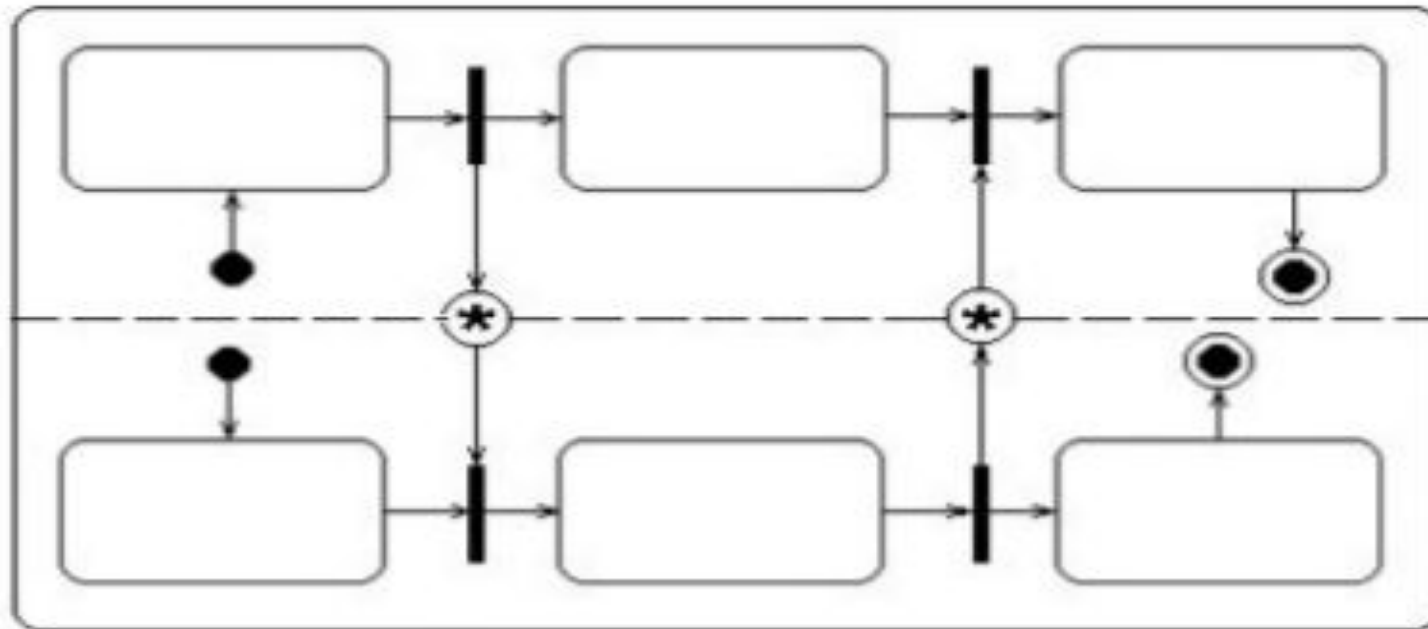




# Диаграмма состояний. Переходы между составными состояниями



# Диаграмма состояний. Синхронизирующие состояния



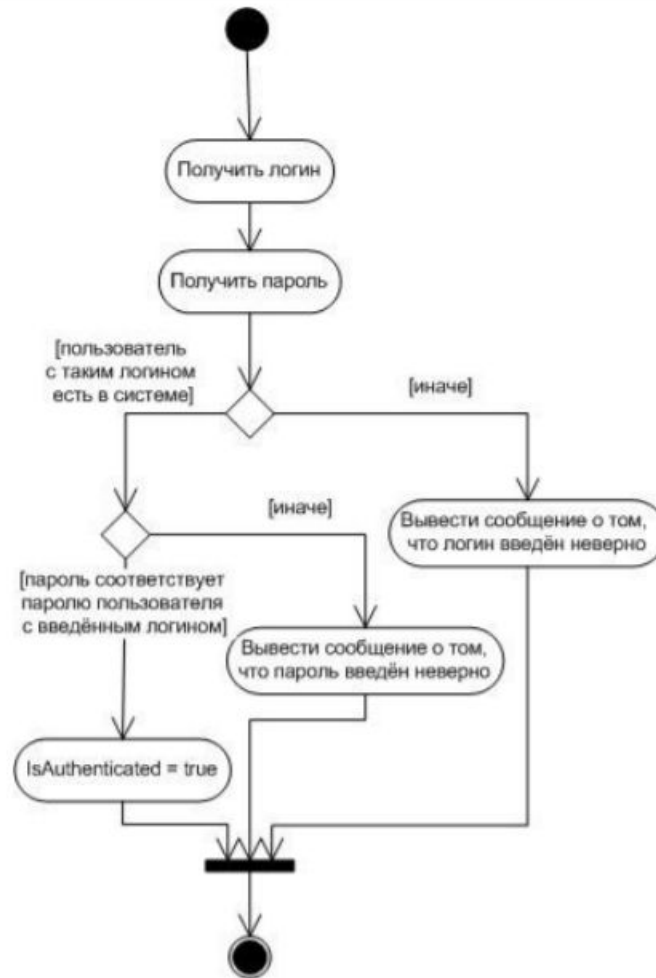
# Диаграмма деятельности

# Диаграмма состояний

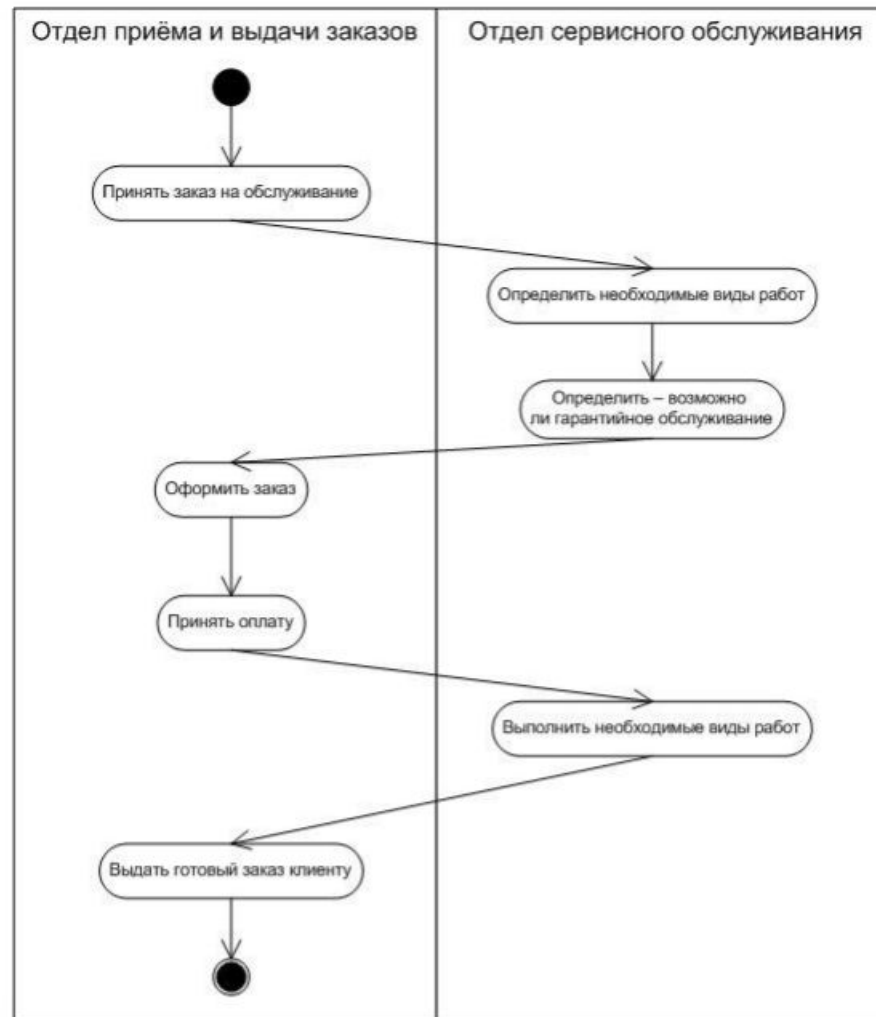
Основной целью использования диаграммы деятельности является визуализация особенностей реализации операций классов, а так же описание реакций на внутренние события системы.



# Диаграмма состояний. Переходы



# Диаграмма состояний. Дорожки



# Диаграмма состояний. Объекты

