

# Математические методы верификации схем и программ

mk.cs.msu.ru → Лекционные курсы  
→ Математические методы верификации схем и программ

## Семинар ОЗ

Упражнения для Uppaal

Лектор:

**Подымов Владислав Васильевич**

E-mail:

**valdus@yandex.ru**

ВМК МГУ, 2023/2024, осенний семестр

# Упражнение 1: изучаем примеры

Открыть, понять, выполнить симуляцию и верификацию для следующих примеров, поставляемых со средством Uppaal:

`<uppaal>/demo/`

1. bridge
2. 2doors
3. fischer

## Упражнение 2: как щёлкать мышкой

Система состоит из пользователя и драйвера

Пользователь:

- ▶ Может нажимать на кнопку мыши (а может и не нажимать)
- ▶ Нажимает на кнопку не чаще чем раз в секунду  
(в связи с физиологическими особенностями)
- ▶ В остальном нажимает на кнопку как угодно

Драйвер:

- ▶ Находясь в режиме ожидания, ждёт нажатия на кнопку мыши
- ▶ Когда кнопка нажата, немедленно выходит из режима ожидания:
  - ▶ Если менее чем через две секунды произошло второе нажатие, то констатирует двойное нажатие и немедленно переходит в режим ожидания
  - ▶ Иначе через две секунды выносит решение об одинарном нажатии и немедленно переходит в режим ожидания

Требуется представить эту систему в виде сети временных автоматов и убедиться в корректности поведения драйвера

## Упражнение 3: умная лампа

Лампа может находиться в трёх режимах: **выкл**, **слаб**, **сил**

Начальный режим — **выкл**

Режимы переключаются при помощи кнопки, на которую можно нажимать и после этого отпустить

Когда кнопка отпускается, лампа немедленно меняет свой режим:

- ▶ Если кнопка была нажата менее чем секунду:
  - ▶ **выкл**→**слаб**
  - ▶ **слаб**→**выкл**
  - ▶ **сил**→**выкл**
- ▶ Если кнопка была нажата хотя бы секунду:
  - ▶ **выкл**→**сил**
  - ▶ **слаб**→**сил**
  - ▶ **сил**→**слаб**

Требуется представить лампу с кнопкой в виде сети временных автоматов и убедиться в корректности переключения режимов лампы

## Упражнение 4: китайский жонглёр



## Упражнение 4: китайский жонглёр

Жонглёр крутит  $N$  тарелок

Пока тарелка крутится, она не падает и не разбивается, а как только перестаёт крутиться, то немедленно падает и разбивается

В самом начале все тарелки раскручены так, что гарантированно будут крутиться 5 секунд

Жонглёр может раскручивать тарелки по одной за раз, какие хочет и сколько хочет

Пока жонглёр раскручивает тарелку, она не перестаёт крутиться

Если жонглёр раскручивал тарелку

- ▶ хотя бы 1 секунду подряд, то тарелка гарантированно будет крутиться ещё хотя бы 3 секунды
- ▶ строго больше 2 секунд подряд, то тарелка гарантированно будет крутиться ещё хотя бы 5 секунд

Сколько тарелок жонглёр может гарантированно раскручивать сколь угодно долго так, чтобы они не падали?

## Упражнение 5: отладка модели банкомата

Обсуждаемая модель *atm* есть в *zip*-архиве на странице курса



Эрик хочет снять деньги со своего счёта через банкомат, но что-то пошло не так

Требуется исправить модель так, чтобы Эрик мог снять столько денег, сколько позволяют его счёт и объём наличных денег в банкомате, и ни цента больше

## Упражнение 5: отладка модели банкомата

Смысл каналов:

- ▶ `bank_card`: вставка и возврат банковской карты
- ▶ `request`: запрос к банкомату на выдачу 10 у.е.
- ▶ `cash`: выдача денег банкоматом

Полученные наличные деньги Эрик кладёт в карман (`cash_in_pocket`)

Если в банкомате (в `in_till`) достаточно наличных денег, то он запрашивает у банка разрешение на выдачу (`ask_permission`), и разрешение либо выдаётся (`OK`), либо нет (`not_OK`) в зависимости от состояния счёта Эрика (`balance`)

В любом случае после завершения всех операций карта возвращается Эрику

## Упражнение 6: отладка ж/д переезда

Обсуждаемая модель *traingate* есть в zip-архиве на странице курса

$N$  поездов пытаются пересечь железнодорожный переезд с одним путём (узкий мост)

После появления в зоне видимости машинист поезда выполняет команды диспетчера: остановиться (`stop`), продолжать движение (`go`)

Когда поезд появляется в зоне видимости (`appr`), но всё ещё далеко от моста, он может остановиться

Когда поезд достаточно близок к мосту, он занимает единственный путь переезда, и после пересечения моста уезжает (`leave`) и освобождает путь

Требуется исправить модель так, чтобы все формулы на вкладке верификации были выполнены