1. Цель работы

Цель лабораторной работы – изучение примитивов синхронизации и методов работы с ними, решение классической задачи узкого моста и тестирование решения в рамках операционной системы PintOS.

# ХОД РАБОТЫ

## Изменения в системе планирования Pintos

Были созданы четыре семафора для регулирования движения автомобилей всех видов по мосту: обычных автомобилей, движущихся вправо; обычных автомобилей, движущихся влево; автомобилей скорой помощи, движущихся вправо; автомобилей скорой помощи, движущихся влево. С помощью функции семафора sema\_down() ограничивается доступ к мосту для автомобилей, а функция sema\_up() открывает путь на мост.

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Одно из условий- нахождение на мосту не более 2-х машин, движущихся в одном направлении. Был создан счетчик машин на мосту и переменная направления движения на мосту в данный момент времени.

Были созданы переменные для подсчета всех автомобилей, желающих проехать по мосту, по виду. Данные счетчики позволяют контролировать условие: автомобили скорой помощи должны проезжать по мосту вне очереди. Таким образом, пока таких автомобилей не ноль, они запускаются на мост, пока обычные автомобили ждут своей очереди.

Инициализация всех глобальных переменных, в том числе и семафоров:

Изображение выглядит как текст, снимок экрана, Шрифт

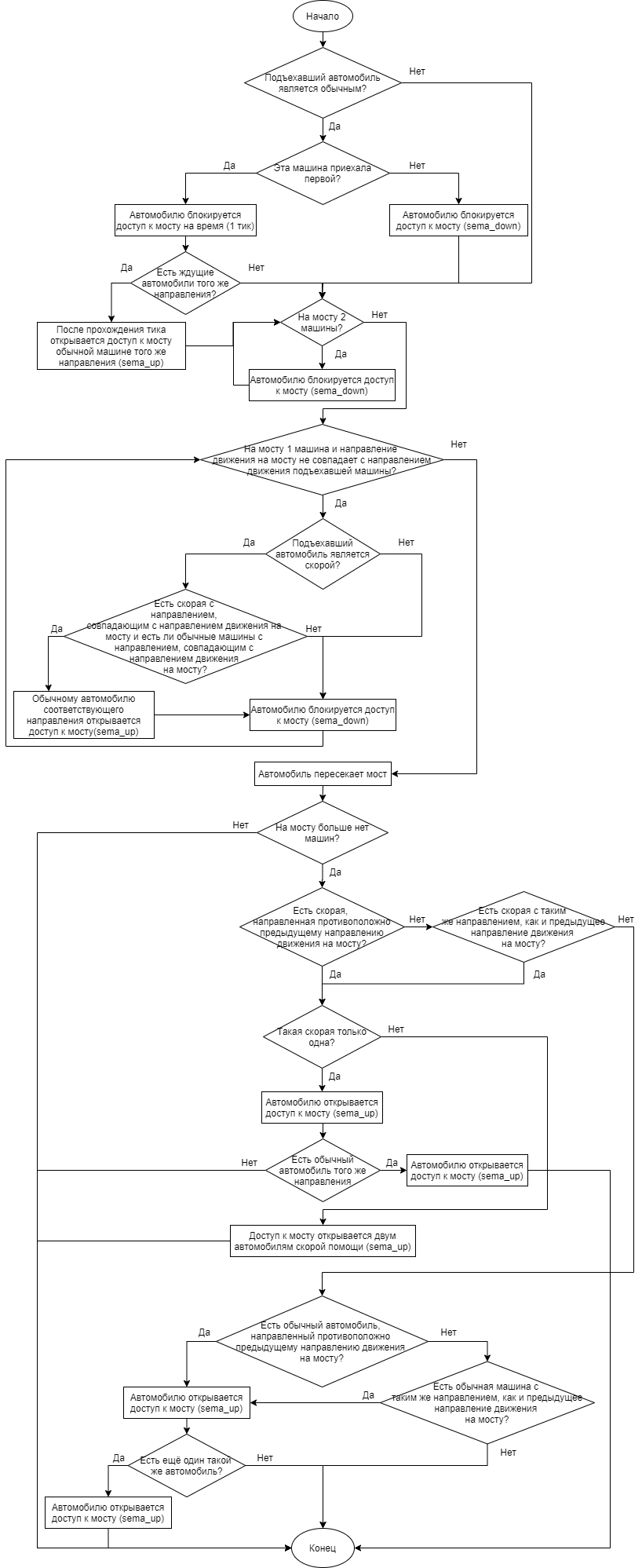
Автоматически созданное описание

Реализована функция прибытия автомобилей к мосту arrive\_bridge(). Если прибывший автомобиль является обычным, то ему блокируется доступ к мосту, чтобы первым делом пересекли мост автомобили скорой помощи. Однако первые две машины, если у них одинаковое направление или только одна, проезжают сразу, так как они первые, и никаких скорых еще в очереди не стоит.

Пока на мосту две машины, автомобили записываются в соответствующий список автомобилей, ждущих проезда, также к соответствующему счетчику ждущих прибавляется 1. Также в этой функции ведется подсчет автомобилей, находящихся на мосту, и запоминается направление движения на нем.

После была реализована функция выезда с моста exit\_bridge(). При выезде автомобиля с моста из счетчика автомобилей, находящихся на мосту, вычитается 1. Если на мосту нет машин, то выбирается, какой машине откроется доступ к мосту. При каждом открытии пути к мосту из счетчиков автомобилей, ждущих у въезда на мост, вычитается единица.

## Диаграмма взаимодействия процессов



# результаты тестов

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, меню, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, шаблон

Автоматически созданное описание

# дополнительное задание

Вариант 7. Спящий парикмахер(семафоры):

У парикмахера есть одно рабочее место и приемная с несколькими стульями. Когда у парикмахера нет посетителей — он спит в своем кресле. Если приходит посетитель — он будит парикмахера, он усаживает клиента в кресло и подстригает его. Когда парикмахер заканчивает подстригать клиента, он отпускает клиента и затем идет в приёмную, чтобы посмотреть, есть ли там ожидающие клиенты. Если они есть, он приглашает одного из них и стрижет его. Если ждущих клиентов нет, он возвращается к своему креслу и спит в нем.

Изображение выглядит как текст, снимок экрана, меню

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, меню, дизайн

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, меню

Автоматически созданное описание

# Вывод

Были изучены примитивы синхронизации и методы работы с ними, решена классическая задача узкого моста и протестировано решение в рамках операционной системы PintOS.