Согласовано:

Менеджер компетенции

Т89 «Архитектор интеллектуальных систем управления»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Ткаченко

Главный эксперт

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_К.А. Олейников

КОНКУРСНОЕ  
ЗАДАНИЕ

КОМПЕТЕНЦИЯ

АРХИТЕКТОР ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

УПРАВЛЕНИЯ

Количество часов на выполнение задания: 15.

****

ВВЕДЕНИЕ

Описание задания

В предлагаемом Вам задании необходимо выполнить следующие задачи:

Заказчику необходимо создать рабочее пространство, позволяющее проводить тестирование новой технологии, такой как искусственный интеллект. Реализовать базу данных для внесения данных тестирования.

Вам необходимо протестировать виртуальную сеть используя созданные Вами технологии. Для этой задачи нужно будет создать систему управления объектами сети. А также разработать алгоритмы искусственного интеллекта для системы управления.

В заключении заказчик просит вас провести тестирование вашей интеллектуальной системы управления. Составить презентацию проделанной работы и создать инструкцию по использованию программы.

Модуль 1

Создать рабочее пространство, добавить стандартные объекты и реализовать функционал с помощью которого пользователь сможет самостоятельно добавлять и настраивать новые объекты.

Модуль 2

Добавить функционал по управлению транспортом, анимации. Задать правила поведения транспорта.

Модуль 3

Создание базы данных с таблицами для хранения информации о работе программы и значениями настройки объектов.

Модуль 4

Подключение, настройка и обучение модели искусственного интеллекта. Визуализация обучения модели, добавление счётчиков и таймеров.

Модуль 5

Документирование работы. Создание презентации для заказчика и инструкции по использованию программы.

Модуль 1 Разработка рабочего пространства и графического интерфейса (4 часа)

Заказчик желает, чтобы архитектор реализовал рабочее пространство, в котором должна быть возможность проводить испытание новых технологий по искусственному интеллекту. Пользователь должен иметь возможность добавить объекты в виртуальную сеть и право изменять характеристики этих объектов.

Задание:

Все графические элементы для реализации программы необходимо брать из архива с ресурсами. Ресурсы хранятся в системе контроля версий по ссылке <https://github.com/WSRAICS/ChampionshipX>

При создании программы необходимо предусмотреть название программы: «World Skills X». Программа должна иметь иконку в виде светофора.

Создать схему дорог города. Необходимо предусмотреть наличие всех элементов дорог с правильным направлением движения (правостороннее), их расположение, согласно шаблону (рисунок 1). Добавить объекты: «светофор», «пешеходный переход», «пешеход» и «движение запрещено» всех видов.

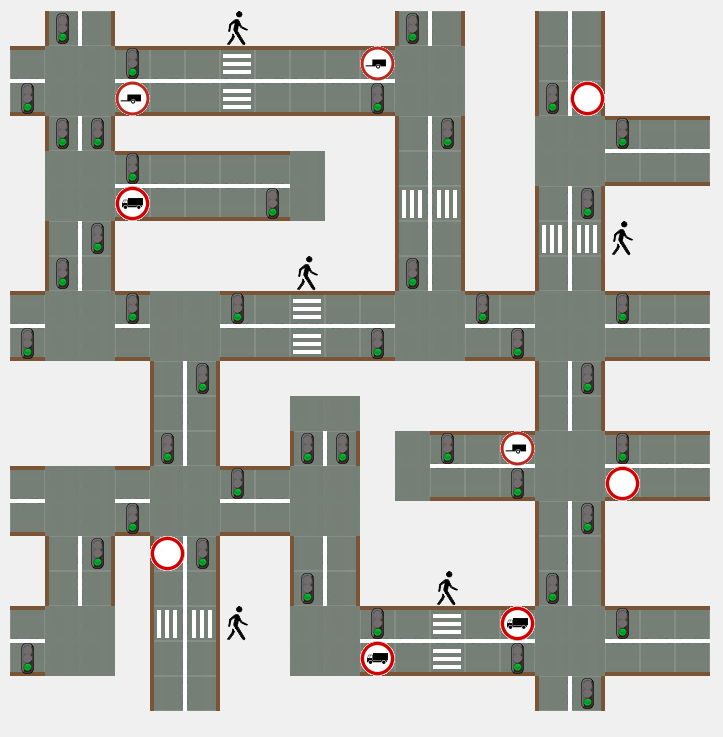


Рисунок 1 – Схема города

|  |  |
| --- | --- |
| https://auto.mail.ru/img/html/pdd/signs/3_2.gif | Движение запрещено |
|  | Проезд грузовых автомобилей запрещён |
|  | Проезд автомобилей с прицепом запрещён |
| TLyellow | Светофор. |
| Zhorizontal | Пешеходный переход. |
|  | Пешеход. |

В пользовательский интерфейс добавьте label, в котором отображаются координаты ячейки дороги, на которую пользователь наводит указатель мыши. Выполнить создание пользовательского интерфейса, позволяющего пользователю добавлять новые объекты и удалять старые объекты со схемы дорог города. Для реализации функции добавления после нажатия на кнопку «Добавить объект» появляются четыре кнопки с картинками объектов (светофор, пешеходный переход, пешеход, движение запрещено). После выбора одной из четырех кнопок пользователь нажимает на рабочее пространство, и создаётся объект. Объект может быть создан только в конкретном, определенном для него месте.

Допустимое расположение объектов:

* Светофор – въезд на перекресток;
* Пешеходный переход – дорога;
* Пешеход – рядом с пешеходным переходом;
* Движение запрещено – дорога.

Для удаления объекта у пользователя должна быть возможность выбрать объект с помощью щелчка мыши. При выборе объекта он выделяется рамкой, появляются настройки объекта; кнопка «Удалить объект» становится активной. Пользователь должен иметь возможность снять выделение с объекта при нажатии на пустое место в схеме или при переключении на другой объект (пропадает рамка и скрываются настройки объекта).

Пользователь должен иметь возможность поменять характеристики выбранных объектов. У каждого типа объектов есть специализированные характеристики.

Характеристики объектов:

* Пешеход – таймер появления на зебре, реализованный с помощью label и кнопок «+1 сек.» и «-1 сек.» (пешеход проходит зебру всегда за 5 секунд);
* Движение запрещено – транспорт, обозначенный на знаке не имеет право проезжать по дороге с данным знаком. Настройка происходит в виде набора полей выбора, каждый из которых предназначен для определенного вида транспорта. При включении одного из поля остальные выключаються

Созданная программа должна быть сохранена в формате скомпилированного приложения, либо создать скрипт файл (.bat), запускающий приложение; необходимо приложить скриншот интерфейса. Приложение, скриншот и проект необходимо загрузить на платформу контроля версий GitHub, в свой репозиторий. Репозиторий назовите своей фамилией на латинице. Инструкция по загрузке находится в репозитории с ресурсами.

Модуль 2 Разработка системы управления (4 часа)

В данном модуле Вам предстоит создавать систему управления, включающую в себя методы по управлению светофорами в ручном и автоматическом режимах. Предусмотреть правила поведения транспортных средств.

Задание:

Создайте кнопку «Режим по времени» и реализуйте функцию по настройке светофоров так что бы они переключались между зеленым и красным цветом раз в 5 секунд.

Создайте кнопку «Режим по транспорту» и реализуйте функцию по настройке светофоров так что бы они переключались на зеленый, когда к ним приближается транспорт (транспорт находится в соседней ячейке).

Создайте label, который отображает текущий режим работы светофоров.

Реализуйте элементы интерфейса для изменения цвета светофора, при нажатии на него и выбора нажатием на соответствующую кнопку одного из двух предложенных цветов или перевода в стандартный режим. Цвет светофора сохраняется на весь промежуток работы программы или пока не переведен в стандартный режим.

Создать кнопку «Тест-шаблон» (тест системы с шаблонным движением транспорта по рисунку 2) с данным функционалом:

* Транспорт появляется в позиции, согласно заданию и с заданным промежутком времени в 7 секунд;
* Транспорт перемещается согласно определенным в задании маршрутам.

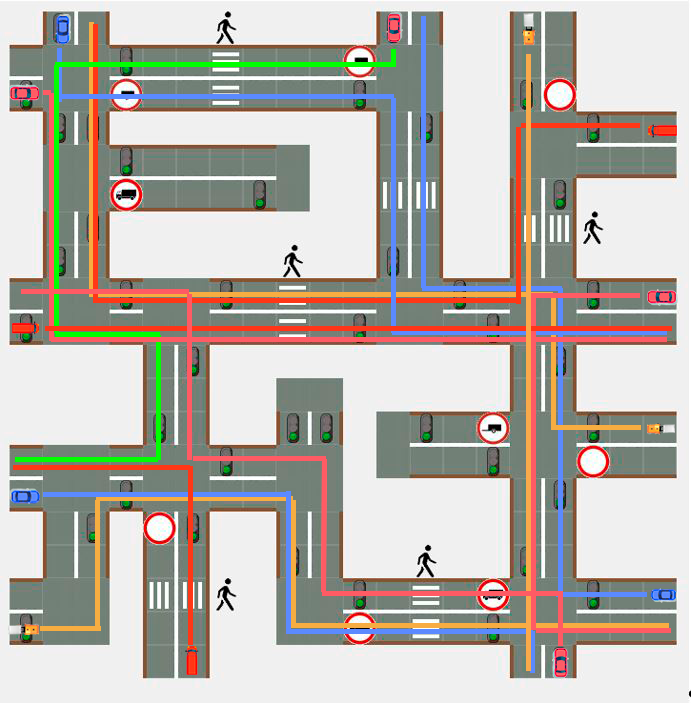


Рисунок 2 – Шаблон движения

|  |  |
| --- | --- |
| C:\Users\sasho\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Cvertical.pngBCbottom | Легковые автомобили |
|  | Грузовик |
|  | Автомобиль с прицепом |

Создать кнопку «Тест-рандом» (тест системы со случайным движением транспорта) с данным функционалом:

* Транспорт появляется со случайным промежутком времени в различных въездах в город;
* Транспорт движется по случайному маршруту, пока не окажется за пределами города.

У движения транспорта есть правила, которые он не должен нарушать. Правила движения касательно транспорта таковы:

* Нельзя выезжать за пределы дороги;
* Нельзя выезжать на закрытые участки дороги;
* Нельзя проезжать, когда на одном из участков зебры есть пешеход;
* Нельзя проезжать сквозь другой транспорт;
* Нельзя проезжать на красный;
* Нельзя выезжать на встречную полосу движения.

Необходимо сделать анимацию движения транспорта (плавное перемещение между клеток дороги), а также реализовать систему поворотных огней, поворотные огни могут быть только передние.

Созданная программа должна быть сохранена в формате скомпилированного приложения, либо создать скрипт файл (.bat), запускающий приложение; необходимо приложить скриншот интерфейса. Приложение или скрипт файл, скриншот и проект необходимо загрузить на платформу контроля версий GitHub, в свой репозиторий. Инструкция по загрузке находится в репозитории с ресурсами.

Модуль 3 Проектирование, создание и подключение базы данных (2 часа)

В данном модуле предстоит спроектировать, создать и подключить базу данных к программе.

Задание:

Спроектировать и создать базу данных сохраняющую информацию о событиях, проходящих в дорожной сети города.

Таблица «Traffic\_light» состоит из следующих полей:

* Номер светофора (Id\_light);
* Количество транспорта, проехавшего за всю работу программы; (Car\_count\_full\_run);
* Среднее количество транспорта, проехавшего через светофор за минуту (Car\_average\_in\_minute);
* Количество переключений цвета светофора (Count\_color\_switches).

Таблица «Cars\_stats» состоит из следующих полей:

* Номер транспорта (Car\_id);
* Количество тиков, прошедших с момента запуска программы до появления транспорта в городе (Car\_spawn\_ticks);
* Количество тиков, прошедших с момента запуска программы до выезда транспорта из города (Car\_exit\_ticks).

После реализации базы данных необходимо ее наполнить данными для проверки. В таблицу «Traffic\_light» данные поступают после проезда машины через светофор. В таблицу «Cars\_stats» данные поступают при въезде и выезде транспорта из города.

Во время запуска одного из тестов, данные в таблицах должны сбрасываться.

Добавьте в интерфейс кнопку «Таблицы», данная кнопка, открывает окно для просмотра таблиц базы данных. Данные из базы данных должны отображаться в табличном виде. Пользователю должно быть доступно отображение обоих таблиц базы данных и их данных по нажатию на соответствующую кнопку с названием как у таблицы.

Созданная программа должна быть сохранена в формате скомпилированного приложения, либо создать скрипт файл (.bat), запускающий приложение. Приложение или скрипт файл, и проект необходимо загрузить на платформу контроля версий GitHub, в свой репозиторий. Инструкция по загрузке находится в репозитории с ресурсами.

Модуль 4 Подключение, настройка и обучение модели искусственного интеллекта (4 часа)

В данном модуле предстоит дополнить графический интерфейс, реализовать систему обучения и взаимодействовать с базой данных.

Задание:

Дополните пользовательский интерфейс, добавив в него кнопки запуска («Старт обучения») и остановки («Остановка обучения») обучения. При остановке обучения транспорт должен возвращаться в начальную позицию, светофоры перестают переключаться, а обучение останавливаться.

В интерфейс должны быть встроены: счетчик номера итерации, таймер работы текущей итерации, таймер времени лучшего прохода, когда все машины завершили движение по своим маршрутам. Также нужно предусмотреть возможность регулировать скорость визуализации обучения.

При старте обучения в lable отображающий режим работы светофоров записывается значение «Интеллектуальный режим».

Движение транспорта происходит по шаблону из модуля 2, только на этот раз машины появляться не бесконечно раз, а ровно 100 машин на итерацию. Машины появляться тогда, когда предыдущая машина, движущаяся по тому же маршруту, завершило движение. Итерация считается завершенной, когда все 100 машин завершили движение.

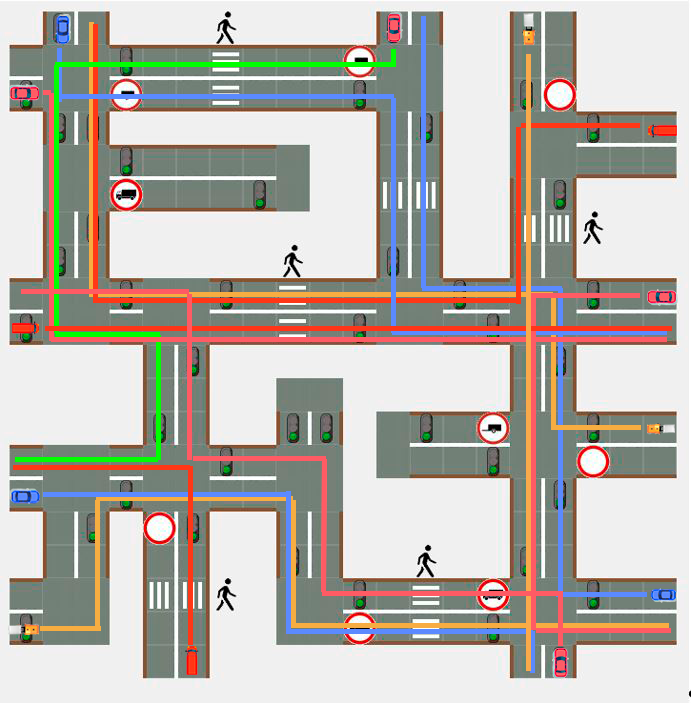


Рисунок 3 – Шаблон движения транспорта из 2 модуля

Реализовать визуализацию работы обучения. Для этого предстоит отобразить движение транспорта, переключение светофоров и других объектов во время обучения и добавить изображения с нейронами (изображения нарисуйте самостоятельно) их описанием и связями между ними. Связи между нейронами должны изменятся по ходу обучения, отражая модель.

При прохождении обучения транспорт должен соблюдать правила движения:

* Нельзя выезжать за пределы дороги;
* Нельзя выезжать на закрытые участки дороги;
* Нельзя проезжать, когда на зебре находится пешеход;
* Нельзя проезжать сквозь курсирующий транспорт;
* Нельзя проезжать на красный;
* Нельзя выполнять повороты на одном месте;
* Нельзя выезжать на встречную полосу движения.

Для сохранения данных обучения вы должны добавить таблицу «Learning\_Records» в базу данных с полями:

* Номер итерации (Iteration\_id);
* Время итерации (Iteration\_time).

Создать функционал, позволяющий вносить данные об обучении в таблицу «Learning\_Records». Данные в таблице должны обновляться после каждого начала обучения.

Созданная программа должна быть сохранена в формате скомпилированного приложения, либо создать скрипт файл (.bat), запускающий приложение; необходимо приложить скриншот интерфейса. Приложение или скрипт файл, скриншот и проект необходимо загрузить на платформу контроля версий GitHub, в свой репозиторий. Инструкция по загрузке находится в репозитории с ресурсами.

Модуль 5 Документирование работы (1 час)

В заключительном модуле необходимо собрать все данные о проделанной работе и оформить презентацию. Для полноты информации о программе необходимо сделать инструкцию по её использованию для пользователя.

Задание:

По итогам выполненной работы необходимо создать презентацию, показывающую заказчику проделанную работу. Презентация должна содержать сравнение между шаблоном и созданной Вами программой, данные полученные во время обучения и описание типа и алгоритма обучения. Название презентации - «Presentation».

В дополнение Вы должны создать инструкцию по использованию программы пользователем. Здесь поясняется пользовательский интерфейс; описываются функции, которыми обладает Ваша программа. Название файла - «Instruction».

Свою презентацию и инструкцию для пользователя необходимо загрузить на платформу контроля версий GitHub, в свой репозиторий.