ФЕДЕРАЛЬНОЕ АГЕНСТВО СВЯЗИ

Ордена Трудового Красного Знамени

федеральное государственное бюджетное

образовательное учреждение высшего образования

«Московский Технический Университет Связи и Информатики»  
(МТУСИ)

Кафедра математической кибернетики и информационных технологий

Задача № 3: Java-Я-звезда!

Выполнила:

Студентка 2 курса

Группы БСТ-1602

Феоктистова Лидия

Москва, 2018

**Задача:**

Дописать программу нахождения оптимального пути между двумя точками при возможности добавления препятствий.

**Анализ предметной области, выбор инструментария:**

Я использовала Java Development Kit (JDK) - комплект разработчика приложений на языке Java, включающий в себя компилятор Java (javac), а для написания программы - программу Notepad++.

**Объяснение функций:**

1. class AStarState

Метод **public int numOpenWaypoints()** возвращает количество точек в наборе открытых точек;

Метод **public Waypoint getMinOpenWaypoint()** проверяет все точки в наборе открытых точек и возвращает ссылку на точку с наименьшей общей стоимостью. Если нет точки в "открытом" наборе, возвращает NULL.;

Метод **public boolean addOpenWaypoint(Waypoint newWP)** если новая точка представляет собой более короткий путь к этому месту, чем текущий маршрут, то заменяет текущую точку на новую, возвращает true, если новая точка была добавлена в набор открытых точек, или false, если новые точки не добавились;

Метод **public boolean isLocationClosed(Location loc)** возвращает true, если указанное

расположение находится в наборе закрытых точек или false иначе;

Метод **public void closeWaypoint(Location loc)** берет точку и перемещает её из набора «открытых точек» к набору «закрытых точек»;

1. class Location

Метод **public boolean equals(Object obj)** возвращает true, если объекты равны или false иначе;

Метод **public int hashCode()** использует код открытых и закрытых точек как hash maps;

**Выводы:**

Я написала программу, позволяющую находить минимальный путь от одной точки к другой, обходя препятствия.