

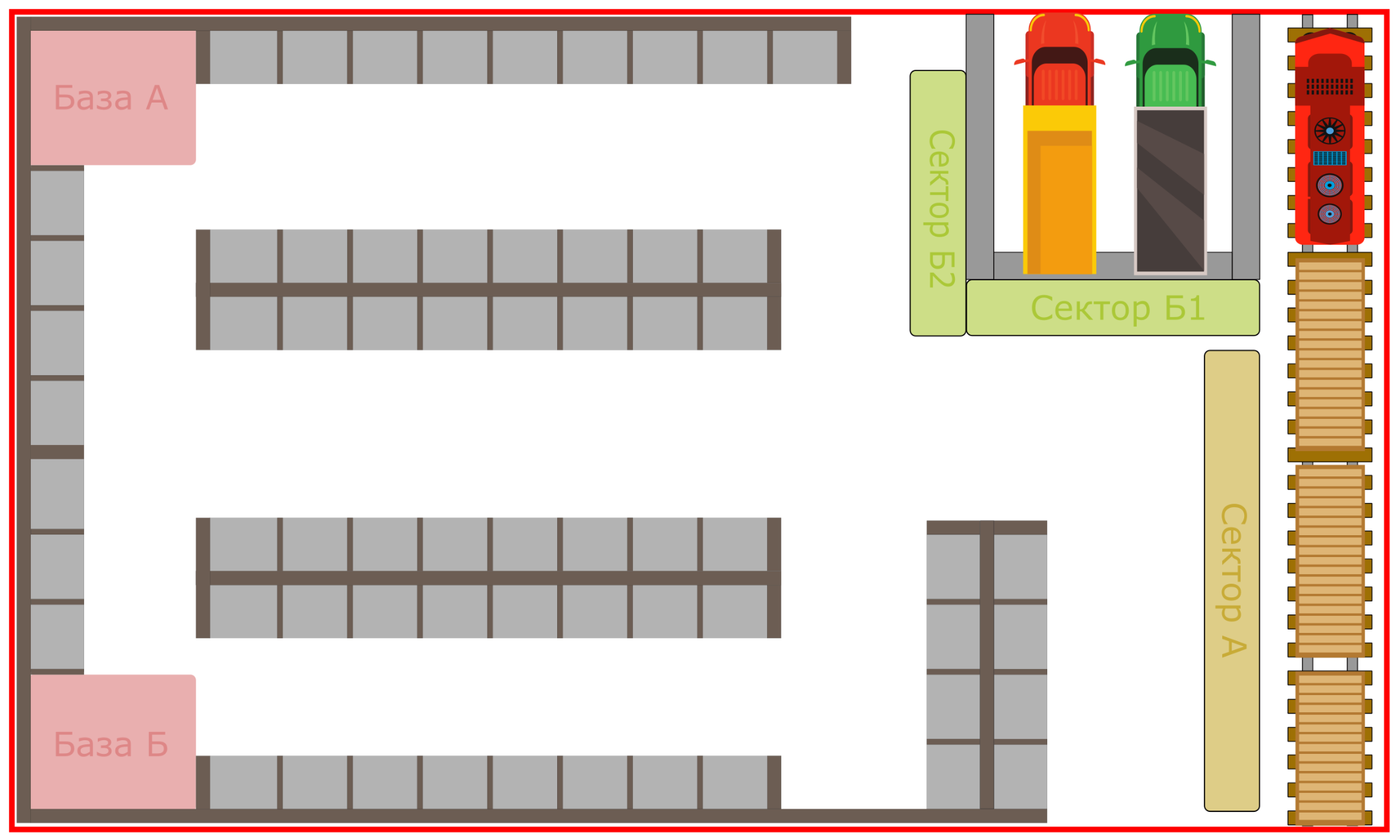
**Задачи финального этапа  
Национальной технологической олимпиады по профилю  
*«Интеллектуальные робототехнические системы»***

### 

Изменение приоритетов во внешней торговле привело к существенному изменению логистики товарных потоков. Склад малой мощности оказался востребован как хаб хранения и распределения большого объема потребительских товаров, для чего необходимо повысить производительность. Одно из решений такой задачи - автоматизация процессов ПРР (погрузочно-разгрузочных работ).

### Основное поле проведения соревнований

Поле высотой 3 метра и шириной 5 метров представляет схему площадки с шестью зонами хранения, тремя зонами погрузки/разгрузки (Сектор Б1, Б2, А). Секторы Б1 и Б2 предназначены для работы с фурами, Сектор А для поездов. База А и Б может используется для старта роботов задаче погрузки и разгрузки. Поле рассчитано на работу 3 - 5 роботов в один промежуток времени, проведения тестов алгоритмов навигации, компьютерного зрения и управления, а также для проведения раундов с оцениванием. Для проведения тестов участникам выделяются слоты времени, на которые капитаны должны записывать каждое утро в течение всех дней олимпиады.

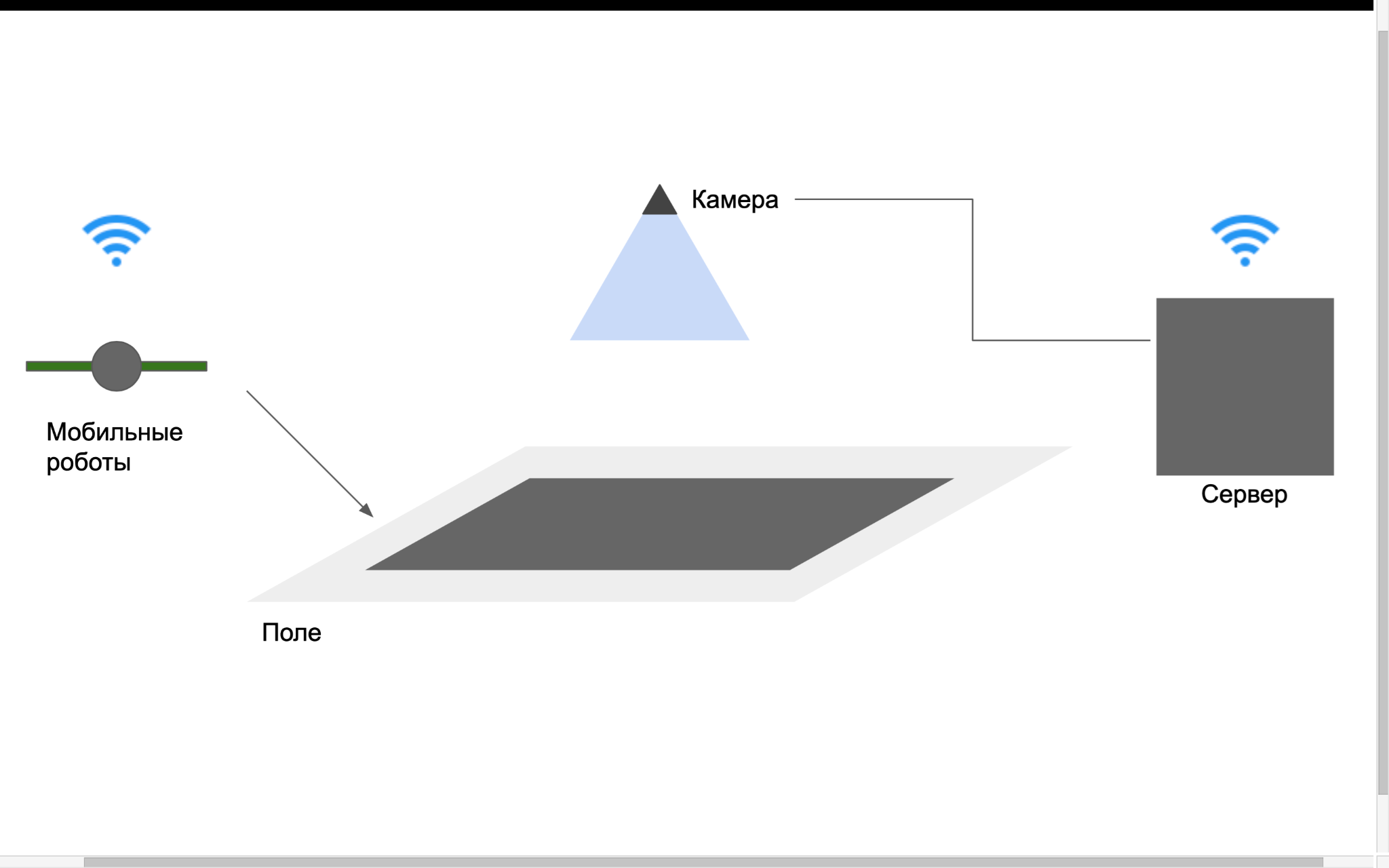


### Тестовое поле проведения соревнований

Поле 2 на 2 метра, присутствуют все те же зоны, что и на основном, размеры ячеек хранения идентичны. На рисунке ниже можно увидеть 4 зоны хранения, две зоны погрузки/разгрузки (Сектора А и Б1) и Базу А для старта движения роботов. Одновременно площадка вмещает в себя 2 - 3 робота. Тестовое поле выделяется по одной штуке на 3 команды. Предназначено для непрерывного тестирования алгоритмов навигации, компьютерного зрения и управления.

### 

### Оснащение основного поля



### Задание 1. Компьютерное зрение

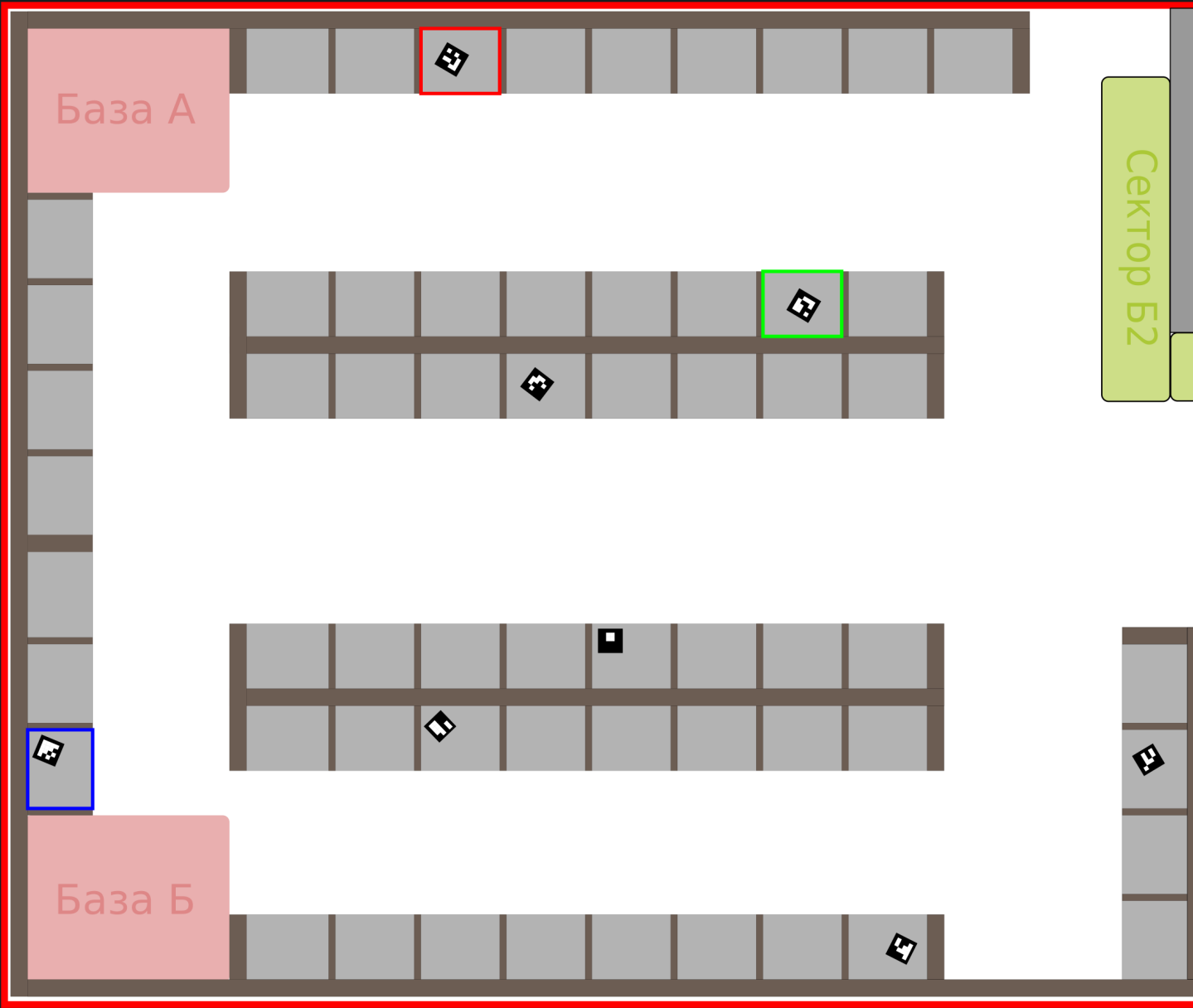
**Баллы за задачу:** 9

**Особенности начисления баллов:** максимальный балл среди всех тестов

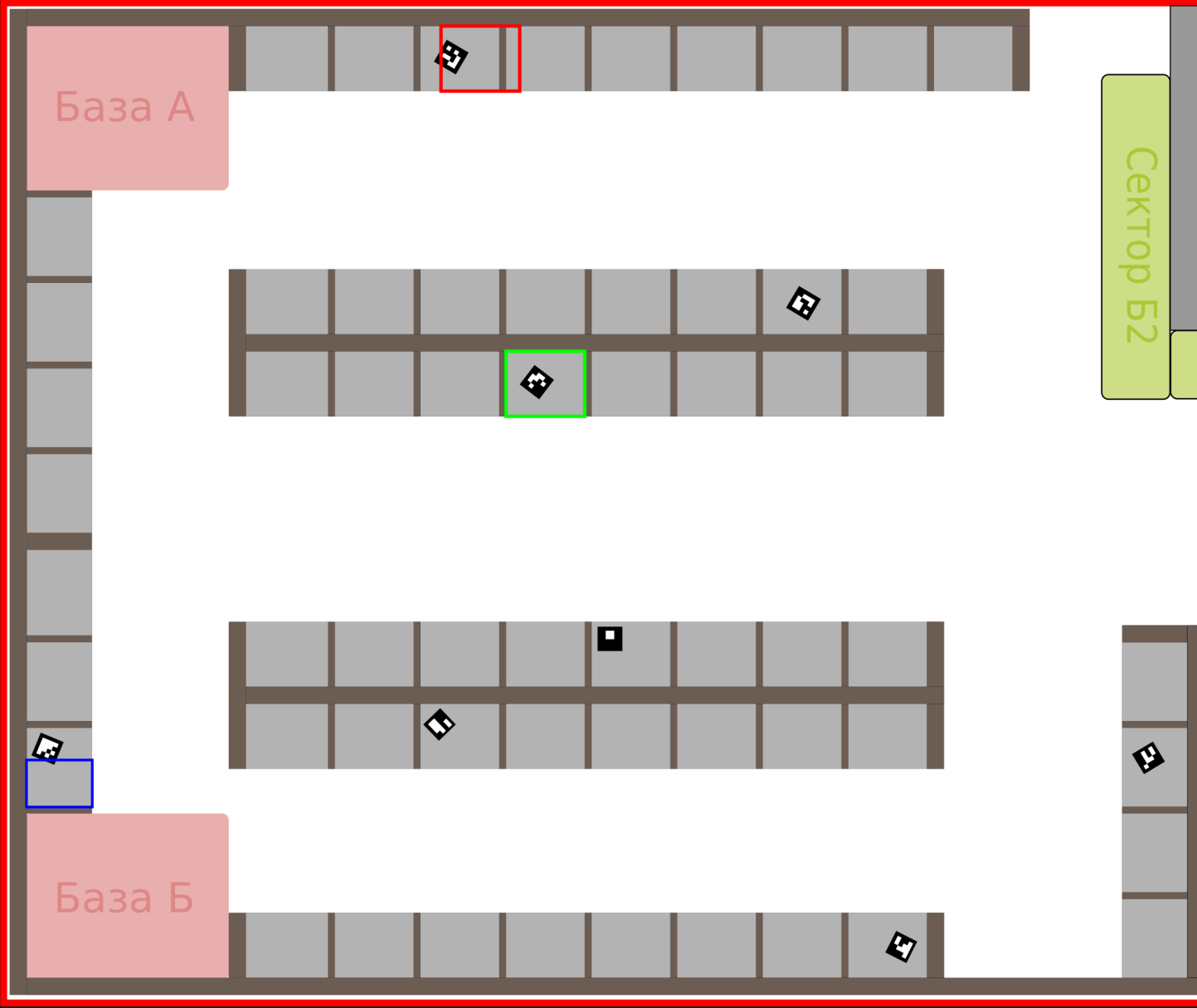
**Описание**

1. **Определение ячейки хранения товара**

Команде называются три случайных номера МТР, расположенных на поле. Капитан команды последовательно вводит три озвученных номера и нажимает “Ввод”. Программа должна обвести ячейки на изображении поля, в которых находятся МТР в рамки синего, зеленого и красного цвета. Синяя рамка ячейки соответствует максимальному из названных номеров МТР, красная рамка – минимальному из названных номеров МТР. За каждую верно выделенную ячейку команда получает **1 балл**. **Максимум 3 балла**.



*Пример верно выполненного задания (3 балла) для следующих входных данных: 5 4 8*



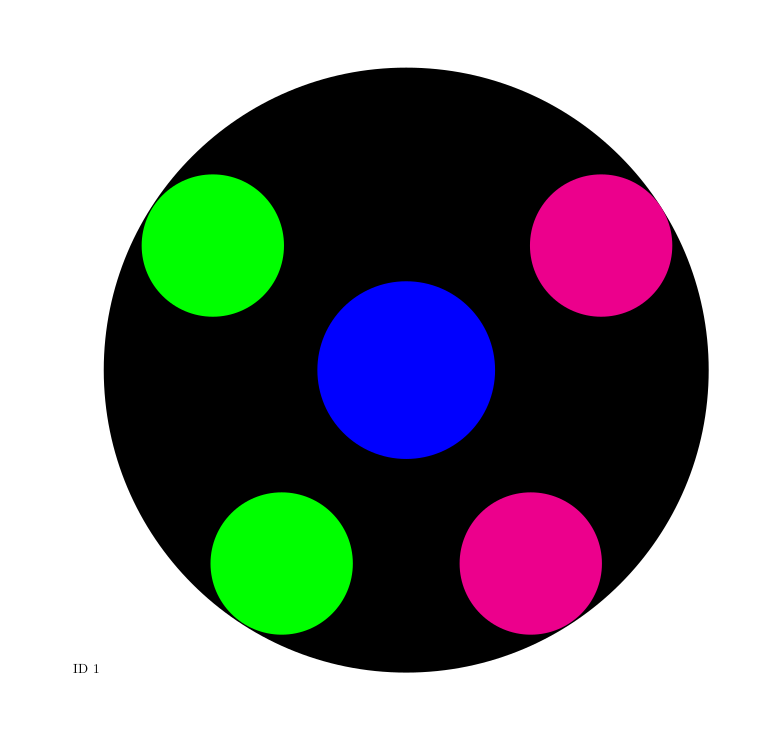
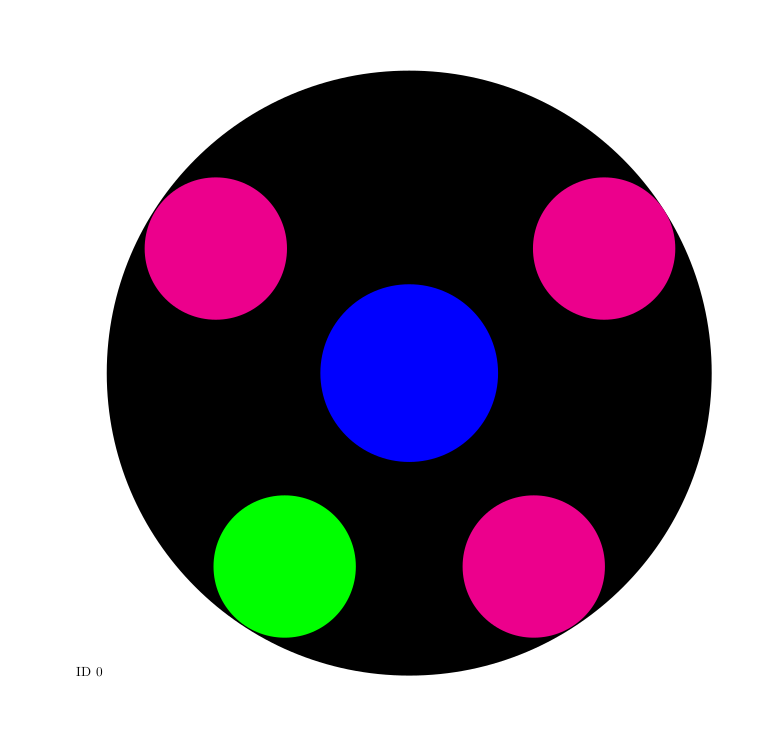
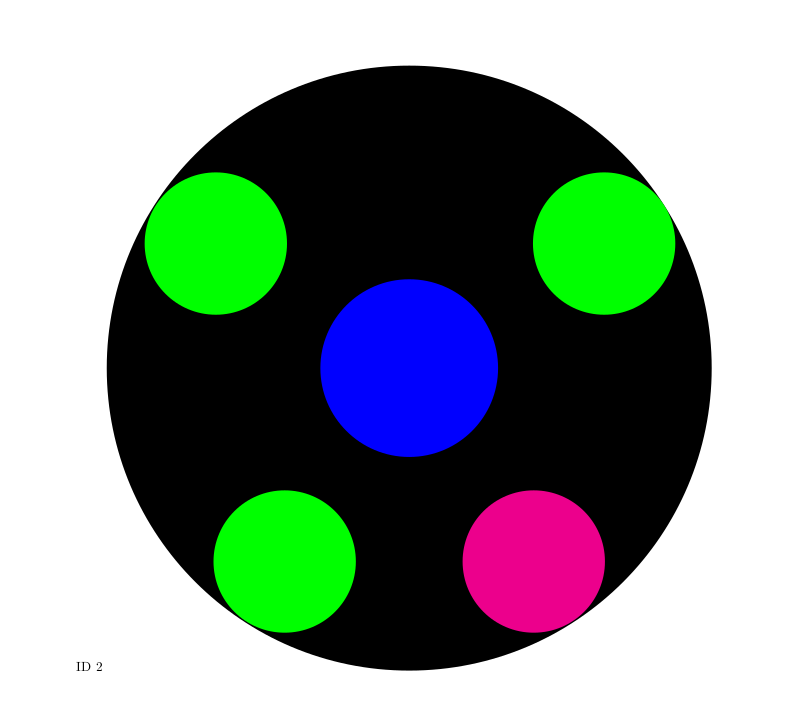
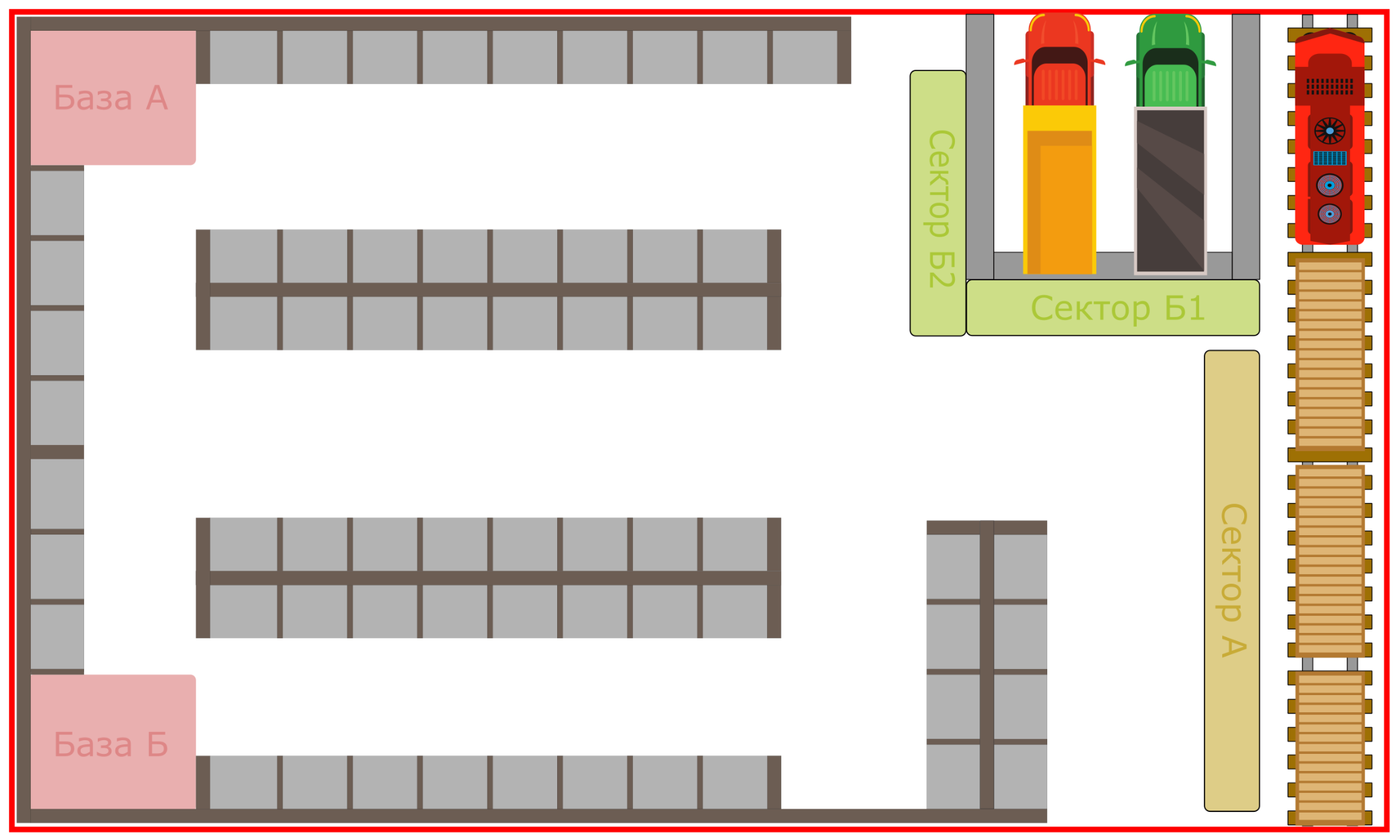
*Пример неверно выполненного задания (0 баллов) для следующих входных данных: 5 4 8*

1. **Определение координат роботов**

На поле в случайных местах устанавливаются 3 робота с произвольной ориентацией. Необходимо определить положения роботов на карте в пиксельной системе координат. При нажатии клавиши “Ввод” капитаном команды, программа должна показать изображении с камеры, где красной линией выделены края маркера робота и рядом написан его id. Линия должна четко проходить по краю, если она явно смещена, ответ не засчитывается. За каждое верно определенный маркер (линия + id) начисляется **1 балл**. **Максимум 3 балла**.

*Пример выходных данных программы:*

| 123 548  376 984  782 451 |
| --- |

Пример правильного ответа

1. **Построение карты занятости ячеек склада (индикация занятых ячеек)**

Ячейки склада случайным образом заполняются МТР. Команде необходимо продемонстрировать работоспособность программы, которая основываясь на изображении полученной с камеры определит какие ячейки заняты, а какие свободны. При нажатии клавиши “Ввод” программа должна выделить зеленым цветом незанятые ячейки и красным цветом занятые. При отсутствии ошибок команда получает **3 балла**. За каждую ошибку у команды снимается **1 балл**. Минимальное количество баллов: 0 баллов.

### Задание 2. Управление роботом

**Баллы за задачу:** 6

**Время одной подзадачи:** 3 минуты

**Особенности начисления баллов:** максимальный балл среди всех тестов

**Подзадачи**

1. **Визуальное управление**

Команда должна написать программу для управления роботом на основе обратной связи по изображению.

При нажатии клавиши “Ввод” программа должна выполнять следующие действия:

1. Открыть окно с видео-трансляцией поля и пометить положение робота на поле.
2. При нажатии судьей курсором мыши по изображению, программа должна, используя удаленное управление роботом посредством bluetooth, заставить робота стабилизироваться в заданной точке. Точка выбирается в соответствии с требованием об отсутствии возможных препятствий на пути и при условии ее достижимости.

Судья последовательно, ожидая достижения целей и полной остановки робота, задает 3 цели для достижения роботом. За каждую достигнутую роботом цель команда получает **1 балл. Максимум 3 балла.**

1. **Захват объекта; выполнение простых команд с объектом**

Команда должна написать программу для захвата МТР из зоны хранения. Робот устанавливается за 10 см до предполагаемой ячейки хранения с МТР.

При нажатии клавиши “Ввод” в программе робот, управляемый посредством bluetooth должен выполнить следующие действия:

1. Подъехать к ячейке с МТР
2. Захватить МТР
3. Отъехать к стартовому положению

За каждое совершенное действие команда получает по **1 баллу**. **Максимум 3 балла**.

### Задание 3. Навигация

**Баллы за задачу:** 6

**Особенности начисления баллов:** максимальный балл среди всех тестов

**Подзадачи**

1. **Позиционирование робота в заданной координате**

Робот устанавливается в случайной свободной (белой) клетке поля.

При нажатии клавиши “Ввод” программа должна выполнять следующие действия:

1. Открыть окно с видео-трансляцией поля и пометить положение робота на поле аналогичным образом, как в Задаче 1. Компьютерное зрени **Определение координат роботов.**
2. При нажатии судьей курсором мыши по изображению, программа должна выделить красным цветом цель, спланировать движение и, используя удаленное управление роботом, посредством bluetooth, заставить робота следовать по маршруту с заданной точке. Центр робота должен оказаться в центре целевой точки. Точка выбирается в соответствии с требованием ее достижимости.

Судья последовательно, ожидая достижения целей и полной остановки робота, задает 3 цели для достижения роботом. За каждую достигнутую роботом цель команда получает **1 балл.**

1. **Планирование безопасного движения группы роботов** (2 робота, без столкновений)

Команда должна разработать программу, которая основываясь на показаниях камеры позволит безопасно управлять группой роботов (2 робота). Роботы устанавливаются в случайных свободных (белых) клетках поля. При нажатии клавиши “Ввод” программа должна выполнять следующие действия:

1. Открыть окно с видео-трансляцией поля и пометить положение роботов на поле аналогичным образом, как в Задаче 1. Компьютерное зрени **Определение координат роботов.**
2. При нажатии судьей курсором мыши по изображению, программа должна запомнить и выделить первую цель. При повторном нажатии в другом месте программа должна выделить вторую цель.
3. Используя удаленное управление роботом посредством bluetooth, имея информацию о карте и положении роботов спланировать движение роботов к точке с избеганием столкновений с другими роботами.
4. Роботы, под управлением программы должны выполнить следование по найденному маршруту, избегая столкновений с препятствием и друг с другом. Робот, с младшим id маркера, должен приехать в первую заданную судьей точку, а робот со старшим id маркера во вторую. Точки выбираются в соответствии с требованием ее достижимости.

При отсутствии ошибок команда получает **3 балла**. За каждую ошибку, столкновение или не достижение цели у команды снимается **1 балл**. Минимальное количество баллов: 0 баллов.

### 

### Задание 4. Погрузка и разгрузка (Задача инженерного тура)

**Баллы за задачу:** ∞

**Особенности начисления баллов:** начисление баллов пропорционально пройденным тестам

**Условие**

Участникам необходимо реализовать программу для управления группой из 3 - 5 роботов, которые выполняют ПРР из зоны хранения в зону погрузки/разгрузки (Сектор А или Б) и наоборот.

Перед началом задания проводится инициализация системы:

1. Роботы размещаются на поле с произвольным положением и ориентацией
2. В ячейки зоны хранения устанавливаются МТР в соответствии со списком для раунда
3. В свободной для передвижения зоне склада устанавливаются статичные препятствия
4. Чтением сообщений системы по протоколу TCP/IP выдается лист заказов (в виде строки) на ПРР для команды. Лист содержит информацию о МТР и о зоне доставки. Здесь  
   - task\_id – идентификатор задачи,  
   - cargo\_id – идентификатор МТР, который нужно перевезти,  
   - sector – значение сектора погрузки/разгрузки (есть сектор А и Б)  
   - action – тип действия, отвезти в зону погрузки/разгрузки to и из нее from.

*Пример:*

| [  {'task\_id': 101, 'cargo\_id': 3, 'sector': 'A', 'action': 'to'},  {'task\_id': 102, 'cargo\_id': 4, 'sector': 'A', 'action': 'to'},  {'task\_id': 103, 'cargo\_id': 4, 'sector': 'B', 'action': 'to'},  {'task\_id': 104, 'cargo\_id': 9, 'sector': 'A', 'action': 'from'},  {'task\_id': 105, 'cargo\_id': 5, 'sector': 'B', 'action': 'to'},  {'task\_id': 106, 'cargo\_id': 6, 'sector': 'B', 'action': 'to'},  {'task\_id': 107, 'cargo\_id': 1, 'sector': 'A', 'action': 'from'},  {'task\_id': 108, 'cargo\_id': 3, 'sector': 'B', 'action': 'to'}  ] |
| --- |

1. На подготовку к старту команде дается 5 минут.

Для успешного выполнения заказа, робот должен:

1. Отправить в систему сообщение (по протоколу TCP/IP) со строкой о начале выполнении задания, например, “START 2131”, где после слова START указывается id задания
2. Отправиться к ячейке с МТР с заданным cargo\_id, избегая столкновений с объектами
3. Сразу как участник подъехал к объекту и намерен начать захват он должен отправить в систему сообщение (по протоколу TCP/IP) со словарем, например:

| {  task\_id: 2131,  action: "reached"  } |
| --- |

1. Захватить объект с заданным cargo\_id и отправить в систему сообщение (по протоколу TCP/IP) со словарем, например:

| {  task\_id: 2131,  action: "grabbed"  } |
| --- |

1. Доставить объект в зону назначения (это может быть как любая зона хранения на выбор участника в случае 'action': 'from', так и заданный сектор А или Б в случае 'action': 'to'. Аналогично отправить в систему сообщение (по протоколу TCP/IP) со словарем, например:

| {  task\_id: 2131,  action: "delivered"  } |
| --- |

1. Положить объект в корректную зону и отправить в систему сообщение (по протоколу TCP/IP) со словарем, например:

| {  task\_id: 2131,  action: "put"  } |
| --- |

1. После того, как робот выполнил задание, участник должен отправить сообщение, например, “END 2131”, где после слова END указывается id задания

Важно соблюдать порядок выполнения заданий. Нельзя пропускать какое-то задание, то есть, например, нельзя начать выполнение задания 103, если не начато задание 102. Также, нельзя начать выполнять заданий больше, чем имеется роботов. То есть, например, если есть 3 робота, нельзя сообщить системе, что выполняется больше 3 заданий.

Для общения с системой по протоколу TCP/IP на языке Python, рекомендуется использовать следующие команды:

* recv() - для чтения сообщений системы (для получения задания)
* sendall() - для отправки сообщений системе, важно, что данный метод принимает только объекты в битовом представлении

**Пример программы для общения с системой:**

| import socket  import sys  import json  import time  # Create a TCP/IP socket  sock = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)  # Connect the socket to the port where the server is listening  server\_address = ('localhost', 10000)  # Messages examples  msgs = ["START 101", "START 102", {"task\_id":102, "action":"reached"}, {"task\_id":102, "action":"reached"}, {"task\_id":102, "action":"delivered"}, {"task\_id":102, "action":"put"}, "END 102"]  sock.connect(server\_address)  try:  # get message from server  data = (sock.recv(1024)).decode()  # print task from refbox  print('received', data)  time.sleep(1)  for message in msgs:  # Send data  print('sending', message)  if isinstance(message, str):  # if string  sock.sendall(message.encode())  else:  # if dict  user\_encode\_data = json.dumps(message, indent=2).encode()  sock.sendall(user\_encode\_data)    time.sleep(1)  finally:  print('closing socket')  sock.close() |
| --- |

**Ссылка на репозиторий с системой:** <https://gitlab.com/StriderOne/nto_refbox.git>

Участник получает баллы за:

* приехать вплотную к целевому МТР – 1 балл
* осуществить захват целевого МТР – 1 балл
* отвезти МТР в пункт назначения – 1 балл
* освободить МТР в зоне назначения – 1 балл

Участник теряет баллы за:

* легкое столкновение с препятствие – 1 балл
* наезда на линию разметки границы поля – 1 балл

Участник получает 0 баллов и задача перезапускается:

* сильное столкновение (предмет сдвинулся)
* выезд за пределы поля
* наезд на линии разметки стен

*Результат*: по итогу выполнения финального задания участники должны написать систему из двух программ (для робота и серверного ПК), способную:

1. С помощью методов компьютерного зрения определять основные объекты на складе: робота, МТР, препятствия, стены и ячейки зоны хранения
2. Строить карту по результатам пункта 1
3. Строить путь по карте из пункта 2 для нескольких роботов
4. Находить кратчайший путь для нескольких роботов
5. Управлять роботом по заданной траектории
6. Управлять роботом для захвата и освобождения МТР
7. Выполнять коммуникацию между роботами и серверным ПК