RCJ Rescue Simulation

Даниил Анисимов, Евгений Шандаров
ТУСУР, Томск

Лига RCJ Rescue Simulation

В данной лиге участникам должны разработать программу для робота спасателя, чтобы исследовать лабиринт, использовать методы обнаружения для поиска “жертв”, избегая при этом препятствия и ловушки.

Описание трассы

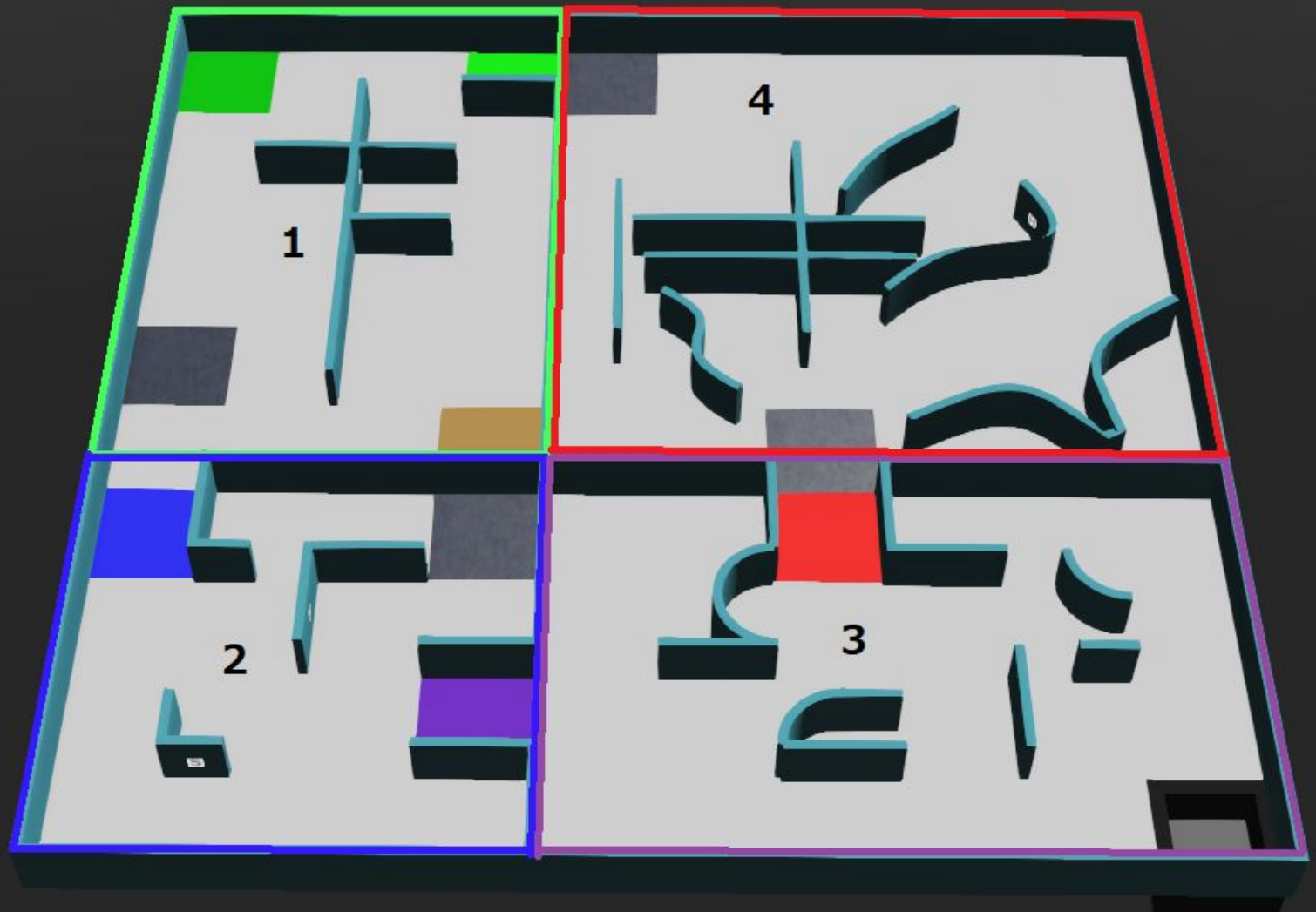
Трасса соревнований это лабиринт состоящий из 4 зон.

1-3 зоны полностью состоят из равных плиток и робот начинает свою работу на одной из этих 3 зон.

4 зона не основана на плитках и является дополнительной, стены в данной зоне могут быть не только закруглены, но и могут быть разной длины и стоять под разным углом.

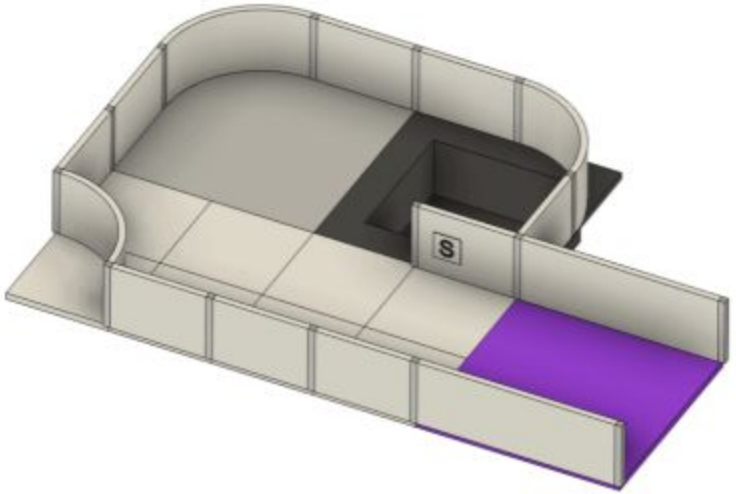
Переходы между зонами обозначается плитками разного цвета:

- 1-2 зона - Синяя плитка
- 2-3 зона - Фиолетовая плитка
- 3-4 зона - Красная плитка
- 4-1 зона - Зеленая плитка
- контрольная точка - Серая плитка


1
4
2
3

Бонус карты

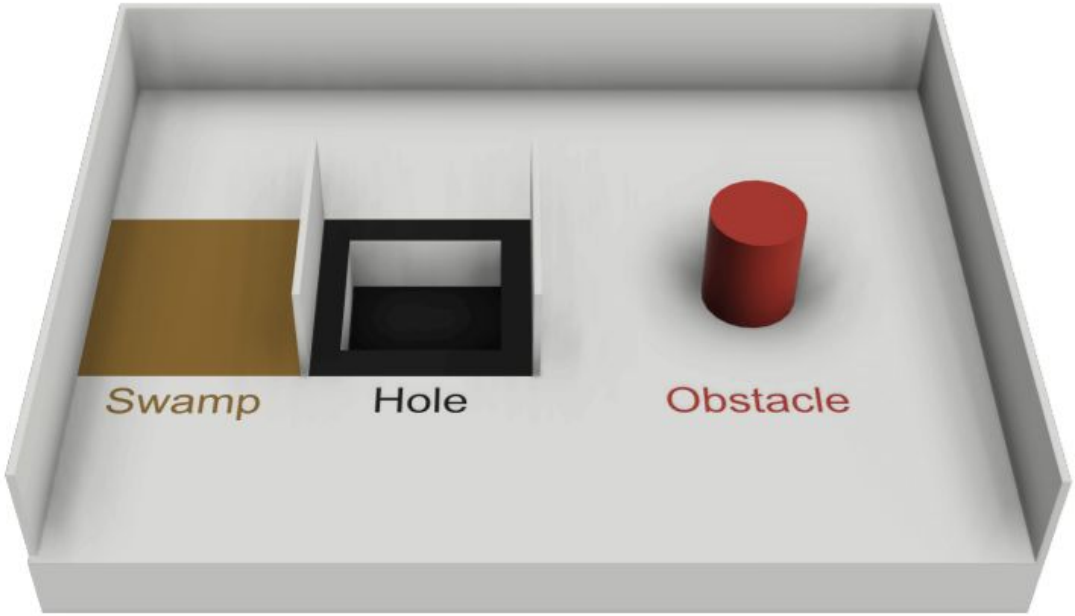
- Стены отмечены цифрой «1»; отверстия как «2»; болота как «3»; контрольно-пропускные пункты как «4»; начальная плитка как «5»; соединительные плитки с 1 по 2 как «6», с 2 по 3 как «7», с 3 по 4 как «8» и с 1 по 4 как «9»; жертвы в качестве соответствующего кода жертвы (H,S,U,F,P,C,O), а любые другие плитки/ребра/вершины должны быть равны «0».
- Для изогнутых стен в области 3 вершина должна быть представлена «0».
- Присутствие пострадавшего должно быть отмечено на ячейке, выражающей соответствующую стенку. Организаторы должны объединить запись, если на стене более одной жертвы



$$\rightleftarrows \quad \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 4 & 0 & 4 & 0 & 2 & 0 & 2 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 4 & 0 & 4 & 0 & 2 & 0 & 2 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & S & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 7 & 0 & 7 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 7 & 0 & 7 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Виды препятствий

- Болота (swamp)
- Препятствия (Obstacle)
- Ямы (hole)



Жертвы

Жертвы представлены изображением размером 2 см на 2 см, размещенным в любом месте на стенах

- Пострадавшая жертва: H
- Стабильная жертва: S
- Невредимая жертва: U

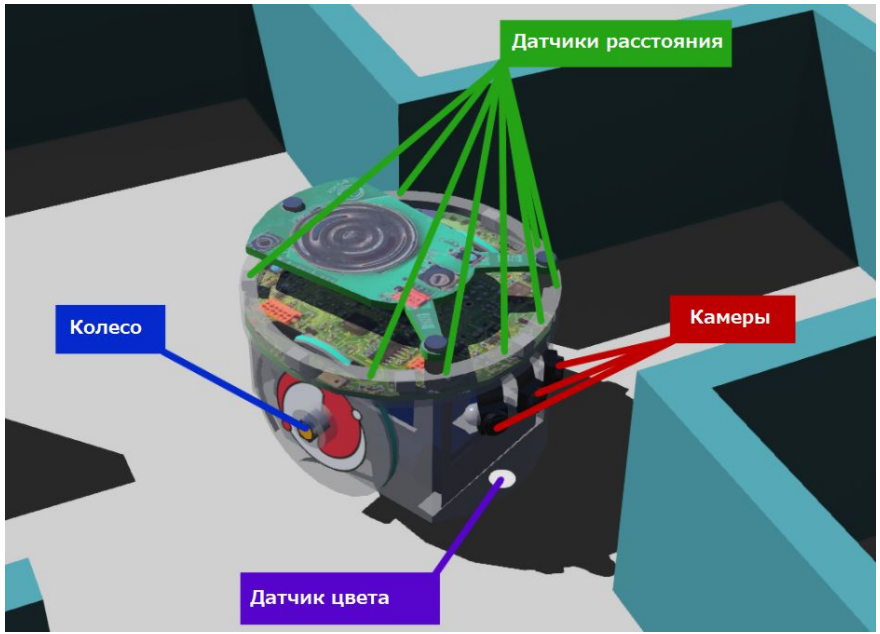
Знаки опасности

Знаки опасности также представлены изображениям 2 см на 2 см, размещенным в любом месте на стенах:

- Воспламеняющийся газ [F]
- Яд [P]
- Коррозионный [C]
- Органический пероксид [O]

Конструкция робота

- 8 датчиков дистанции
  - 2 справа
  - 2 слева
  - 4 спереди
- 3 камеры на передней части робота
- 2 колеса, приводы с энкодерами
- датчик цвета в нижней передней части робота



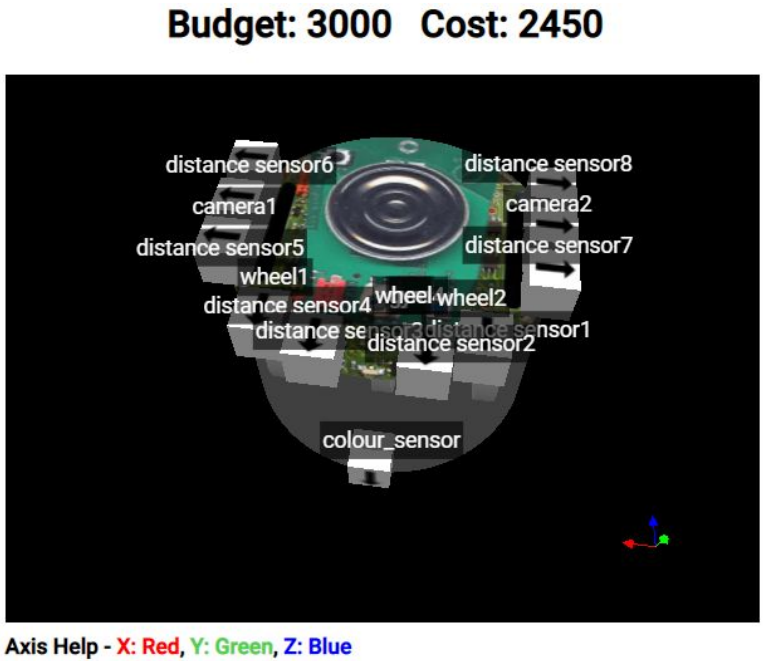
Создание робота

Для более эффективного выполнения задач лиги участники могут построить своего собственного робота, используя доступные модули.

При этом необходимо уложиться в бюджет - 3000 единиц условной валюты. (Цена изначального робота - 2850)

Доступные модули и их цена:

- Гироскоп - 100
- GPS - 250
- Камеры - 500
- Датчик цвета - 100
- Акселерометр - 100
- Лидарный датчик - 500
- Датчики расстояния - 50
- Колеса - 300



Программа

Программа для робота создается на одном из трех предлагаемых языков программирования:

- C
- C++
- Python

Написание программы состоит из 3 основных пунктов:

1) Инициализация модуля робота (датчики, камеры, колесы)
2) Включение модуля робота
3) Создание алгоритм движения робота

```python
############################### Инициализация датчиков
gps = robot.getDevice("gps") # инициализация GPS модуля

colour_camera = robot.getCamera("colour_sensor") # инициализация датчика цвета

distanceSensors = [] #создаем пустой список для датчиков расстояния
for i in range(8):
    distanceSensors.append(robot.getDevice("ps" + str(i)))   #инициализация датчика расстояния x8

leftMotor = robot.getDevice("wheel1 motor")    # инициализация двигателей
rightMotor = robot.getDevice("wheel2 motor")

leftEncoder = leftMotor.getPositionSensor()    # инициализация инкодеров
rightEncoder = rightMotor.getPositionSensor()

############################### Включение датчиков

timestep = int(robot.getBasicTimeStep()) # устанавливаем время обработки

gps.enable(timestep) # включение GPS модуля

colour_camera.enable(timeStep) # включение датчика цвета

for i in range(8):
    distanceSensors[i].enable(timestep) # включение датчика расстояния x8

leftMotor.setPosition(float('inf')) # установка позиции колеса
rightMotor.setPosition(float('inf'))

leftEncoder.enable(timeStep)  # включение инкодеров
rightEncoder.enable(timeStep)

###############################
```

Установка

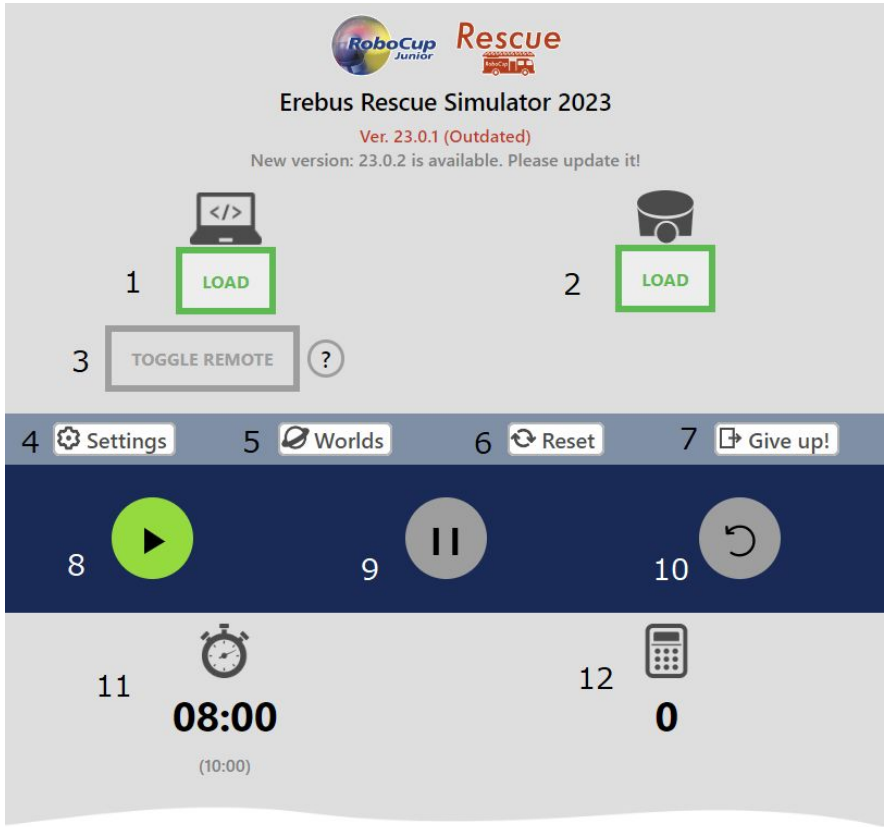
Установка системы моделирования состоит из 3 основных шагов:

1) Загрузить **Python 3.x** (https://www.python.org/downloads/)
2) Загрузить платформу для моделирования **Webots** (последней версии) (https://cyberbotics.com/#download)
3) Загрузить и разархивировать последнюю версию **Erebus** (https://gitlab.com/rcj-rescue-tc/erebus/erebus/-/releases)

Для запуска среды моделирования теперь необходимо запустить файл world.wbt из /game/worlds/world1.wbt, открыв его в Webots.

Контроллер соревнований

1) кнопка необходима для загрузки программы в робота,
2) кнопка необходима для загрузки модели собственного робота,
3) кнопка (Toggle remote) служит для удаленного управления симуляцией,
4) кнопка (Settings) настройки контроллера,
5) кнопка(Worlds) используется для выбора мира в котором будет производиться симуляция,
6) кнопка (Reset) перезапускает контроллер и симуляцию,
7) кнопка (Give up) производит выход из контроллера,
8) кнопка запускающая матч,
9) кнопка приостанавливающая матч,
10) кнопка возвращающая робота на точку сохранения,
11) таймер матча,
12) подсчет очков текущей попытки.



Ресурсы для помощи

Раздел на сайте: http://robocuprussiaopen.ru/about/leagues/RCJ-rescue-simulation/

Регламент (рус):
http://robocuprussiaopen.ru/data/2023/rules/RCJ-Rescue-Simulation-rules-2023-0303.pdf

Регламент (англ.яз):
http://robocuprussiaopen.ru/data/2023/rules/RCJRescueSimulation2023RulesFinal.pdf

Шаблон программы - http://robocuprussiaopen.ru/data/RCJ-rescue-simulation/(PRIMER)_.py

Документация -https://v23.erebus.rcj.cloud/docs/

Спасибо за внимание!

Анисимов Даниил Андреевич
+7 901 137 4701