Задача 2. Программирование в симуляторе (50 баллов)

*Компетенции:* python; компьютерное зрение; регуляторы; MUR IDE

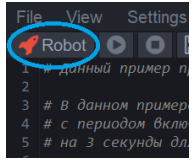
# Подготовка к работе

Для выполнения задания вам необходимо:

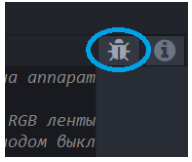
1. Скачать и установить среду разработки MUR IDE для программирования виртуального автономного подводного робота: <https://murproject.com/#download>

2. После установки на рабочем столе должен появиться ярлык MUR IDE. Запустить MUR IDE.

3. Для начала работы в симуляторе перевести режим работы IDE в Local. Для этого следует нажать на кнопку с иконкой ракеты и надписью Remote в левом верхнем углу. Цвет кнопки станет синим и надпись изменится на Local.



4. Запустить симулятор нажатием кнопки с изображением жука в правом верхнем углу. Откроется окно симулятора с черным экраном.



5. Скачать сцену по ссылке. Можно разместить ее в любом удобном каталоге.

6. В запущенном симуляторе перейти в меню Scene → Open. В появившемся диалоговом окне выбрать сцену.

7. Все готово, можно начать программировать виртуальный аппарат на языке программирования Python.

8. В качестве решения задачи необходимо отправить файл с кодом в формате \*.py. Для этого в меню следует выбрать File → File save as.

# Условие

При проведении работ по осмотру дна морских судов и мелкому ремонту все чаще компании прибегают к помощи подводных аппаратов.

Необходимо выполнить работы по диагностике состояния судна, запрограммировав аппарат на обнаружение повреждений и прочих объектов согласно заданию.

## Описание сцены

В сцене представлен корабль, корпус которого окрашен в красный и серый цвет. Для удобства ориентирования вокруг корабля на дне нарисована фиолетовая полоса. Начальное положение аппарата: напротив кормы корабля у винтов.

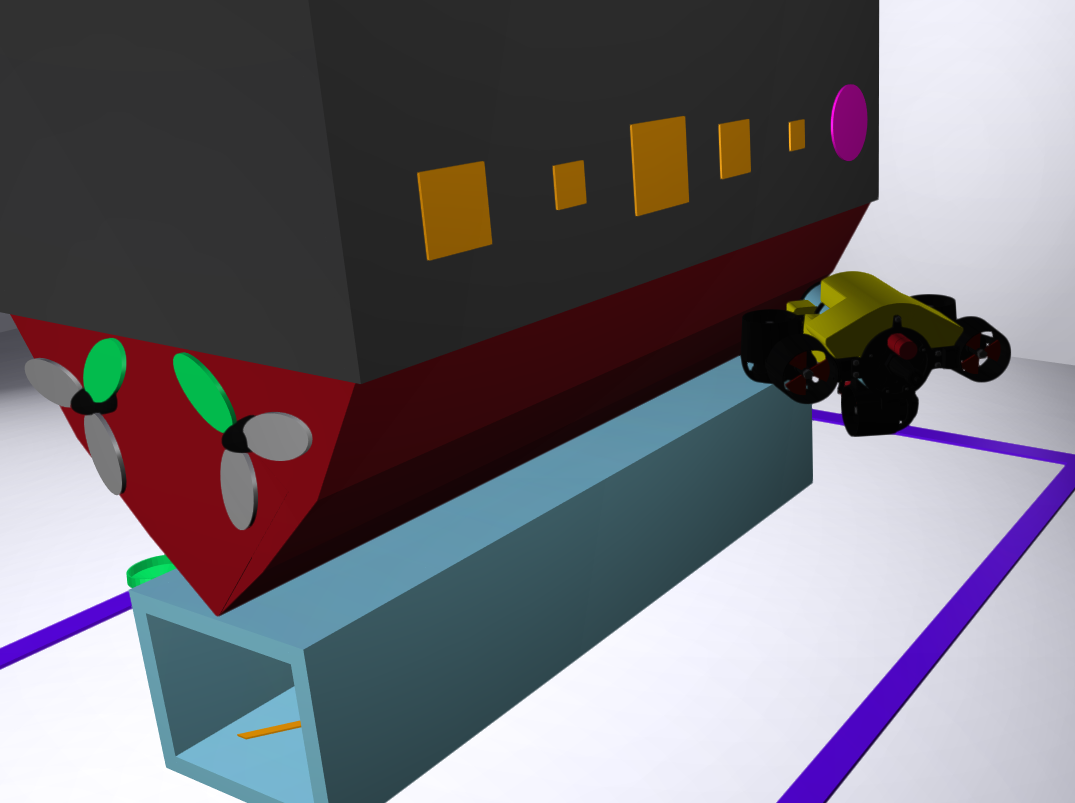


*Начальное положение аппарата*

## Задачи

### **Задача 1. Дефектоскопия - 10 баллов.**

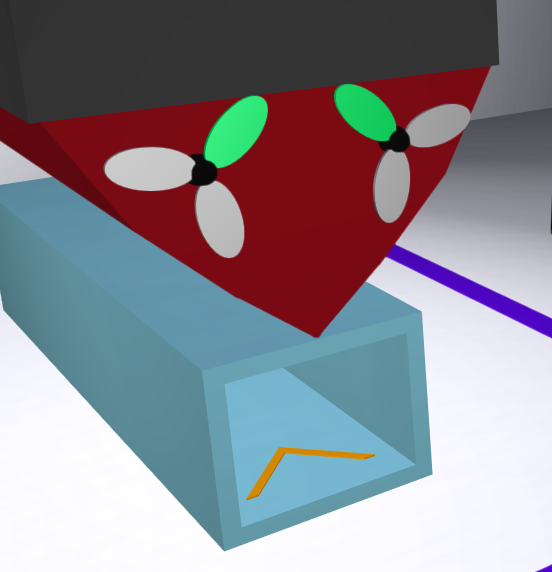
* На правом борту судна расположены повреждения оранжевого цвета. Необходимо определить порядковый номер самого крупного повреждения.
* Порядковый номер повреждения определен означает, что **аппарат выстрелил в розовую мишень (находится в конце ряда повреждений) количеством торпед, равным порядковому номеру мишени.**



*Задача 1. Дефектоскопия*

### **Задача 2.** **Очистка винтов - 10 баллов**

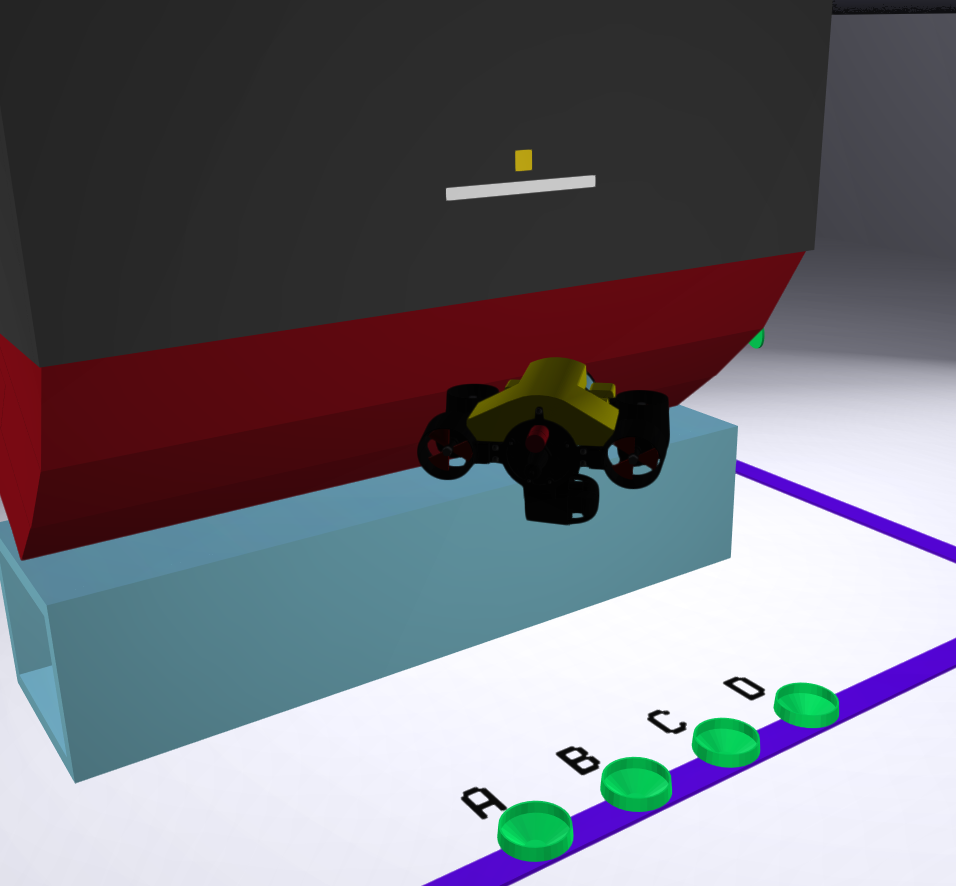
* На корме расположены два рулевых винта серого цвета. Одна из лопастей каждого винта покрылась обрастаниями (такая лопасть будет зеленого цвета).
* Необходимо выполнить очистку винтов от обрастаний.
* Винт очищен означает, что **аппарат коснулся зеленой лопасти своей передней частью (колбой).**
* За касание чистой лопасти (лопасть серого цвета) предусмотрен штраф. Подробнее в разделе “**Критерии оценивания**”.
* При проверке данной задачи при успешном касании зеленой лопасти её цвет сменится на серый.



*Задача 2. Очистка винтов*

### **Задача 3.** **Определение размеров трещины - 15 баллов.**

* На левом борту судна обнаружена трещина белого цвета.
* Над трещиной расположен желтый калибровочный квадрат с известной длиной стороны 1 единица.
* Необходимо выяснить, сколько единиц составляет длина трещины и указать её.



*Задача 3. Определение размеров трещины*

* Для указания длины трещины рядом расположены 4 корзины зеленого цвета. Рядом с каждой корзиной указана буква, обозначающая разряд. Размер необходимо закодировать с применением двоичного кода, сбросив маркеры (таблица кодировки приведена ниже).
* Отсутствие маркера в корзине приравнивается к 0.
* Наличие маркера в корзине приравнивается к 1.
* Длина трещины может меняться от 1 до 9 и всегда является целым числом.

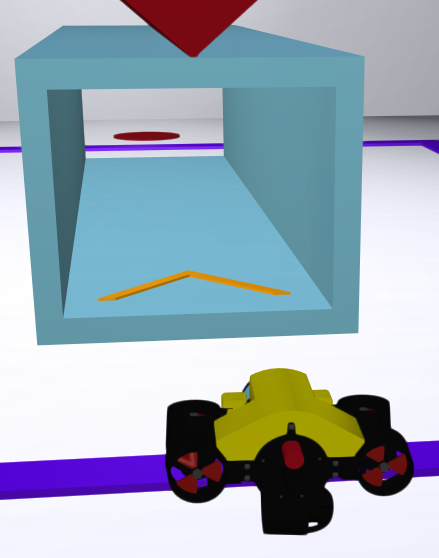
**Таблица кодировки**

| **Длина** | **Корзины** | | | |
| --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **D** |
| **0** | 0 | 0 | 0 | 0 |
| **1** | 0 | 0 | 0 | 1 |
| **2** | 0 | 0 | 1 | 0 |
| **3** | 0 | 0 | 1 | 1 |
| **4** | 0 | 1 | 0 | 0 |
| **5** | 0 | 1 | 0 | 1 |
| **6** | 0 | 1 | 1 | 0 |
| **7** | 0 | 1 | 1 | 1 |
| **8** | 1 | 0 | 0 | 0 |
| **9** | 1 | 0 | 0 | 1 |

### **Задача 4.** **Исследование корпуса -** **10 баллов.**

Необходимо исследовать дно корабля под килем. Задание выполнено верно, если аппарат зашел со стороны винтов и вышел со стороны носовой части.

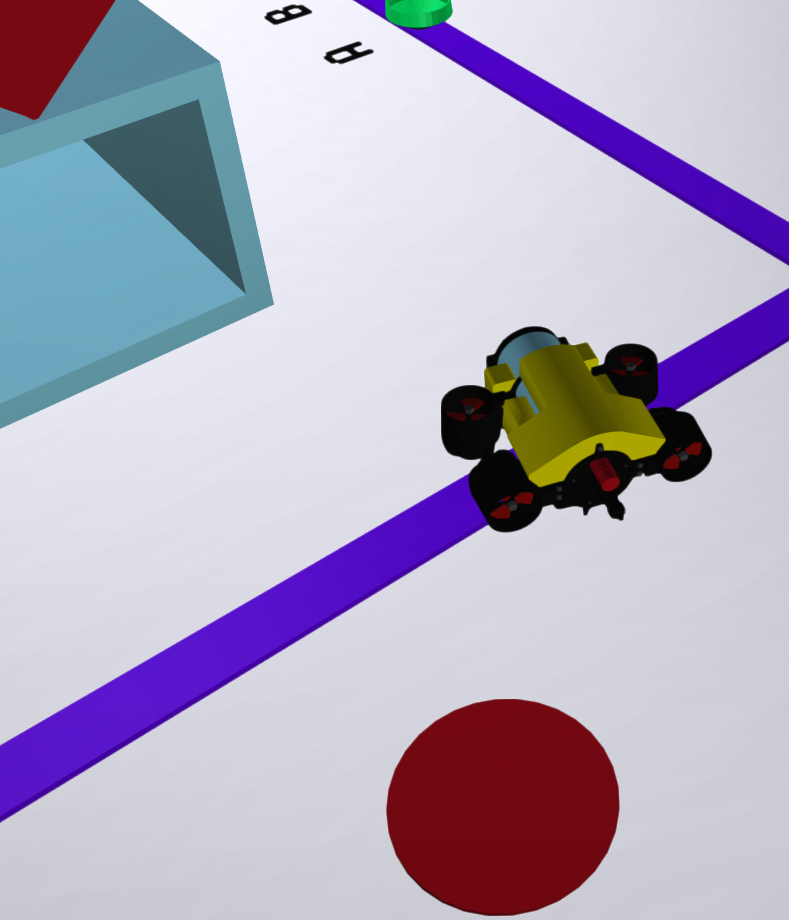
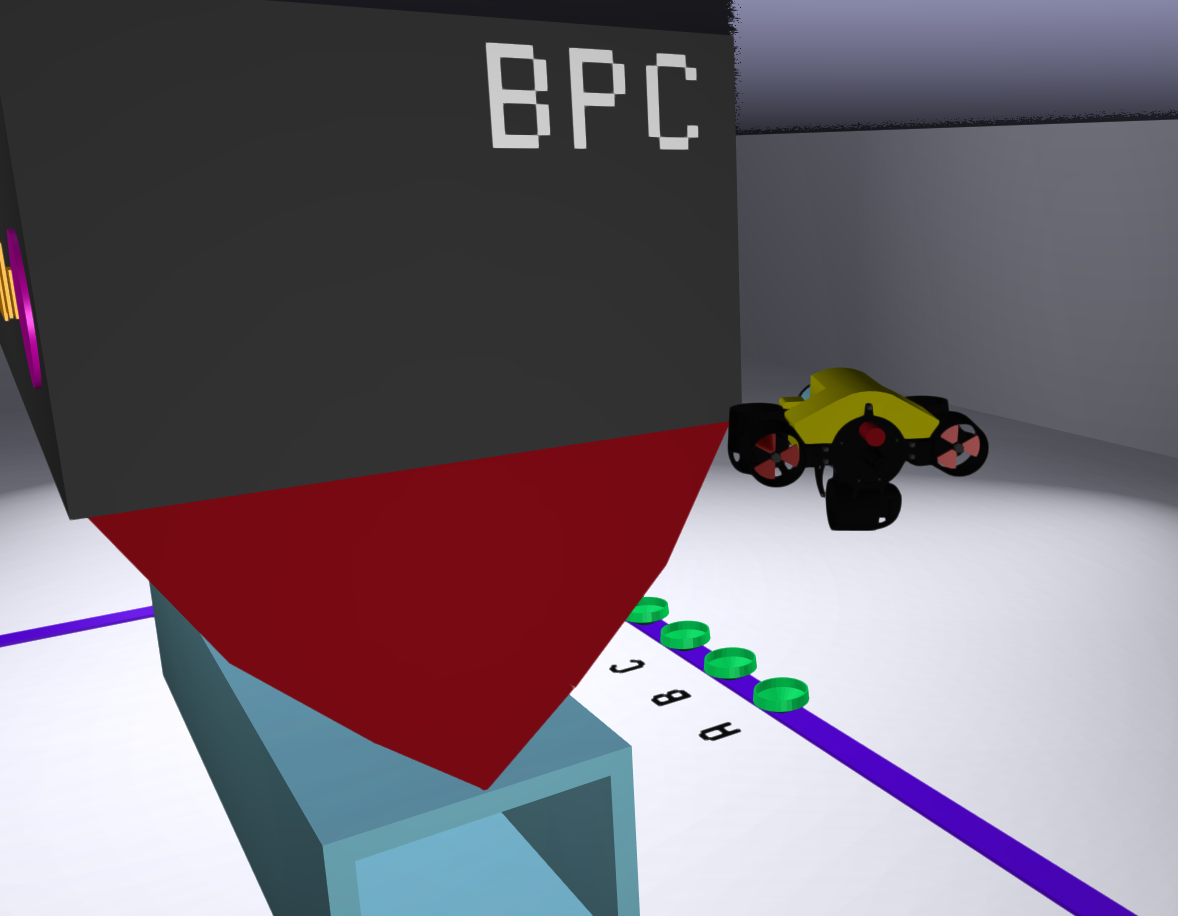
Вход в коридор со стороны кормы обозначен оранжевой стрелкой.



*Рисунок 4. Исследование корпуса*

### **Задача 5.** **Всплытие - 5 баллов**

После завершения миссии аппарат должен всплыть со стороны носовой части над красным кругом.



*Рисунок 5. Область всплытия*

# Требования к выполнению задания

* Время выполнения задания (попытка) - не более **5 мин.**
* Попытка завершается в тот момент, как робот всплыл на поверхность. Всплытие на поверхность означает, что своей верхней частью робот пересек поверхность воды. В этом случае время останавливается, и в зачет идут те баллы, которые робот успел набрать до всплытия.
* Попытка завершается, если истекло максимальное время на попытку (5 мин). В этом случае время выполнения попытки устанавливается 5 мин, и в зачет идут те баллы, которые робот успел набрать до остановки времени.
* Попытка останавливается в случае завершения кода.

# Критерии оценивания

Баллы за прохождение начисляются следующим образом:

| **Задача** | **Критерий** | **Балл** |
| --- | --- | --- |
| 1 | Определен самый большой дефект  (количество выстрелов в мишень совпадает с порядковым номером объекта) | 10 |
| 2 | Винты очищены ( по 5 за каждый ) | 10 |
| 3 | Определен размер трещины  15 - верно  5 - с ошибкой ±1 | 15 |
| 4 | Обследование выполнено | 10 |
| 5 | Всплытие | 5 |
| 6 | Штраф за маркер вне корзины | -2 |
| 7 | Штраф за снаряд, выпущенный не в мишень | -2 |
| 8 | Штраф за касание неверной (серой) лопасти | -2 |

# Подготовка и тестирование

Для тестирования участникам доступны 3 сцены. Обратите внимание, что контрольные сцены будут отличаться.

Скачать сцены для тестирования можно [**по ссылке**](https://disk.yandex.ru/d/PG8qTRkwhmF-ig)

Размеры трещин в каждой схеме:

scene\_1.xml - 9

scene\_2.xml - 5

scene\_3.xml - 3

## Что может меняться

* Размер квадратов-повреждений.
* Длина трещины
* Расположение обрастаний на винтах
* Положение корзин не меняется
* Центр трещины всегда совпадает с центром калибровочного квадрата. При этом положение калибровочного квадрата не меняется

# Порядок оценки решений и порядок работы системой автоматической проверки

* **Решением задачи является файл в формате \*.py,** который загружается на платформу MUR Contest и оценивается автоматически.
* В MUR contest вам будут доступны два соревнования: НТО (тестовые) и НТО (финальные).
* В НТО (тестовые) вы можете выполнять проверку своего решения на тестовых сценах и получать результаты. **Количество попыток: 10.**
* В НТО (финальные) вы загружаете свое итоговое решение. Это решение проверяется на контрольных сценах после завершения приема работ.
* С инструкциями по работе с системой можно ознакомиться в личном кабинете участника внутри соревнования.
* Участники могут заменять файл итоговой программы до завершения приема работ.
* Заключительная проверка выполняется после завершения приема работ второго этапа. **Проверку проходит последний файл, присланный участником.**
* Задача проверяются на трех сценах, которые отличаются от предоставленных.
* Сцены различаются расположением объектов, их цветом.
* В зачет идет **худший** результат из трех попыток.
* Все работы проверяются на плагиат. **О проверке на плагиат читайте в** [**Правилах второго этапа**](https://docs.google.com/document/d/1MONNfyCbtsGGgbl53uG-WwNMJQSUxWGeINt3rytJafQ/edit?usp=sharing)**.**