

# Регламент категории “Программирование”

версия 1.10.24

## 1 Общая информация

- Участниками старшей категории могут стать ученики **2006-2009** года рождения.
- Формат участия: индивидуальный
- Формат проведения: заочный (онлайн)
- Время на выполнение задания: 3 часа.
- Группа для участников в Telegram: <https://t.me/+A1DxZBcfYLRiNjA6>

## 2 Описание соревнований и правил

- Данная категория проводится в формате хакатона. Участникам необходимо решить задание по программированию подводного робота в симуляторе MUR IDE за ограниченное время.
- **Задачи:** участникам предоставляется три задачи разного уровня сложности, которые нужно решить в течение соревнований. Задачи участники получают в начале соревнований.
- **Время:** на решение всех задач отводится 3 часа.
- **Среда разработки:** MUR IDE, язык программирования Python,
- Оценка работ: в системе автоматической проверки.
- Для проверки участникам необходимо загрузить решения каждой задачи в формате `name_task1.py`.
- Условия задач и критерии оценки будут размещены на платформе после старта соревнований. Для отладки участникам предоставляются тестовые сцены.
- Тестирование и оценка задач будет проводиться в системе автоматической проверки. Доступ к личному кабинету участника команды получают в начале соревнований.

## 3 Описание задания

Соревновательное задание будет состоять из трех задач:

1. Ориентирование - 10 баллов
2. Маневры - 15 баллов
3. Распознавание - 25 баллов

**Итого за все задачи: 50 баллов.**

В рамках соревнований участники должны уметь решать следующие задачи с использованием MUR IDE и симулятора:

- применение регуляторов по курсу и глубине,
- следование по линии,
- анализ изображений передней и донной камеры,

- распознавание фигур (круг, треугольник, квадрат)
- подсчет фигур одного типа (круги, треугольники, квадраты),
- выполнение простых маневров (поворот, касание стенки, погружение/всплытие),
- сброс маркеров в корзины,
- выстрел торпедами в цели.

### Требования к решению

- Одна задача - один файл
- Скрипт на языке Python
- Версия, совместимая с MUR IDE

## 4 Тренировочные задания и сцены

Тренировочные сцены для подготовки: [ссылка](#)

Для подготовки к соревнованиям участникам предлагаются следующие тренировочные сцены, которые позволят отработать навыки программирования по основным направлениям:

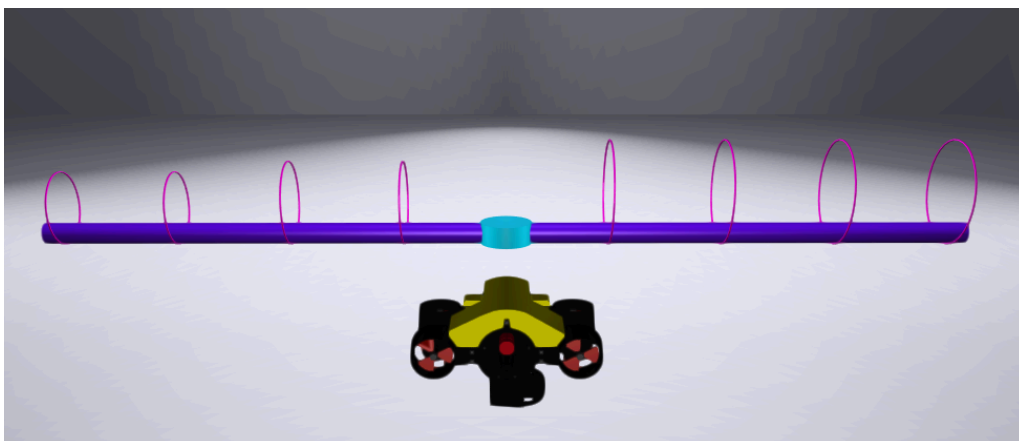
### 4.1. Ориентирование (scene\_1\_1, scene\_1\_2)

Начальное положение аппарата - над центральным кругом. Задача: пройти по линии через все кольца.

Глубина расположения линии может меняться и будет указана в условии задачи.

В данных сценах линия расположена на глубинах: (сцена 1\_1 : **3.81 м**, сцена 1\_2: **1.87 м**)

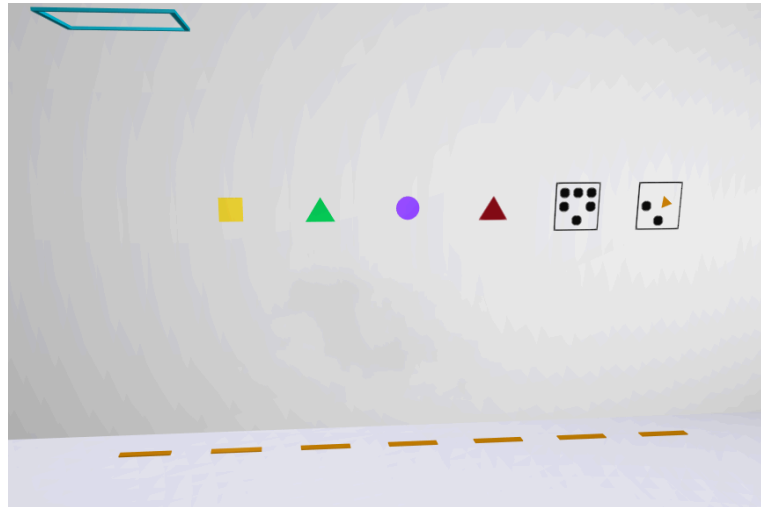
Размер колец может меняться (варианты представлены в сценах).



### 4.2. Распознавание фигур и маневры (scene\_2)

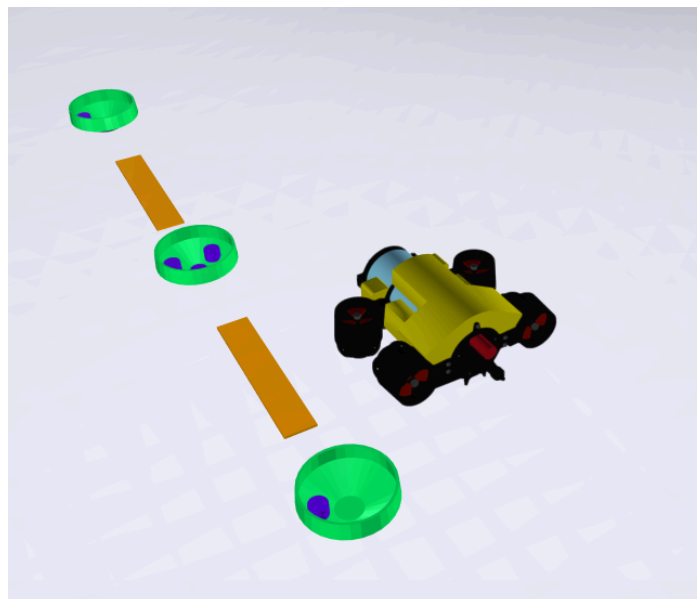
В данной сцене вы можете потренироваться в движении лагом вдоль стены, распознавании фигур, совершении простых маневров напротив фигур, выстреле

торпедами в фигуру. В конце набора фигур располагаются две таблички с фигурами, на которых можно потренироваться в распознавании и подсчете фигур.



#### 4.3. Корзины (scene\_3)

В данной сцене вы можете потренироваться в подсчете фигур в корзинах и сбросе маркеров в них.



### 5 Порядок начисления баллов и определения победителей

- У каждого участника будет **3 попытки** на загрузку решения.
- Каждая попытка - это тестирование вашего решения на трех контрольных сценах. Обращаем внимание, что контрольные сцены отличаются от сцен, предоставленных для тестирования.

- После завершения тестирования участник получает отчет о набранных баллах к каждом из трех тестов.
- Результат попытки - это худший результат из трех тестов.
- Итоговый результат - это наилучшая попытка из всех совершенных.
- Для участников соревнований доступен общий рейтинг. В рейтинге отображается название команды, лучшая попытка и время выполнения задания.
- Если несколько участников набирают одинаковое количество баллов, то в зачет идет общее время, затраченное на выполнение трех задач. Выше в рейтинге оказывается тот участник, который выполнил миссию быстрее.

## 6. Онлайн система автоматической проверки

- Работает со всеми библиотеками MUR IDE.
- При загрузке решений на проверку формируется очередь на проверку работ.
- Порядок очереди будет доступен участникам на отдельной вкладке.
- Пользователь может отменить проверку (удалить решение), если проверка ещё не началась. После начала проверки отменить ее уже нельзя.
- Попытка считается потраченной, если началась проверка решения.
- Нельзя загрузить новое решение, пока не завершится проверка предыдущего.
- После завершения проверки работы на указанный при регистрации e-mail придет уведомление о готовности результатов.
- Полный отчет о попытке пользователь может увидеть в своем личном кабинете.
- При проверке решений графический интерфейс opencv не будет доступен (например не будет доступна функция `cv2.imshow('image', image)`). И попытка вызвать подобные функции могут вызвать ошибку.

## Описание симулятора

- Version: MUR IDE 0.1.0 ( <https://murproject.com/#muride> )
- Download MURIDE: [https://murproject.com/documents/76/mur\\_installer\\_0\\_1\\_0.exe](https://murproject.com/documents/76/mur_installer_0_1_0.exe)
- Python version: python 3.11.2
- OpenCV library version: opencv 4.7.0

## Уроки по работе в MUR IDE и компьютерному зрению

<https://vk.com/video/@murproject>