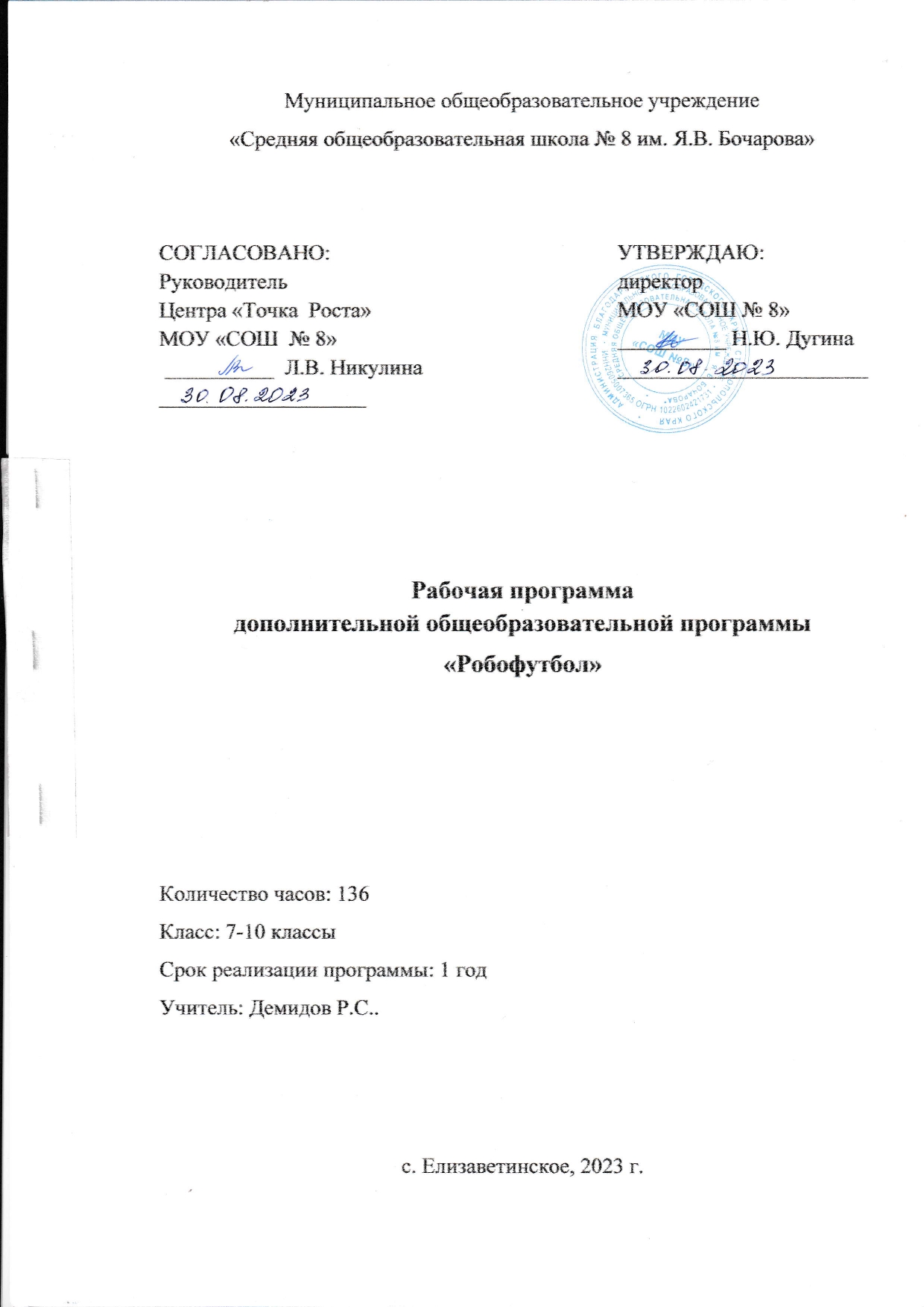
****

**Образовательная программа**

Эта образовательная программа предназначена для использования преподавателями в кружках по робототехнике и рассчитана на один учебный год. Темы, предлагаемые к изучению, покрывают необходимый минимум знаний для участия в соревнованиях по робототехнике. Соревнованием, в котором ученики разных кружков будут соперничать друг с другом, будет гуманоидный робофутбол. Первый чемпионат будет проведен в ноябре, второй в мае, и они пройдут в симуляции.

Логичный вопрос, который должен возникнуть - а почему это вообще нужно ученикам? Где им пригодится робофутбол? Не лучше ли им заняться ”настоящей” учебой вместо каких-то игрушек? Краткий ответ таков, что нет, не лучше, поскольку робофутбол - это как раз один из лучших способов познакомиться с робототехникой.

Во имя целостности структурируем ответы на вопросы формального характера, чтобы подтвердить обоснованность представленной программы.

**Актуальность**

Робототехника не наука, а скорее совокупность задач из разных областей программи- рования, искусственного интеллекта, физики и математики. Это та самая настоящая практика, о которой мечтают и преподаватели, и ученики, и родители. Часто бывает, что в курсе информатики многие важные темы затрагиваются, но остаются без конкретного применения и не закрепляются. Если один ученик решил задачу с квадратичным време- нем работы, а другой с линейным, это может остаться незамеченным, и положительной обратной связи не возникнет. В случае же немедленного внедрения в робота код первого ученика просто будет работать быстрее, и из подобных вещей в итоге сложится место его команды в турнирной таблице. Примеры можно привести в рамках всех перечисленных дисциплин. Здорово, когда школьники знакомы с производными и интегрированием. Но при этом гораздо более ценно, когда они понимают, как именно ходьба гуманоидного робота связана с задачей стабилизации обратного маятника, как это просимулировать, и как должны быть устроены ноги робота, чтобы их момент инерции был невелик.

Задачаробофутбола была выбрана лигой RoboCup как один из тех видов соревнова- ний, в которых могут быть применены очень многие наработки из области робототехники. В их числе командная игра, ходьба по неровной поверхности, компьютерное зрение в меняющихся условиях, нахождение собственных координат. Команда Старкит МФТИ выиграла чемпионат мира в 2021 году и после этого приступила к созданию курса, который и предлагается к изучению.

**Цель**

Курс предназначен для того, чтобы познакомить слушателей со всеми основными составными частями робототехнической системы. Этот уровень должен быть достаточен для участия в соревнованиях по робофутболу школьного уровня, поэтому число занятий и глубина погружения в материал потребуют и от учеников, и от преподавателей боль- шой отдачи. После успешного прохождения курса ученики овладеют теоретическими, математическими и программными методами, а также инструментами разработки и проектирования, достаточными для дальнейшего изучения робототехники на серьезном уровне.

**Задачи**

• Развитие навыков в области программирования, робототехники и искусственного интеллекта. Работа со сложными системами естественным образом потребует от учеников овладения теоретической и практической базой в предметной области

• Получение опыта командной работы на примере соревнований с другими командами близкого уровня

• Стимулирование самостоятельной работы для достижения поставленной за- дачи. В рамках разработки сложного робототехнического комплекса ученики будут многие сотни раз обращаться к документации, тематическим форумам, лекциям, видеоурокам

**Сроки реализации**

Вся программа разделена на девять месячных блоков. Два из них - ноябрьский и майский - посвящены соревнованиям. В сентябре и октябре ученики знакомятся с предметной областью, с составными частями ПО робота. Декабрь будет посвящен изучению методов локализации. В январе на занятиях будут изучены библиотеки и инструменты разработки и тестирования, а именнно ROS и вещи, с которыми он может работать в связке. Курс в целом достаточно нагружен программными средствами и фреймворками, ученики познакомятся с git, OpenCV, PyTorch, Webots, ROS, OpenAI Gym. Опционально можно рассказать им и про LATEX.

В течение учебного года Вы можете отходить от программы настолько, насколько считаете правильным, углубляясь в те или иные обрасти математики, физики, про- граммирования или чего угодно еще, что Вы считаете полезным. Главное, чтобы это помогало ученикам в написании кода для успешного участия в соревнованиях. Методи- ческие материалы, лекции и код призваны быть фундаментом, от которого Вы можете отталкиваться при своем рассказе.

**Форма занятий**

Рекомендуемое число занятий - две пары в неделю, то есть четыре урока (академи- ческих часа), разделенных на две равные части. Такая продолжительность и частота позволят поддерживать погруженность в контекст, которую было бы трудно удержать с одним занятием в неделю или меньшей длительностью.

Типичный размер команды, при котором самопроизвольно возникает и поддерживает- ся разумное распределение обязаннностей, роли в команде и эффективная коммуникация

- это три-шесть человек. Если Вы набрали в кружок десять учеников, можно предложить им разделиться на две команды, которые будут параллельно решать одни и те же задачи, а потом делиться опытом друг с другом.

С кружками часто бывает, что сначала школьников приходит много, они оптимистич- но настроены и заинтересованы, а потом интерес немного угасает, и их число естественным образом снижается. Нужно иметь это обстоятельство в виду и дать широкую огласку набору в кружки. Можно написать пост в школьный паблик, дать объявление в чате родителей, скинуть видео с игрой роботов. Поскольку робототехника на данный момент не изучается так же системно, как математика или физика, попадание в кружок - это зачастую воля случая, и надо дать шанс попробовать и заинтересоваться как можно большему количеству школьников.

В меру сил имеет смысл поддерживать активное общение с учениками в чате, прово- дить дополнительные консультации и увеличивать продолжительность занятий, если в этом возникает необходимость. Практика показывает, что увлеченная аудитория способна заниматься робототехникой и программированием гораздо дольше одной пары.

Около раза в месяц будут проводиться общие для всех кружков онлайн-тестирования для проверки усвоения знаний.

Требования к слушателям, возраст учеников

Для быстрого старта в робототехнике и преподаватель, и ученики должны быть знакомы с языком программирования Python. В случае необходимости стоит начать именно с этого, изучив вместе со слушателями условные операторы, циклы, функции и основные структуры данных, такие как массивы и словари.

Жестких требований к возрасту слушателей нет, но разумно начинать изучать робототехнику, уже когда сформированы представления о программировании в целом. Поэтому имеет смысл отталкиваться от программы в конкретной школе, а также от уровня ученика. Отбор на входе в кружок не должен быть слишком строгим, лучше дать человеку попробовать, чем не пропустить заинтересованного, который догонит общий уровень подготовки.

Погружение в предметную область предполагается постепенным: в начале слушатели в обзорном формате познакомятся с основными составными частями программного обеспечения робота, а за этом последует более детальное и формализованное изложение материала.

Для синхронизации в начале некоторых блоков предусмотрены лекции от сотрудников команды Старкит. Можно смотреть их отдельно, а потом формировать с их помощью свой рассказ, можно смотреть их с учениками. Цель этих лекций - с некоторой периодичностью рассказывать об устройстве той или иной области внутри робототехники с высоты птичьего полета, то есть это лекции скорее обзорные, чем технические.

Поскольку и дисциплина достаточно молодая, и кружки только будут открываться, вся эта затея потребует большого объема самостоятельной работы, подготовки, обучения. Крайне приветствуются вопросы любого уровня сложности, общение с коллегами и обмен опытом через форум.

Почитать о лиге ELSIROS и поставить на компьютер комплект программного обеспе- чения можно здесь. Все вопросы, касающиеся программирования роботов, предлагается обсуждать на специальном форуме. Конечно, общение в мессенджерах привычнее и быстрее, но использование форума позволит накопить базу часто задаваемых вопросов, для которых в соответствующих ветках уже будут ответы.

|  |  |  |  |
| --- | --- | --- | --- |
| Тематическое планирование | | | |
| № п/п | Тема | Кол-во часов | Дата |
|  | Введение[сентябрь] |  |  |
| 1-2 | 1. Знакомство с областью, современная робототехника [лекция от Старкит] История робототехники, развитие шагающих механизмов, типы актуаторов, иерар- хия управления, составные части робототехнической системы, пионерские разра- ботки в области, перспективы, проблемы и задачи современной робототехники | 2 |  |
| 3-4 | 2. Webots/ELSIROS  Робофутбол, основные задачи, установка программного обеспечения, настройка среды, запуск игры со стандартным кодом | 2 |  |
| 5-6 | 3. Настройка среды  Установка интерпретатора Python, установка Jupyter Notebook, получение доступа к камере, операции с массивами, Google Colaboratory | 2 |  |
| 7-8 | 4. Введение в классическое зрение, часть 1  Представление цифровых изображений, цветовые пространства, детектирование объектов по цвету, маски | 2 |  |
| 9-10 | 5. Введение в классическое зрение, часть 2  Обработка масок, морфологические операции, свертки, связные компоненты | 2 |  |
| 11-12 | 6. Cлежение за мячом, часть 1  Получение кадров с камеры в симуляции, создание и настройка детектора мяча, управление сервомоторами в симуляции | 2 |  |
| 13-14 | 7. Слежение за мячом, часть 2  Среда и агент, обратная связь, цикл управления. Создание программы, следящей за движениями мяча на поле | 2 |  |
| 15-16 | 8. Системы контроля версий  Предпосылки к появлению систем контроля версий, история систем контроля версий, git, операции с git, полезные практики при работе с git | 2 |  |
|  | ELSIROS[октябрь] |  |  |
| 17-18 | 1. Структура проекта ELSIROS  Структура файлов и папок, основные модули, транспорт данных, допущения и условности симуляции | 2 |  |
| 19-20 | 2. Транспорт данных в ELSIROS  Обмен данными с помощью контроллера player | 2 |  |
| 21-22 | 3. Создание движений  Создание простого движения и его запуск, Pose designer | 2 |  |
| 23-24 | 4. Сенсоры и актуаторы  Считывание положений сервомоторов, управление сервомоторами, считывание показаний IMU, состояние падения | 2 |  |
| 25-26 | 5. Стратегия  Формализация игровых ситуаций, конечные автоматы, обратная связь, создание стратегии робота на простом примере | 2 |  |
| 27-28 | 6. Данные о мире    Получение данных о положении мяча и робота из контроллера player, встраивание в существующий код | 2 |  |
| 29-30 | 7. Состояние игры  Получение данных о состоянии игры из контроллера player, встраивание в суще- ствующий код | 2 |  |
| 31-32 | 8. Стратегия  Работа со strategy designer (выбор направления удара), алгоритм работы полевого игрока и вратаря, модификация стратегии, запуск тестовых игр | 2 |  |
|  | Первые соревнования[ноябрь] |  |  |
| 33-34 | 1. Спортивная робототехника, опыт соревнований [лекция от Старкит] Видеозаписи с выступлений, рассказ о различных видах спорта, роботах и занима- тельных случаях | 2 |  |
| 35-36 | 2. Планирование  Распределение задач, зон ответственности, создание плана работ | 2 |  |
| 37-40 | 3. Работа над кодом  Проверка осуществимости желаемых модификаций | 4 |  |
| 41-42 | 4. Работа над кодом | 2 |  |
| 43-44 | 5. Работа над кодом | 2 |  |
| 45-46 | 6. Предварительное тестирование  Запуск тестовых игр, выяснение причин нежелательного поведения робота, дора- ботка кода | 2 |  |
| 47-48 | 7. Финальное тестирование, загрузка кода  Запуск тестовых игр, фиксация конфигурации кода, сборка архива, загрузка для участия в соревнованиях | 2 |  |
| 49-50 | 8. Подведение итогов  Рассказ от победившей команды, подведение итогов внутри команд | 2 |  |
|  | Локализация[декабрь] |  |  |
| 51-52 | 1. Простые методы локализации  Постановка задачи локализации, природа шума в измерениях, обзор существующих методов, триангуляция, альфа-бета фильтр | 2 |  |
| 53-54 | 2. Триангуляция  Реализация локализации с помощью триангуляции | 2 |  |
| 55-56 | 3. Фильтр частиц, часть 1  Предпосылки к разработке фильтра частиц, обзор метода, параметрическое про- странство, генерация гипотез | 2 |  |
| 57-58 | 4. Фильтр частиц, часть 2  Колесо отсева, построение итоговой гипотезы, сходимость метода | 2 |  |
| 59-60 | 5. Работа с фильтром частиц  Визуализация фильтра частиц, настройка параметров, особенности работы | 2 |  |
| 61-62 | 6. Реализация фильтра частиц | 2 |  |
| 63-64 | 7. Ориентирование в трехмерном пространстве  Практика с ARUCO метками, измерение расстояний, получение положения камеры относительно метки | 2 |  |
|  | Фреймворки[январь] |  |  |
| 65-66 | 1. Установка ROS | 2 |  |
| 67-68 | 2. Введение в ROS  Топики, ноды, Publisher, Subscriber | 2 |  |
| 69-70 | 3. Практика с ROS  Cборка пакетов, простой пример из Publisher и Subscriber | 2 |  |
| 71-72 | 4. Визуализация в ROS  Работа с инструментами rviz и rqt | 2 |  |
| 73-74 | 5. ROS и Webots  Обмен данными, получение изображения с камеры | 2 |  |
|  | Классическое компьютерное зрение[февраль] |  |  |
| 75-76 | 1. Алгоритмы  Асимптотические сложности, алгоритмы графовых поисков  8 подходов к детектированию мяча | 2 |  |
| 77-78 | 2. Модель камеры  Pinhole модель, обзор типов камер, стереозрение, облака точек | 2 |  |
| 79-80 | 3. Фильтры  Cглаживание, выделение границ, свертки, фильтр Харриса | 2 |  |
| 81-82 | 4. Детектирование объектов | 2 |  |
| 83-84 | 5. Методы обработки изображений  Вид сверху (birdview), контуры, подсчет числа пальцев по веб-камере | 2 |  |
| 85-86 | 6. Ключевые точки  Ключевые точки, дескрипторы, SIFT, практика по сопоставлению | 2 |  |
| 87-88 | 7. Стереозрение  Параллакс, диспаратность, карты глубины, стереосопоставление | 2 |  |
| 89-90 | 8. Распознавание паттернов  Метод наименьших квадратов для нахождения линий, преобразование Хафа, RANSAC | 2 |  |
|  | Нейросети[март] |  |  |
| 91-92 | 1. Введение  Исторический экскурс, слои, формализация задач, градиентный спуск | 2 |  |
| 93-94 | 2. Практика в обучении сети  Цикла обучения, вывод функции ошибки, гиперпараметры, обучение линейного слоя умножению на два | 2 |  |
| 95-96 | 3. Линейный оператор  Доработка цикла обучения, отслеживание сходимости параметров линейного слоя | 2 |  |
| 97-98 | 4. Домик  Задача классификации на датасете Домик | 2 |  |
| 99-100 | 5. MNIST, часть 1  Полносвязная нейронная сеть на MNIST | 2 |  |
| 101-102 | 6. Работа с данными в Pytorch  Аугментация данных, загрузка, разметка реальных изображений, автоматическая разметка в симуляторе | 2 |  |
| 103-104 | 7. MNIST, часть 2  Сверточная нейронная сеть на MNIST, свертки, подсчет числа обучаемых парамет- ров в модели | 2 |  |
| 105-106 | 8. Примеры и практики из области  Готовые модели (yolo, mediapipe), дообучение (fine-tuning) на второй половине MNIST | 2 |  |
| 107-108 | 9. Сверточная сеть на мячах  Загрузка данных, доработка модели, тестирование на видеопотоке | 2 |  |
|  | Динамические системы[апрель] |  |  |
| 109-110 | 1. Обратный маятник, часть 1  Постановка задачи, стабилизация маятника, визуализация управляющего воздей- ствия | 2 |  |
| 111-112 | 2. Обратный маятник, часть 2  Реализация простой симуляционной среды, добавление задержек, ошибок измерений | 2 |  |
| 113-114 | 3. Работа со средой  Исследование качества стабилизации от параметров системы, варьирование момента инерции, предельного момента, шумов и задержек | 2 |  |
| 115-116 | 4. Cart-pole  Стабилизация системы из каретки и стержня | 2 |  |
| 117-118 | 5. Симуляция физики  Реализация демо с объектами и гравитацией | 2 |  |
| 119-120 | 6. Визуализация  Визуализация трехмерных объектов, триангуляция поверхностей, освещение по- верхностей | 2 |  |
| 121-122 | 7. Динамическая стабилизация кондо∗  Рассмотрение робота как обратного маятника, PID-контроллер | 2 |  |
|  | Вторые соревнования[май] |  |  |
| 123-124 | 1. Планирование  Распределение задач, зон ответственности, создание плана работ | 2 |  |
| 125-126 | 2. Работа над кодом  Проверка осуществимости желаемых модификаций | 2 |  |
| 127-128 | 3. Работа над кодом | 2 |  |
| 129-130 | 4. Работа над кодом | 2 |  |
| 131-132 | 5. Предварительное тестирование  Запуск тестовых игр, выяснение причин нежелательного поведения робота, дора- ботка кода | 2 |  |
| 133-134 | 6. Финальное тестирование, загрузка кода  Запуск тестовых игр, фиксация конфигурации кода, сборка архива, загрузка для участия в соревнованиях | 2 |  |
| 135-136 | 7. Подведение итогов  Рассказ от победившей команды, подведение итогов внутри команд | 2 |  |
|  | Итого | 136 |  |