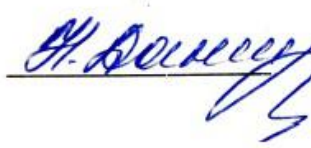


Муниципальное автономное общеобразовательное учреждение города
Новосибирска
«Лицей №22 «Надежда Сибири»
Главный корпус на Советской: г. Новосибирск, ул. Советская, 63, тел. 222-35-15,
e-mail: l_22@edu54.ru
Корпус 99 на Чаплыгина: г. Новосибирск, ул. Чаплыгина, 59, тел. 223-74-15

РАССМОТРЕНО на заседании инженерной кафедры протокол № 1 от 25.08.2025  Кириленко К.А. ФИО руководителя кафедры	СОГЛАСОВАНО Заместитель директора  Н.А.Данилова от 29.08.2025
--	---

РАБОЧАЯ ПРОГРАММА
Информатика. Моделирование физических процессов
10 «ИП» класса
(уровень среднего общего образования)

Разработчик:

Кириленко Ксения Алексеевна

Рабочая программа по учебному предмету «Информатика. Моделирование физических процессов» (предметная область «Математика и информатика») (далее соответственно – программа по моделированию физических процессов, моделирование физических процессов) составлена на основе Федеральной рабочей программы по информатике и является авторской, включает пояснительную записку, содержание обучения, планируемые результаты освоения программы по моделированию физических процессов тематическое планирование.

Пояснительная записка отражает общие цели и задачи изучения искусственного интеллекта, место в структуре учебного плана, а также подходы к отбору содержания, к определению планируемых результатов.

Содержание обучения раскрывает содержательные линии, которые предлагаются для обязательного изучения в 10 классе на уровне среднего общего образования.

Планируемые результаты освоения программы по моделированию физических процессов включают личностные, метапредметные результаты за период обучения в 10 классе на уровне среднего общего образования, а также предметные достижения обучающегося.

1. Пояснительная записка

Общая характеристика учебного предмета "Моделирование физических процессов"

Моделирование физических процессов представляет собой междисциплинарную область, объединяющую физику, математику и информатику для создания компьютерных моделей, имитирующих поведение реальных систем. Обучение основам физического моделирования с использованием библиотеки Runtok в школе открывает учащимся возможность не просто изучать, а буквально «оживлять» законы механики, развивая глубокое понимание физических принципов и продвинутые навыки программирования, востребованные в современном технологическом мире.

Функциональная значимость предмета для школьников заключается в освоении мощного инструмента для создания точных 2D-симуляций, включающих такие сложные концепции, как динамика твердых тел, коллизии, трение, силы, моменты инерции и сочленения (джойнты). Эти знания переводят абстрактные физические формулы из учебника в наглядные и интерактивные цифровые эксперименты, позволяя решать прикладные задачи из области робототехники, игровой индустрии и компьютерной анимации.

Знание принципов физического движка Runtok и умение его применять важно для каждого школьника, стремящегося понять, как создаются современные симуляторы и видеоигры. Понимание того, как описывать форму, массу и свойства объектов, программировать их взаимодействие и реагировать на столкновения, помогает учащимся развивать пространственное и системное мышление, а также способность к инженерному проектированию.

Библиотека Rymunk, выполняя свои основные функции, позволяет учащимся абстрагироваться от сложной низкоуровневой математики и сосредоточиться на концептуальном моделировании системы. Она предоставляет готовый каркас для описания физического мира, что дает возможность сосредоточиться на постановке задачи, анализе результатов и верификации модели, что является ключевым этапом в любой научно-исследовательской и инженерной деятельности.

Обучение моделированию с помощью Rymunk направлено на развитие интеллектуальных и технических способностей учащихся, включая умение формализовать физическую задачу на языке программирования Python, работать с векторами и силами, настраивать параметры модели для достижения правдоподобного поведения и визуализировать результаты с помощью графических библиотек. Это способствует развитию аналитического и критического мышления, навыков отладки сложных систем и работы с документацией, что является crucial для успешного обучения и дальнейшего профессионального роста в IT-сфере.

Содержание программы по моделированию физических процессов ориентировано также на развитие функциональной грамотности учащихся, включая умение читать и понимать техническую документацию (API), использовать готовые программные компоненты для построения более сложных систем, оценивать адекватность модели реальному миру и применять полученные знания для создания собственных проектов, расширяя свои творческие и профессиональные возможности.

Цели и задачи изучения учебного предмета «Искусственный интеллект».

Изучение искусственного интеллекта направлено на достижение следующих целей:

- Формирование системного понимания основ компьютерного моделирования физических систем как междисциплинарного метода научного познания, связывающего теоретические законы физики с их практической реализацией средствами программирования.
- Развитие компетенций в области создания, анализа и верификации 2D-моделей механических явлений (кинематика, динамика, столкновения) с использованием современного инструментария — физического движка Rymunk.
- Стимулирование интереса к исследовательской и проектной деятельности в областях программирования, физики, робототехники и игровой разработки через создание наглядных и интерактивных симуляций.

Задачи изучения предмета:

- Развивать алгоритмическое и логическое мышление через процесс перевода физических законов в программный код.
- Формировать навыки проектной деятельности: от постановки задачи и планирования этапов работы до тестирования, отладки и презентации готового проекта.

- Воспитывать научную добросовестность, понимая важность верификации модели и критической оценки результатов симуляции.
- Развивать умение самостоятельно учиться: работать с технической документацией (API Rymunk), искать решения возникающих проблем в профессиональных сообществах и ресурсах.

Особенности классов

Рабочая программа по предмету «Информатика. Моделирование физических процессов» для 10-го «ИП» класса предназначена для углубленного изучения учащимися информационно-технологического профиля в группе «Киберфизические индустрии». На изучение данного модуля отведено 33 часа в 10-м классе.

Место предмета в учебном плане лица

Учебный план на изучение «Информатика. Моделирование физических процессов» в 10 «ИП» классе среднего общего образования отводит 1 учебный час в неделю (всего 33 учебных часов) за счёт части, формируемой участниками образовательных отношений.

Учебный год	Количество часов
	10 «ИП»
2025/2026	33

К тематическому планированию применяется модульный принцип построения образовательной программы, что позволяет выстраивать индивидуальную образовательную парадигму и обеспечивать саморазвитие при индивидуальном темпе работы с учебным материалом, контроль и самоконтроль знаний.

Используемые образовательные технологии, в том числе дистанционные

Обучение искусственному интеллекту может осуществляться с использованием дистанционных образовательных технологий (далее ДОТ), которое предполагает как самостоятельное прохождение учебного материала учеником, так и с помощью сопровождения учителя. При применении ДОТ используются платформы: лицейская платформа дистанционного обучения Moodle, ФГИС «Моя школа», ГИС «Электронная школа» Новосибирской области.

При реализации рабочей программы могут быть использованы материалы для подготовки к профилям олимпиады КД НТИ и стандартов Всероссийского чемпионатного движения по профессиональному мастерству «Профессионалы».

Информация о промежуточной аттестации

Промежуточная аттестация осуществляется по окончании учебного модуля с целью проверки степени и качества усвоения материала по результатам изучения тематических модулей и проводится в форме аттестационных работ.

Текущий контроль осуществляются с целью проверки степени и качества усвоения материала в ходе его изучения в следующих формах: самостоятельных и проверочных работ.

Текущий контроль и промежуточная аттестация осуществляются в соответствии с «Положением об осуществлении текущего контроля успеваемости и промежуточной аттестации обучающихся, их формах, периодичности и порядке проведения муниципального автономного общеобразовательного учреждения города Новосибирска «Лицей № 22 «Надежда Сибири» (протокол педагогического совета №1 от 29.08.2023).

Итоговая аттестация проводится в соответствии с законодательством РФ.

Промежуточная аттестация по искусственному интеллекту в 10 «ИП» классе

№ модульн ой	Название модуля	Количество часов модуле	Номер урока ПА	Форма ПА
МР № 1	Простая физическая сцена	7	7	Практическая работа
МР № 2	Система со столкновениями	5	12	Практическая работа
МР № 3	Механическая система	5	17	Практическая работа
МР № 4	Создание комплексной физической симуляции	16	33	Практическая работа

2. Содержание учебного предмета

Модуль 1 «Простая физическая сцена»

Основы программирования на Python. Знакомство с концепцией физического моделирования. Установка необходимых библиотек. Базовые понятия PyMunk. Создание физического пространства (Space). Добавление простых статических тел. Создание и настройка простых физических форм. Работа с отрезками и окружностями. Создание динамических тел. Настройка массы, момента инерции. Базовые преобразования тел. Работа с полигональными формами. Создание составных тел из нескольких простых форм. Применение сил к телам. Настройка гравитации. Импульсы и их применение.

Модуль 2 «Система со столкновениями»

Принципы обнаружения столкновений. Настройка обработки простых коллизий между телами. Настройка физических свойств материалов. Коэффициенты упругости и трения для различных поверхностей. Создание пользовательских обработчиков столкновений. Реакция на различные типы коллизий. Работа с PinJoint. Создание простых механических систем с шарнирными соединениями.

Модуль 3 «Механическая система»

Использование скользящих соединений. Создание систем с ограниченным перемещением. Работа с направляющими соединениями. Создание механизмов с заданной траекторией движения. Моделирование упругих свойств. Настройка жесткости и демпфирования в соединениях. Комбинирование различных типов соединений. Построение сложных механических конструкций.

Модуль 4 «Создание комплексной физической симуляции»

Работа с констрейнтами. Ограничение степеней свободы тел. Установка и базовое использование Pygame. Создание простого графического окна. Отображение тел и форм PyMunk в Pygame. Синхронизация физического и графического пространства. Обработка пользовательского ввода. Создание интерактивных элементов управления симуляцией. Разработка интерактивного приложения для экспериментов с физическими объектами. Создание и анализ работы простых механизмов: рычаги, блоки, наклонные плоскости. Моделирование систем передачи движения. Создание цепных и ременных механизмов. Изучение колебательного движения. Создание различных типов маятниковых систем. Создание и тестирование мостовых конструкций. Анализ устойчивости сооружений. Моделирование систем частиц. Основы симуляции жидкостей с помощью частиц. Реализация перемещения и масштабирования камеры. Работа с преобразованием координат. Создание элементов управления: кнопки, переключатели, слайдеры для управления параметрами. Визуализация физических параметров: сил, скоростей, точек контакта. Использование простых методов анализа данных. Визуализация результатов экспериментов. Основные методы оптимизации физических симуляций. Пространственное разбиение для больших сцен.

Планируемые образовательные результаты освоения учебного предмета Искусственный интеллект

Личностные результаты

1. сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
2. готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности;
3. навыки сотрудничества со сверстниками, детьми младшего возраста, взрослыми в образовательной, учебно-исследовательской, проектной и других видах деятельности;
4. эстетическое отношение к миру, включая эстетику научного и технического творчества;

Метапредметные результаты:

1.2.5	Анализировать полученные в ходе решения задачи результаты, критически оценивать их достоверность, прогнозировать изменение в новых условиях
1.2.6	Уметь переносить знания в познавательную и практическую области жизнедеятельности; уметь интегрировать знания из разных предметных областей; осуществлять целенаправленный поиск переноса средств и способов действия в профессиональную среду
1.2.7	Способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания; ставить и формулировать собственные задачи в образовательной деятельности и жизненных ситуациях; ставить проблемы и задачи, допускающие альтернативные решения; выдвигать новые идеи, предлагать оригинальные подходы и решения; разрабатывать план решения проблемы с учетом анализа имеющихся материальных и нематериальных ресурсов
1.3	Работа с информацией
1.3.1	Владеть навыками получения информации из источников разных типов, самостоятельно осуществлять поиск, анализ, систематизацию и интерпретацию информации различных видов и форм представления
1.3.2	Создавать тексты в различных форматах с учетом назначения информации и целевой аудитории, выбирая оптимальную форму представления и визуализации
1.3.3	Оценивать достоверность, легитимность информации, ее соответствие правовым и морально-этическим нормам
1.3.4	Использовать средства информационных и коммуникационных технологий в решении когнитивных, коммуникативных и организационных задач с соблюдением требований эргономики, техники безопасности, гигиены, ресурсосбережения, правовых и этических норм, норм информационной безопасности
1.3.5	Владеть навыками распознавания и защиты информации, информационной безопасности личности
2	Коммуникативные УУД
2.1	Общение
2.1.1	Осуществлять коммуникации во всех сферах жизни; владеть различными способами общения и взаимодействия
2.1.2	Развернуто и логично излагать свою точку зрения с использованием языковых средств
2.1.3	Аргументированно вести диалог
3	
3.1	Самоорганизация
3.1.1	Самостоятельно осуществлять познавательную деятельность, выявлять проблемы, ставить и формулировать собственные задачи в образовательной деятельности и жизненных ситуациях; давать оценку новым ситуациям
3.1.2	Самостоятельно составлять план решения проблемы с учетом имеющихся ресурсов, собственных возможностей и предпочтений; делать осознанный выбор, аргументировать его, брать ответственность за решение; оценивать приобретенный опыт; способствовать формированию и проявлению широкой эрудиции в разных областях знаний
3.2	Самоконтроль

3.2.1	Давать оценку новым ситуациям, вносить коррективы в деятельность, оценивать соответствие результатов целям
3.2.2	Владеть навыками познавательной рефлексии как осознания совершаемых действий и мыслительных процессов, их результатов и оснований; использовать приемы рефлексии для оценки ситуации, выбора верного решения; уметь оценивать риски и своевременно принимать решения по их снижению
3.3	Эмоциональный интеллект, предполагающий сформированность: саморегулирования, включающего самоконтроль, умение принимать ответственность за свое поведение, способность адаптироваться к эмоциональным изменениям и проявлять гибкость, быть открытым новому; внутренней мотивации, включающей стремление к достижению цели и успеху, оптимизм, инициативность, умение действовать, исходя из своих возможностей

Выпускник научится:

1. самостоятельно определять цели деятельности и составлять планы деятельности; самостоятельно осуществлять, контролировать и корректировать деятельность; использовать все возможные ресурсы для достижения поставленных целей и реализации планов деятельности; выбирать успешные стратегии в различных ситуациях;

2. продуктивно общаться и взаимодействовать в процессе совместной деятельности, учитывать позиции других участников деятельности, эффективно разрешать конфликты;

3. владеть навыками познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем;

4. использовать средства информационных и коммуникационных технологий в решении когнитивных, коммуникативных и организационных задач с соблюдением требований эргономики, техники безопасности, гигиены, ресурсосбережения, правовых и этических норм, норм информационной безопасности.

Выпускник получит возможность научиться:

1. быть готовым и способным к самостоятельной информационно-познавательной деятельности, включая умение ориентироваться в различных источниках информации, критически оценивать и интерпретировать информацию, получаемую из различных источников;

2. быть способным и готовым к самостоятельному поиску методов решения практических задач, применению различных методов познания.

Регулятивные универсальные учебные действия

1. самостоятельно определять цели, задавать параметры и критерии, по которым можно определить, что цель достигнута;

2. оценивать возможные последствия достижения поставленной цели в деятельности, собственной жизни и жизни окружающих людей, основываясь на соображениях этики и морали;

3. ставить и формулировать собственные задачи в образовательной деятельности и жизненных ситуациях;

4. оценивать ресурсы, в том числе время и другие нематериальные ресурсы, необходимые для достижения поставленной цели;

5. выбирать путь достижения цели, планировать решение поставленных задач, оптимизируя материальные и нематериальные затраты;

6. организовывать эффективный поиск ресурсов, необходимых для достижения поставленной цели;

7. сопоставлять полученный результат деятельности с поставленной заранее целью.

Познавательные универсальные учебные действия

1. искать и находить обобщенные способы решения задач, в том числе, осуществлять развернутый информационный поиск и ставить на его основе новые (учебные и познавательные) задачи;
2. критически оценивать и интерпретировать информацию с разных позиций, распознавать и фиксировать противоречия в информационных источниках;
3. использовать различные модельно-схематические средства для представления существенных связей и отношений, а также противоречий, выявленных в информационных источниках;
4. находить и приводить критические аргументы в отношении действий и суждений другого; спокойно и разумно относиться к критическим замечаниям в отношении собственного суждения, рассматривать их как ресурс собственного развития;
5. выходить за рамки учебного предмета и осуществлять целенаправленный поиск возможностей для широкого переноса средств и способов действия;
6. выстраивать индивидуальную образовательную траекторию, учитывая ограничения со стороны других участников и ресурсные ограничения;
7. менять и удерживать разные позиции в познавательной деятельности.

Коммуникативные универсальные учебные действия

1. осуществлять деловую коммуникацию как со сверстниками, так и со взрослыми (как внутри образовательной организации, так и за ее пределами), подбирать партнеров для деловой коммуникации исходя из соображений результативности взаимодействия, а не личных симпатий;
2. при осуществлении групповой работы быть как руководителем, так и членом команды в разных ролях (генератор идей, критик, исполнитель, выступающий, эксперт и т.д.);
3. координировать и выполнять работу в условиях реального, виртуального и комбинированного взаимодействия;
4. развернуто, логично и точно излагать свою точку зрения с использованием адекватных (устных и письменных) языковых средств;
5. распознавать конфликтогенные ситуации и предотвращать конфликты до их активной фазы, выстраивать деловую и образовательную коммуникацию, избегая личностных оценочных суждений.

Предметные результаты

Выпускник будет демонстрировать:

- 1) систематизация знаний, относящихся к *математическим объектам информатики*; умение строить математические объекты информатики, в том числе логические формулы;
- 2) сформированность базовых навыков и умений по соблюдению требований *техники безопасности*, гигиены и ресурсосбережения при работе со средствами информатизации;
- 3) владение опытом построения и использования *компьютерно-математических моделей*, проведения экспериментов и статистической обработки данных с помощью компьютера, интерпретации результатов, получаемых в ходе моделирования реальных процессов; умение оценивать числовые параметры моделируемых объектов и процессов; сформированность представлений о необходимости *анализа соответствия модели* и моделируемого объекта (процесса);
- 4) сформированность представлений о способах хранения и простейшей обработке данных; умение пользоваться *базами данных* и справочными системами; владение основными сведениями о базах данных, их структуре, средствах создания и работы с ними;

Выпускник получит возможность продемонстрировать:

1. владение системой базовых знаний, отражающих *вклад информатики* в формирование современной научной картины мира;

2. владение навыками *алгоритмического мышления* и понимание необходимости формального описания алгоритмов.

ПРОВЕРЯЕМЫЕ НА ЕГЭ ПО ИНФОРМАТИКЕ ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ОСНОВНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ СРЕДНЕГО ОБЩЕГО ОБРАЗОВАНИЯ

Код проверяемого требования	Проверяемые требования к предметным результатам освоения основной образовательной программы среднего общего образования
1.	<i>Знать (понимать)</i>
1.1	Понимание основных принципов устройства и функционирования современных стационарных и мобильных компьютеров; тенденций развития компьютерных технологий; владение навыками работы с операционными системами и основными видами программного обеспечения для решения учебных задач по выбранной специализации
1.4	Понимание базовых алгоритмов обработки числовой и текстовой информации (запись чисел в позиционной системе счисления, делимость целых чисел; нахождение всех простых чисел в заданном диапазоне; обработка многоразрядных целых чисел; анализ символьных строк и других), алгоритмов поиска и сортировки
1.5	Знание функциональные возможности инструментальных средств среды разработки
1.6	Владение основными сведениями о базах данных, их структуре, средствах создания и работы с ними
1.7	Понимание возможностей и ограничений технологий искусственного интеллекта в различных областях; наличие представлений об использовании информационных технологий в различных профессиональных сферах
2.	<i>Уметь</i>
2.1	Умение использовать компьютерно-математические модели для анализа объектов и процессов: формулировать цель моделирования, выполнять анализ результатов, полученных в ходе моделирования; оценивать адекватность модели моделируемому объекту или процессу; представлять результаты моделирования в наглядном виде
2.2	Умение классифицировать основные задачи анализа данных (прогнозирование, классификация, кластеризация, анализ отклонений); понимать последовательность решения задач анализа данных: сбор первичных данных, очистка и оценка

Код проверяемого требования	Проверяемые требования к предметным результатам освоения основной образовательной программы среднего общего образования
	качества данных, выбор и (или) построение модели, преобразование данных, визуализация данных, интерпретация результатов
2.9	Умение анализировать алгоритмы с использованием таблиц трассировки; определять без использования компьютера результаты выполнения несложных программ, включающих циклы, ветвления и подпрограммы, при заданных исходных данных
2.10	Умение определять сложность изучаемых в курсе базовых алгоритмов (суммирование элементов массива, сортировка массива, переборные алгоритмы, двоичный поиск) и приводить примеры нескольких алгоритмов разной сложности для решения одной задачи
2.11	Владение универсальным языком программирования высокого уровня (Паскаль, Python, Java, C++, C#), представлениями о базовых типах данных и структурах данных; умение использовать основные управляющие конструкции; умение осуществлять анализ предложенной программы: определять результаты работы программы при заданных исходных данных; определять, при каких исходных данных возможно получение указанных результатов; выявлять данные, которые могут привести к ошибке в работе программы; формулировать предложения по улучшению программного кода
2.12	Умение реализовывать на выбранном для изучения языке программирования высокого уровня (Паскаль, Python, Java, C++, C#) типовые алгоритмы обработки чисел, числовых последовательностей и массивов: представление числа в виде набора простых сомножителей; нахождение максимальной (минимальной) цифры натурального числа, записанного в системе счисления с основанием, не превышающим 10; вычисление обобщённых характеристик элементов массива или числовой последовательности (суммы, произведения среднего арифметического, минимального и максимального элементов, количества элементов, удовлетворяющих заданному условию); сортировку элементов массива; умение использовать в программах данные различных типов с учётом ограничений на диапазон их возможных значений, применять при решении задач структуры данных (списки, словари, стеки, очереди, деревья); применять стандартные и собственные подпрограммы для обработки числовых данных и символьных строк; использовать

Код проверяемого требования	Проверяемые требования к предметным результатам освоения основной образовательной программы среднего общего образования
	при разработке программ библиотеки подпрограмм; умение использовать средства отладки программ в среде программирования

ПЕРЕЧЕНЬ ЭЛЕМЕНТОВ СОДЕРЖАНИЯ, ПРОВЕРЯЕМЫХ НА ЕГЭ ПО ИНФОРМАТИКЕ

Код	Проверяемый элемент содержания
3	Алгоритмы и программирование
3.1	Формализация понятия алгоритма. Машина Тьюринга как универсальная модель вычислений
3.2	Оценка сложности вычислений. Время работы и объём используемой памяти, их зависимость от размера исходных данных. Оценка асимптотической сложности алгоритмов. Алгоритмы полиномиальной сложности. Переборные алгоритмы. Примеры различных алгоритмов решения одной задачи, которые имеют различную сложность
3.3	Определение возможных результатов работы простейших алгоритмов управления исполнителями и вычислительных алгоритмов. Определение исходных данных, при которых алгоритм может дать требуемый результат
3.4	Алгоритмы обработки натуральных чисел, записанных в позиционных системах счисления: разбиение записи числа на отдельные цифры, нахождение суммы и произведения цифр, нахождение максимальной (минимальной) цифры. Представление числа в виде набора простых сомножителей. Алгоритм быстрого возведения в степень. Поиск простых чисел в заданном диапазоне с помощью алгоритма «решето Эратосфена»
3.5	Многоразрядные целые числа, задачи длинной арифметики
3.6	Язык программирования (Паскаль, Python, Java, C++, C#). Типы данных: целочисленные, вещественные, символьные, логические. Ветвления. Сложные условия. Циклы с условием. Циклы по переменной. Обработка данных, хранящихся в файлах. Текстовые и двоичные файлы. Файловые переменные (файловые указатели). Чтение из файла. Запись в файл. Разбиение задачи на подзадачи. Подпрограммы (процедуры и функции). Использование стандартной библиотеки языка программирования
3.7	Рекурсия. Рекурсивные процедуры и функции. Использование стека для организации рекурсивных вызовов

Код	Проверяемый элемент содержания
3.8	Численные методы. Точное и приближённое решения задачи. Численное решение уравнений с помощью подбора параметра. Численные методы решения уравнений: метод перебора, метод половинного деления. Приближённое вычисление длин кривых. Вычисление площадей фигур с помощью численных методов (метод прямоугольников, метод трапеций). Поиск максимума (минимума) функции одной переменной методом половинного деления
3.9	Обработка символьных данных. Встроенные функции языка программирования для обработки символьных строк. Алгоритмы обработки символьных строк: подсчёт количества появлений символа в строке, разбиение строки на слова по пробельным символам, поиск подстроки внутри данной строки, замена найденной подстроки на другую строку. Генерация всех слов в некотором алфавите, удовлетворяющих заданным ограничениям. Преобразование числа в символьную строку и обратно
3.10	Массивы и последовательности чисел. Вычисление обобщённых характеристик элементов массива или числовой последовательности (суммы, произведения, среднего арифметического, минимального и максимального элементов, количества элементов, удовлетворяющих заданному условию). Линейный поиск заданного значения в массиве. Алгоритмы работы с элементами массива с однократным просмотром массива. Сортировка одномерного массива. Простые методы сортировки (метод пузырька, метод выбора, сортировка вставками). Сортировка слиянием. Быстрая сортировка массива (алгоритм QuickSort). Двоичный поиск в отсортированном массиве
3.11	Двумерные массивы (матрицы). Алгоритмы обработки двумерных массивов: заполнение двумерного числового массива по заданным правилам, поиск элемента в двумерном массиве, вычисление максимума (минимума) и суммы элементов двумерного массива, перестановка строк и столбцов двумерного массива
3.12	Словари (ассоциативные массивы, отображения). Хэш-таблицы. Построение алфавитно-частотного словаря для заданного текста
3.13	Стеки. Анализ правильности скобочного выражения. Вычисление арифметического выражения, записанного в постфиксной форме. Очереди. Использование очереди для временного хранения данных
3.14	Алгоритмы на графах. Построение минимального остовного дерева взвешенного связного неориентированного графа. Количество различных путей между вершинами ориентированного ациклического графа. Алгоритм Дейкстры
3.15	Деревья. Реализация дерева с помощью ссылочных структур. Двоичные (бинарные) деревья. Построение дерева для заданного арифметического

Код	Проверяемый элемент содержания
	выражения. Рекурсивные алгоритмы обхода дерева. Использование стека и очереди для обхода дерева
3.16	Динамическое программирование как метод решения задач с сохранением промежуточных результатов. Задачи, решаемые с помощью динамического программирования: вычисление рекурсивных функций, подсчёт количества вариантов, задачи оптимизации
3.17	Понятие об объектно-ориентированном программировании. Объекты и классы. Свойства и методы объектов. Объектно-ориентированный анализ. Разработка программ на основе объектно-ориентированного подхода. Инкапсуляция, наследование, полиморфизм
4	Информационные технологии
4.1	Анализ данных. Основные задачи анализа данных: прогнозирование, классификация, кластеризация, анализ отклонений. Последовательность решения задач анализа данных: сбор первичных данных, очистка и оценка качества данных, выбор и (или) построение модели, преобразование данных, визуализация данных, интерпретация результатов. Программные средства и Интернет-сервисы для обработки и представления данных. Большие данные. Машинное обучение
4.2	Анализ данных с помощью электронных таблиц. Вычисление суммы, среднего арифметического, наибольшего (наименьшего) значения диапазона. Вычисление коэффициента корреляции двух рядов данных. Построение столбчатых, линейчатых и круговых диаграмм. Построение графиков функций. Подбор линии тренда, решение задач прогнозирования. Решение задач оптимизации с помощью электронных таблиц

3. Тематическое планирование

№ п/п	Наименование разделов и тем программы	Количество часов			Электронные (цифровые) образовательные ресурсы
		Всего	Контрол ьные работы	Практич еские работы	
Модуль №1 «Простая физическая сцена» - 7 часов					
1.1	Простая физическая сцена	6			
1.2	Самостоятельная работа «Простая физическая сцена»	1		1	
Модуль №2 «Система со столкновениями» - 5 часов					

2.1	Система со столкновениями	4			
2.2	Самостоятельная работа «Система со столкновениями»	1		1	
Модуль №3 «Механическая система» - 5 часов					
3.1	Механическая система	4			
3.2	Самостоятельная работа «Механическая система»	1		1	
Модуль №4 «Создание комплексной физической симуляции» - 16 часов					
3.1	Интеграция с Pygame	15			
3.2	Аттестационная работа «Создание комплексной физической симуляции»	1		1	

5. Приложения к программе

ПОУРОЧНОЕ ПЛАНИРОВАНИЕ

№ п/п	Наименование разделов и тем программы	Количество часов			Электронны е (цифровые) образовател ьные ресурсы
		Всего	Контрольные работы	Практические работы	
Модуль №1 «Простая физическая сцена» - 7 часов					
1.1	Введение в физическое моделирование и программирова ние	1			
1.2	Первые шаги в PyMunk:	1			

	создание пространства				
1.3	Геометрические примитивы: сегменты и круги	1			
1.4	Динамические тела: основные свойства	1			
1.5	Полигоны и составные фигуры	1			
1.6	Силы в природе и программе	1			
1.7	Самостоятельная работа 1: Простая физическая сцена	1		1	
Модуль №2 «Система со столкновениями» - 5 часов					
2.1	Основы столкновений: обнаружение коллизий	1			
2.2	Упругость и трение: свойства материалов	1			
2.3	Обработчики столкновений	1			
2.4	Простые механизмы: шарнирные соединения	1			
2.5	Самостоятельная работа 2: Система со	1		1	

	столкновениям и				
Модуль №3 «Механическая система» - 5 часов					
3.1	Подвижные соединения: SlideJoint	1			
3.2	Направляющие соединения: GrooveJoint	1			
3.3	Пружины и демпферы	1			
3.4	Создание сложных механических систем	1			
3.5	Самостоятельн ая работа 3: Механическая система	1		1	
Модуль №4 «Создание комплексной физической симуляции» - 16 часов					
4.1	Ограничения движений	1			
4.2	Основы визуализации: знакомство с Pygame	1			
4.3	Визуализация физических объектов	1			
4.4	Интерактивное взаимодействие	1			
4.5	Создание физической песочницы	1			
4.6	Моделирование простых	1			

	механизмов				
4.7	Цепные и ременные передачи	1			
4.8	Маятники и колебательные системы	1			
4.9	Моделирование мостовых конструкций	1			
4.10	Основы частиц и жидкостей	1			
4.11	Управление камерой и видом сцены	1			
4.12	Пользовательский интерфейс для симуляции	1			
4.13	Инструменты отладки и диагностики	1			
4.14	Анализ результатов симуляции	1			
4.15	Оптимизация производительности	1			
4.16	Итоговая контрольная работа «Создание комплексной физической симуляции с использованием всех изученных элементов и	1		1	

	принципов.»				
--	--------------------	--	--	--	--

КИМ

Модуль 1 «Простая физическая сцена»

Цель: Закрепить базовые понятия PyMunk: создание пространства, статических и динамических тел, простых форм (отрезки, окружности, полигоны). Освоить применение сил.

Задача: Создать сцену "Падающий хаос", где различные по форме и массе тела падают на платформу и взаимодействуют друг с другом.

Пошаговое руководство:

Инициализация:

Импортируйте необходимые библиотеки (pymunk, pygame или matplotlib для визуализации).

Создайте физическое пространство space и установите гравитацию по вертикали (0, -1000).

Создание статической платформы:

Создайте StaticBody.

Добавьте к нему форму Segment (отрезок), который будет представлять собой пол. Расположите его внизу экрана.

Добавьте форму к пространству.

Создание динамических тел в цикле:

В цикле создайте несколько динамических тел (Body) с случайной массой.

Для каждого тела определите случайную форму:

Вариант 1: Окружность (Circle) со случайным радиусом.

Вариант 2: Прямоугольник (как Poly с вершинами для коробки) со случайными размерами.

Задайте телу случайное начальное положение в верхней части экрана.

Добавьте тело и его форму в пространство.

Применение силы:

Для некоторых тел (например, каждого пятого) примените горизонтальный импульс (apply_impulse_at_local_point) сразу после создания, чтобы они отталкивались от центра.

Визуализация и симуляция:

Организируйте основной цикл, на каждом шаге которого:

Происходит обновление пространства (`space.step(dt)`).
Отрисовываются все тела и формы.

Модуль 2 «Система со столкновениями»

Цель: Освоить работу с обработчиками столкновений, настройку физических свойств материалов (трение, упругость) и создание составных тел.

Задача: Построить башню из блоков и сбить ее шаром, настроив различные материалы для блоков и шара. Реализовать подсчет столкновений шара с блоками.

Пошаговое руководство:

Подготовка сцены:

Создайте пространство, гравитацию и статический пол (как в работе №1).

Создайте дополнительные статические стены по бокам экрана, чтобы тела не улетали за пределы.

Создание материалов:

Определите два набора свойств для форм (`elasticity`, `friction`):

`material_wood`: низкая упругость (0.2), высокое трение (1.0).

`material_ball`: высокая упругость (0.95), низкое трение (0.1).

Построение башни:

Используя вложенные циклы, постройте башню из прямоугольных динамических тел (блоков). Например, 5 этажей по 3 блока в каждом.

Всем блокам башни назначьте `material_wood`.

Создание "снаряда":

Создайте динамическое тело в виде окружности (`Circle`) с большой массой.

Расположите его слева от башни.

Примените к нему мощный горизонтальный импульс, чтобы он полетел в сторону башни.

Назначьте снаряду `material_ball`.

Обработчик столкновений:

Создайте функцию-обработчик `begin_collision(arbiter, space, data)`.

В функции проверяйте, является ли одно из сталкивающихся тел снарядом, а другое — блоком башни.

При обнаружении такого столкновения увеличивайте счетчик на 1 и выводите его значение в консоль.

Зарегистрируйте этот обработчик для типа столкновения между формой снаряда и формой блока (`space.add_collision_handler()`).

Модуль 3 «Механическая система»

Цель: Научиться создавать сложные механические системы с помощью соединений (PinJoint, GrooveJoint), моделировать упругие свойства (пружины) и управлять ими.

Задача: Сконструировать катапульту с упругим элементом и мишенью, соединенной с основанием скользящим соединением.

Пошаговое руководство:

Основание и стойка:

Создайте массивное статическое тело как основание.

Создайте статическое тело в виде высокой стойки, на которой будет крепиться катапульта.

Рычаг катапульти:

Создайте динамическое тело в виде длинного и узкого прямоугольника (рычаг).

Соедините его центр с верхом стойки с помощью PinJoint. Это будет точка крепления.

Упругий элемент:

На основании, под рычагом, создайте еще одно статическое тело (крюк или проушину).

Соедините этот крюк с дальним концом рычага с помощью DampedSpring.

Настройте параметры пружины: `rest_length` (длина, к которой она стремится), `stiffness` (жесткость), `damping` (демпфирование). Подберите значения так, чтобы рычаг резко взводился.

Снаряд и "ложе":

На дальнем конце рычага создайте небольшой статический выступ (или используйте саму форму рычага) как ложе для снаряда (шара).

Снаряд создается как динамическое тело с округлой формой.

Мишень со скользящим соединением:

Создайте динамическое тело-мишень (например, прямоугольник).

Создайте статическое тело-направляющую над основанием, используя GrooveJoint.

Задайте горизонтальную линию, по которой может двигаться мишень.

С помощью PinJoint соедините мишень с бегунком, который скользит по этой направляющей. Это позволит мишени двигаться только горизонтально.

Взаимодействие:

Реализуйте механизм "взвода" катапульти (например, симуляцией клика мыши, который применяет силу, оттягивающую рычаг вниз, а затем отпускание).

Снаряд должен вылетать и сбивать мишень, которая отлетает строго горизонтально.

Модуль 4 «Создание комплексной физической симуляции»

Цель работы: Закрепить и продемонстрировать на практике полученные знания и навыки работы с библиотекой PyMunk, умение создавать комплексные физические симуляции, сочетающие различные изученные элементы: тела, формы, соединения, взаимодействия, а также визуализировать результат с помощью Pygame.

Постановка задачи:

Разработать интерактивное приложение на Python, которое:

1. Реализует физическую модель согласно выданному варианту.
2. Использует не менее 3-х различных изученных элементов (разные типы тел, формы, джойнты, свойства материалов).
3. Визуализирует модель с помощью библиотеки Pygame.
4. Содержит простые элементы интерактивности (возможность добавления объектов, запуска механизма, изменения параметров).
5. Работает стабильно и демонстрирует правдоподобное физическое поведение.

Рекомендации по выполнению:

1. Выделите ключевые объекты и взаимодействия, которые нужно смоделировать.
2. Разбейте сложную модель на простые части (например, сначала создайте опоры, потом подвижные элементы, потом соедините их).
3. Сначала добейтесь работы базовой физики, потом добавляйте детали и интерактивность. Часто тестируйте результат.
4. Меняйте параметры (массу, трение, упругость), чтобы добиться желаемого поведения.
5. Оставляйте комментарии в коде, какой частью модели отвечает тот или иной блок.

Варианты заданий

Вариант 1: Маятниковые часы

Создайте работающие маятниковые часы с маятником (PivotJoint), шестеренками (круги, соединенные PinJoint) и стрелками. Реализуйте запуск маятника и его постепенное затухание.

Вариант 2: Подъемный кран

Смоделируйте башенный кран с поворотной платформой, стрелой и грузом на тросе (используйте SlideJoint или GrooveJoint для троса). Реализуйте подъем, опускание груза и поворот крана.

Вариант 3: Настольный футбол

Создайте игровое поле с воротами, мячом и несколькими штангами с игроками (PivotJoint), которые можно двигать влево-вправо. Реализуйте подсчет голов.

Вариант 4: Качели-балансир

Создайте качели из доски (SegmentShape) и опоры. Реализуйте возможность сажать на разные концы качелей шарики разной массы и наблюдать за балансировкой.

Вариант 5: Кегельбан

Расставьте кегли (составные тела из кругов и прямоугольников). Создайте шар, который можно запускать с определенной силой, чтобы сбить их. Реализуйте подсчет сбитых кеглей.

Вариант 6: Гусеничная платформа

Соберите простую модель гусеницы из множества соединенных PivotJoint траков. Создайте платформу, которая на них стоит. Реализуйте движение платформы с помощью вращения колес.

Вариант 7: Падающее домино

Создайте сложную цепь костяшек домино, которая включает не только прямую линию, но и повороты, лестницы и другие фигуры. Реализуйте запуск цепи падения.

Вариант 8: Сортировка шаров

Постройте конструкцию из наклонных плоскостей и желобов, которая сортирует падающие шары по размеру или цвету (массе), направляя их в разные отсеки.

Вариант 9: Рыцарский замок с разводным мостом

Смоделируйте замок с башнями и разводным мостом. Мост должен подниматься и опускаться с помощью системы цепей (SlideJoint) и противовеса.

Вариант 10: Американские горки

Спроектируйте и постройте трек для американских горок (используйте SegmentShape для рельс). Запустите по нему вагонетку, которая должна пройти весь путь до конца.

Вариант 11: Дверь с доводчиком

Смоделируйте дверь на петлях (PivotJoint), которая автоматически плавно закрывается с помощью пружинного механизма (DampedSpring). Добавьте возможность открывать кк мышью.

Вариант 12: Балансирующая птица-неваляшка

Создайте неваляшку (округлое тело с низким центром тяжести) и наклонную плоскость. Покажите, как неваляшка всегда возвращается в вертикальное положение.

Вариант 13: Метательный механизм (требушет)

Соберите простую модель катапульты или требушета с рычагом (PivotJoint) и пращей. Реализуйте возможность менять силу броска и угол выстрела.

Вариант 14: Фонтан

Смоделируйте фонтан, в котором насос (источник силы) постоянно подбрасывает вверх частицы-капли воды. Реализуйте их падение и столкновение с чашей фонтана.

Вариант 15: Система шкивов

Создайте механизм из нескольких шкивов (кругов, закрепленных PivotJoint) и веревки (цепочки SlideJoint), который позволяет поднимать груз с помощью меньшего усилия.