

КОМИТЕТ ОБРАЗОВАНИЯ ГАТЧИНСКОГО МУНИЦИПАЛЬНОГО РАЙОНА
ЛЕНИНГРАДСКОЙ ОБЛАСТИ
МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
"ГАТЧИНСКИЙ ЦЕНТР НЕПРЕРЫВНОГО ОБРАЗОВАНИЯ "ЦЕНТР ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ"
188300, г. Гатчина Ленинградской обл., ул. Рошинская, 8, тел/факс (881371) 43296

ПРИНЯТА:
на заседании Педагогического Совета
МБОУ ДО «ГЦНО «ЦИТ»
протокол № 1
от «03» 09 2019 г.

УТВЕРЖДАЮ:
Директор
МБОУ ДО «ГЦНО «ЦИТ»
А.Э. Морослип.
Приказ № 33 от «03» 09 2019 г.



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ
ПРОГРАММА ТЕХНИЧЕСКОЙ НАПРАВЛЕННОСТИ**

**«Программирование игр различных типов: аркадные,
онлайн-стратегия, qwesty, пошаговая стратегия»**

Категория слушателей: *учащиеся 8-9 классов*
Организация обучения: *очная*
Срок обучения: *72 часа*
Разработчик программы: *Алексеев П.И., преподаватель*

Гатчина

2019

Пояснительная записка

Дополнительная общеразвивающая программа технической направленности «Разработка игр» разработана на основе:

- Федерального закона от 29 декабря 2012 года №273-ФЗ «Об образовании в Российской Федерации».
- Концепции развития дополнительного образования детей утверждена распоряжением Правительства Российской Федерации от 04 сентября 2014 года №1726-р
- Приказ Министерства просвещения РФ от 9 ноября 2018 г. N 196 “Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам”

Направленность

Разработка компьютерных игр — интересная ученикам отрасль мультимедиа на стыке искусства и информатики. Благодаря этому легче усваиваются правильные подходы в программировании, а так же безболезненно происходит знакомство с современными средами разработки. Приобретаемые обучающимися навыки позволят им применять свои знания при разработке любого программного обеспечения, не только из сферы компьютерных игр.

Данная программа имеет техническую направленность.

Актуальность

Программа курса способствует развитию навыков программирования, проектирования программного обеспечения, а также помогает развитию пространственного мышления, аналитической геометрии, креативности, умению доводить работу до конца.

Новизна

Программа курса построена на современных представлениях о программировании и проектировании программного обеспечения, в частности о разработке игр.

Цель

Развитие творческой, активно действующей личности, формирование представления о правильном процессе разработки, изучение принципов программирования и проектирования, знакомство со средами разработки.

Задачи

Обучающие:

- сформировать представление о процессе разработки ПО;
- изучить принципы программирования и проектирования;
- познакомится с современными средами разработки;
- развить интерес и положительный настрой к программированию;

Развивающие:

- повысить мыслительную активность учащихся и углубить навыки логического и творческого мышления при работе над созданием компьютерных игр;
- развить умение искать и отбирать необходимую информацию с использованием интернет-ресурсов;
- развить индивидуальные способности учащихся, их самостоятельность, потребность в самообразовании;
- развить навыки самостоятельной работы.

Воспитательные:

- воспитать трудолюбие и чувство ответственности;
- проявить навыки самостоятельной работы;
- наработать навыки сотрудничества при решении коллективной задачи.

Педагогическая целесообразность

Необходимость разработки данной программы обусловлена, с одной стороны, пересмотром содержания общего образования в целом, с другой стороны, развитием информационных технологий и, следовательно, методов разработки программного обеспечения. Продуктивным является использование интересной ученикам темы и межпредметных связей с другими предметами, прежде всего с физикой и аналитической геометрией. Программа допускает возможность корректировки и видоизменения тематического содержания в процессе обучения.

В ходе обучения, учащиеся будут реализовывать отдельные игровые проекты. Метод проектов, с одной стороны, выступает как педагогическая технология, а с другой стороны, как организация деятельности обучающихся, построенная по принципу инженерного проектирования и межпредметных связей. Основой использования метода проектов в технологическом образовании школьников являются общепедагогические и дидактические принципы:

- связь теории с практикой
- научность, сознательность и активность усвоения знаний
- доступность, систематичность и преемственность обучения
- наглядность и прочность усвоения знаний.

Программа *педагогически целесообразна*, так как активизирует творческую деятельность, углубляет знания, умения и навыки, полученные на уроках.

Организационно-педагогические условия реализации образовательной программы

Сроки реализации:	1 учебный год, 72 часа.
Возраст обучающихся:	13–16 лет
Состав учебной группы:	постоянный
Форма обучения:	очная
Наполняемость группы:	10 человек
Продолжительность одного занятия:	45 минут
Объем нагрузки в неделю:	2 занятия по 45 минут, включая 10-минутную паузу на проветривание кабинета и на проведение гимнастики для глаз

Учебный план

Тема	Количество часов			Форма проведения промежуточной аттестации
	Всего	Теория	Практика	
Разработка игры Finger Battle.	2	1	1	
Doodle Jump. Простая аркада.	10	3	7	
Bullet Hell. Более сложная аркада.	16	4	12	
Ragdoll-физика. Система частиц и связей.	2	1.5	0.5	
Agar.io. Многопользовательская игра.	16	6	10	
Системы контроля версий.	2	1	1	
Platformer. Сложная расширяемая игра.	24	10	14	
Итого	72	26.5	45.5	

Содержание учебного курса:

1. Разработка игры Finger Battle – 2 часа

Знакомство со средой, отрисовка примитивов, обработка событий клавиатуры

2. Doodle Jump – 10 часов

Составление плана. Мозговой штурм по фичам. ООП. Класс игрока. Простейшая физика.

Управление игроком. Прыжки. Платформы.

Процедурная генерация уровня. Столкновение с платформами.

Камера, прокрутка экрана.

Увеличение сложности. Стартовый экран. Условие проигрыша. Новая игра.

3. Bullet Hell – 16 часов

Составление плана. Мозговой штурм по фичам. Класс игрока. Управление мышкой.

Стрельба. Класс пуль.

Камера. Прокручивающийся фон с эффектом параллакса.

Противники. Наследование. Класс противника. Уничтожение пуль за экраном.

Столкновения пуль, жизнь, взрывы.

Разные типы стрельбы. Свойства противников. Разные типы противников. Анимированные взрывы.

Загрузка уровней. Чтение уровня. Редактор уровней.

Проигрыш, переход на следующий уровень, босс на каждом уровне.

4. Ragdoll-физика. Система частиц и связей – 2 часа

Простейшая физика на основе частиц и связей между ними. Реалистично падающее и отскакивающее от поверхности квадратное упругое физическое тело.

5. Agar.io. Многопользовательская игра. – 16 часов

Введение в многопользовательские игры. Менеджер состояний. Управление игроком.

Общение с сокет-сервером.

Добавление новых и удаление вышедших игроков.

Скины, имена, разный радиус.

Еда. Съедание игроков. Камера.

Масштаб у камеры. Фиксированная скорость (нормализация вектора). Багфикс.

Завершение игры, разбор деталей.

6. Системы контроля версий – 2 часа

Введение в систему контроля версий, git, SourceTree.

7. Платформер – 24 часа

Система состояний, меню, игрок.

Уровни, загрузка уровней. Класс игрока. Столкновения.

Столкновения.

Столкновения. Управление игроком. Плавная камера.

Прыжок от земли. Стрельба.

Рефакторинг: GameObject. Стрельба и взрывы.

Скорость стрельбы. Столкновения с GameObject-ами.

Смена оружия. Жизнь игроков.

Искусственный интеллект противников.

Уровни. Аптечка и другие предметы.

Переход на следующий уровень. Монетки.

Уровни. Плейтест.

Средства обучения

Программное обеспечение и инструменты: ОС MS WINDOWS, Visual Studio Code, MS Paint, Adobe Photoshop или аналог, Typescript, PixiJS, Node.js, Parcel bundler, Yarn.

Материально-техническое обеспечение: компьютерный класс на 10 рабочих мест с выходом в интернет, мультимедийный проектор, или телевизор.

Кадровое обеспечение: методист и преподаватель, владеющий компьютерными технологиями и языками программирования.

Обязательная предварительная подготовка к курсу:

Уверенный пользователь ПК, знакомство с основами какого-либо языка программирования. Знать основы логики из курса информатики. Желательно иметь навыки работы в графических редакторах и поисковых системах Интернет.

Планируемые результаты освоения образовательной программы.

В результате изучения курса обучающиеся должны:

- знать основные алгоритмические конструкции и структуры данных, уметь использовать их для построения алгоритмов;
- понимать основы объектно-ориентированного языка программирования;
- понимать базовые принципы проектирования программного обеспечения;
- понимать, как устроена разработка игр на всех стадиях;
- уметь применять полученные знания, умения и навыки в проектных работах.

Предполагается, что в результате изучения курса учащиеся могут:

- создавать игровые приложения под платформу Web на языке TypeScript с использованием графического движка PixiJS;
- использовать основные приёмы создания программ;

Система оценки результатов освоения образовательной программы

(Контроль освоения образовательной программы)

Для **текущего контроля** используются задания по каждому уроку. Урок считается усвоенным, если ученик подготовил заданное приложение (программу) и продемонстрировал работу программы на занятии.

Для **промежуточной аттестации** используется подборка задач разных уровней по каждой теме. Тема считается усвоенной, если ученик выполнил зачетное задание (проект) и продемонстрировал работу программы.

Итоговая аттестация (реализации данной дополнительной программы): Подготовка проекта, оформление и защита проекта или исследовательской работы на Фестивале ЦИТа.

Средства обучения

Перечень оборудования (инструменты, материалы и приспособления).

Персональный компьютер	1
Мультимедийный проектор	1
Устройство для зашторивания окон	1

Перечень технических средств обучения.

Маркерная доска	1
Маркер для доски	1
Губка для доски	1
Парты	10
Стулья	10

Календарный учебный график

	Начало	Окончание	Продолжительность (количество учебных недель)
I полугодие	02.09.2019	28.12.2019	17
II полугодие	09.01.2020	30.05.2020	19

Занятия проводятся согласно учебного плана 1 раз в неделю.

Место и время проведения занятий соответствует расписанию, утвержденному директором.

Праздничные дни:

День народного единства – 4 - 6 ноября;

Международный женский день - 8 марта;

Праздник весны и труда –1 мая;

День Победы – 9 мая.

Каникулы:

1-8 января;

Календарно-тематическое планирование

<i>№ урока</i>	<i>Темы занятий</i>	<i>Материал</i>	<i>Дата проведения</i>
1	Вводное занятие	Инструктаж по технике безопасности и безопасному интернету.	
2	Finger Battle	Разработка игры Finger Battle: знакомство со средой, отрисовка примитивов, обработка событий клавиатуры	
3	Doodle Jump	Составление плана. Мозговой штурм по фичам. ООП. Класс игрока. Простейшая физика.	
4	Doodle Jump	Управление игроком. Прыжки. Платформы.	
5	Doodle Jump	Процедурная генерация уровня. Столкновение с платформами.	
6	Doodle Jump	Камера, прокрутка экрана.	
7	Doodle Jump	Увеличение сложности. Стартовый экран. Условие проигрыша. Новая игра.	
8	Bullet Hell	Составление плана. Мозговой штурм по фичам. Класс игрока. Управление мышкой.	
9	Bullet Hell	Стрельба. Класс пуль.	
10	Bullet Hell	Камера. Прокручивающийся фон с эффектом параллакса.	
11	Bullet Hell	Противники. Наследование. Класс противника. Уничтожение пуль за экраном.	

12	Bullet Hell	Столкновения пуль, жизнь, взрывы.	
13	Bullet Hell	Разные типы стрельбы. Свойства противников. Разные типы противников. Анимированные взрывы.	
14	Bullet Hell	Загрузка уровней. Чтение уровня. Редактор уровней. Работа над ошибками.	
15	Bullet Hell	Проигрыш, переход на следующий уровень, босс на каждом уровне.	
16	Физика в играх	Ragdoll-физика. Система частиц и связей.	
17	Agar.io	Введение в многопользовательские игры. Менеджер состояний. Управление игроком.	
18	Agar.io	Общение с сокет-сервером.	
19	Agar.io	Добавление новых и удаление вышедших игроков.	
20	Agar.io	Скины, имена, разный радиус.	
21	Agar.io	Еда. Съедание игроков. Камера.	
22	Agar.io	Масштаб у камеры. Фиксированная скорость (нормализация вектора). Багфикс.	
23	Agar.io	Завершение игры, разбор деталей.	
24	Системы контроля версий	Введение в систему контроля версий, git. SourceTree. План новой игры.	
25	Platformer.	Система состояний, меню, игрок.	

26	Platformer.	Уровни, загрузка уровней. Класс игрока. Столкновения.	
27	Platformer.	Столкновения.	
28	Platformer.	Столкновения. Управление игроком. Плавная камера.	
29	Platformer.	Прыжок от земли. Стрельба.	
30	Platformer.	Рефакторинг: GameObject. Стрельба и взрывы.	
31	Platformer.	Скорость стрельбы. Столкновения с GameObject-ами.	
32	Platformer.	Смена оружия. Жизнь игроков.	
33	Platformer.	Искусственный интеллект противников.	
34	Platformer.	Уровни. Аптечка и другие предметы.	
35	Platformer.	Переход на следующий уровень. Монетки.	
36	Platformer.	Уровни. Плейтест.	