

РАЗРАБОТКА ПРОГРАММЫ «ИГРА АРКАНОИД» НА ЯЗЫКЕ PYTHON

А.П. Жолбин^а, студент гр. 17В11

Научный руководитель: Разумников С.В.^б, кандидат технических наук, доцент ЮТИ ТПУ

Юргинский технологический институт (филиал)

Национального исследовательского Томского политехнического университета

652055, Кемеровская обл., г. Юрга, ул. Ленинградская, 26

E-mail: ^аpszholbin@gmail.com, E-mail: ^бdemolove7@inbox.ru

Аннотация: В качестве темы для статьи была выбрана игра «Арканойд». Актуальность этой темы заключается в том, что на сегодняшний день компьютерные игры занимают важное место в жизни человека. Они могут использоваться в качестве развлекательного времяпрепровождения или в обучающих целях, а профессия разработчика компьютерных игр является перспективной и хорошо оплачиваемой. Именно поэтому важно понять процесс разработки игры и усовершенствовать свои навыки в создании алгоритмов и программировании.

Ключевые слова: алгоритм, язык программирования, среда программирования, библиотека, метод, функция, переменная, условие, python, блок-схема, разработка

Annotation: The Arkanoid game was chosen as the topic for the article. The relevance of this topic lies in the fact that today computer games occupy an important place in human life. They can be used as an entertaining pastime or for educational purposes, and the profession of a computer game developer is promising and well-paid. That is why it is important to understand the game development process and improve your skills in creating algorithms and programming.

Keywords: algorithm, programming language, programming environment, library, method, function, variable, condition, python, flowchart, development

Арканойд (англ. Arkanoid) – компьютерная игра, уходящая корнями в системы середины 1980-х годов. Игровой процесс представляет собой управление платформой, от которой отталкивается шарик. Цель игры состоит в том, чтобы платформой отбивать шарик так, чтобы он попадал в кирпичи на поле, ломал их. Уровень засчитывается пройденным, если сломаны все кирпичи. Игра считается проигранной, если шарик упадет в нижнюю часть экрана несколько раз и кончатся доступные игроку жизни.

Для достижения поставленной цели были поставлены следующие задачи: изучить возможности и библиотеки Python, предназначенные для разработки компьютерных игр, написать код программы на языке Python, реализующий игру «Арканойд», произвести отладку кода путем тестирования, устранить найденные ошибки.

Теоретическое назначение – программа может быть использована в учебных целях. Область применения – на занятиях в высших учебных заведениях.

Практическое назначение – программа может использоваться для игрового времяпрепровождения. Область применения – развлекательная или развивающая.

Научное назначение – программа может быть использована для разработки и проверки стратегий в игре «Арканойд». Область применения – научная деятельность, связанная с влиянием размеров игровых элементов на скорость реакции и обработки информации человеком.

Готовая программа содержит игровое поле размером 500x500 пикселей, условно разделенное на две части – часть с платформой и часть с кирпичами. Цвет заднего фона сделан монотонным и простым, чтобы игрок не отвлекался на него и не терял на его фоне шарик, платформу и кирпичи.

Платформа представляет из себя прямоугольный объект с яркой текстурой, чтобы её можно было легко заметить и сложно потерять из виду. Управление платформой производится мышкой, чтобы можно было быстро переместить её из одного края экрана в другой. Передвижение платформы ограничено по горизонтальной оси Y, а также по вертикали невидимыми «стенками», совпадающими с границами окна приложения, чтобы было невозможно её вывести за пределы поля зрения игрока.

Помимо платформы, игрок непрямым образом управляет шариком, свободно двигающемуся по игровому полю, отскакивая от поверхностей по физическому закону отражения света в геометрической оптике (угол падения равен углу отражения). В данном случае как поверхности рассматриваются все боковые и верхняя границы окна приложения, платформу и кирпичи. Нижнюю границу окна не следует делать поверхностью отскока шарика, иначе игрок станет наблюдателем, а игра будет играть в саму себя бесконечно.

Каждый раз, когда шарик попадает по кирпичу, тот исчезает после отскока шарика, вместе с этим игроку прибавляется определенное количество очков, зависящее от режима. Сложность игры зависит от выбранного режима и от скорости движения шарика, которая будет расти вместе с набранным количеством очков – скорость будет увеличиваться с шагом в 0.5 пикселей в секунду за каждую тысячу набранных очков, начальная скорость 3 пикселя в секунду.

В верхней части окна во время игры доступна статистика игрока, на которой будет показываться текущий и максимальный в данном режиме счёт, количество режимов и порядковый номер уровня.

Для выполнения поставленной задачи создано окно с игрой, реализован графический интерфейс, включающий в себя стартовый экран с выбором режима и саму игру, разработан алгоритм передвижения платформы и шарика, создана функция, предусматривающая случайную генерацию кирпичей на игровом поле.

В программе используются три подключаемые библиотеки: `pygame`, `random` и `sys`. Стандартные функции, используемые в программе: `isinstance` возвращает булево значение `True`, если переданный ей объект является экземпляром указанного класса, `len` возвращает длину списка, `range` используется в циклах с параметром.

Модуль `random` позволяет генерировать случайные числа. Метод `random.randint` возвращает случайное целое число из заданного диапазона. В программе используется для разброски яблок по полю.

Модуль `sys` позволяет обеспечивать доступ к переменным и функциям, связанным с интерпретатором Python. Метод `sys.exit` используется для корректного завершения работы программы.

Библиотека `Pygame` представляет собой набор инструментов, позволяющих создавать игры. Данный модуль построен на основе библиотеки `SDL`, которая обеспечивает простой доступ к звуковым и визуальным элементам. Методы `pygame.init` и `pygame.quit` используются соответственно для инициализации модуля `Pygame` и завершения его работы. Метод `Pygame.display.setmode` позволяет создать окно игры, в качестве параметров принимает кортеж, состоящий из двух чисел, определяющих ширину и высоту окна. Метод `pygame.display.set_caption` устанавливает заголовок окна, `pygame.display.update` используется для обновления экрана, `blit` позволяет отображать текст на экране. `Pygame.font.SysFont` создаёт объект шрифт. `Pygame.draw.rect` и `pygame.draw.circle` используются для рисования прямоугольников и кругов соответственно. `Pygame.event.get` позволяет получить все события из очереди игровых событий, `pygame.QUIT` используется для закрытия игрового окна, `pygame.K_1` предназначен для обработки нажатий клавиш с клавиатуры. Метод `fill` используется для заливки экрана выбранным цветом.

Описание алгоритма работы программы: нарисовать игровое поле, платформу, шарик, счётчик.; Сгенерировать случайным образом кирпичи на игровом поле с случайными спрайтами из библиотеки; Управление платформы при помощи мышки; Если шарик упал в нижнюю часть окна приложения, вычесть одну жизнь; Если количество жизней равно нулю, вернуться в главное меню; Если шарик столкнулся с поверхностью, оттолкнуть шарик; Если данная поверхность кирпич, удалить его и прибавить очки; Если игрок получил количество очков кратное тысяче, увеличить скорость шарика на 0.5; Если на экране больше не осталось кирпичей, начать цикл с первой позиции. Блок-схема алгоритма работы основной программы приведена ниже на рисунке 1.

XIV Всероссийская научно-практическая конференция
для студентов и учащейся молодежи
«Прогрессивные технологии и экономика в машиностроении»

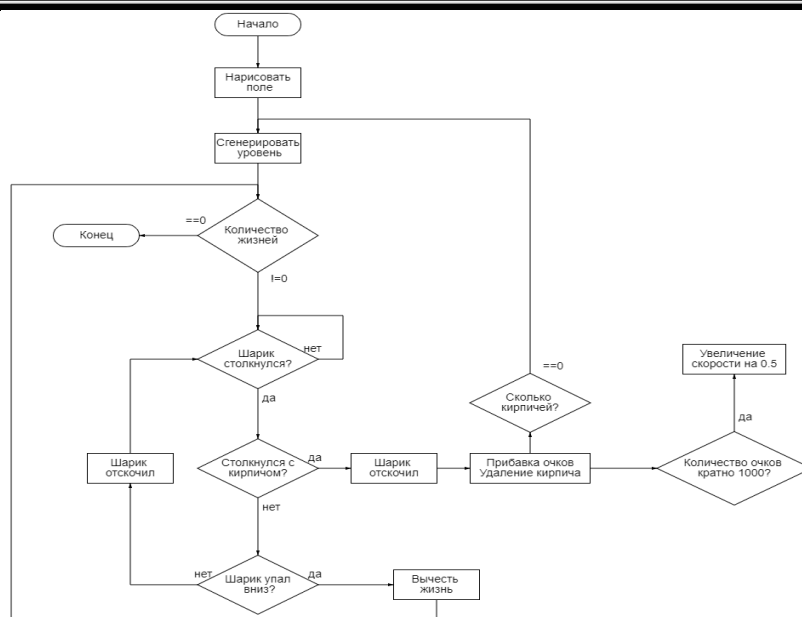


Рис. 1. Блок-схема основной программы

Список используемых источников:

1. Разумников С.В. Теория алгоритмов: методические указания к выполнению курсовой работы для студентов очной формы обучения, обучающихся по направлению 09.03.03 «Прикладная информатика» / С.В. Разумников – Юрга: Изд-во Юргинского технологического института (филиала) Томского политехнического университета, 2022. – 20 с.
2. Буйначев, С.К. Основы программирования на языке Python: учебное пособие / С.К. Буйначев, Н.Ю. Боклаг. – Екатеринбург: Изд-во Урал, ун-та, 2014. – 91, [1] с.
3. Егоров А. Python. Обучение программированию: Канал на Youtube – Электронный ресурс. – URL : <https://www.youtube.com/c/egoroffchannel>. (дата обращения 14.01.2023)
4. Sloan Kelly. Python, PyGame and Raspberry Pi Game Development / S. Kelly – Canada: Niagara Falls, Ontario, 2016. – 198 p.

HR-БЕНЧМАРКИНГ КАК ИНСТРУМЕНТ РАЗВИТИЯ КОМПАНИИ

Д.А.Сабаев², ученик 11 кл.

Научный руководитель: Полицинская Е.В. ^{1а}, к.пед.н., доц.,
¹Юргинский технологический институт (филиал)

Национального исследовательского Томского политехнического университета,
652055, Кемеровская обл., г. Юрга, ул. Ленинградская, 26

²МБОУ «Лицей города Юрги», 652055, Кемеровская обл., Юрга, ул. Кирова, 7а
E-mail: ^akaty031983@mail.ru

Аннотация: Быстрое и постоянное изменение рынков требует реформирования деятельности предприятий. HR-бенчмаркинг – это ориентир, который помогает выбрать правильный путь в развитии вовлечённости и лояльности персонала. Сравнительный анализ персонала помогает повысить эффективность процессов и политику управления персоналом в организации.

Ключевые слова: сравнительный анализ, кадровый бенчмаркинг, конкурентоспособность организации.

Abstract: The rapid and constant change in markets requires reforming the activities of enterprises. HR benchmarking is a benchmark that helps you choose the right path in developing employee engagement and loyalty. Benchmarking of personnel helps to improve the efficiency of processes and personnel management policies in an organization.