

Введение

Курсовой проект посвящен разработке программной реализации популярной логической головоломки "Головоломка Хоши". Данная игра представляет собой захватывающую головоломку, в которой игроку необходимо заполнить поле соответствующими символами, руководствуясь определенными правилами.

Основная цель данного проекта заключается в создании приложения, которое позволит пользователям познакомиться с "Головоломкой Хоши", а также предоставит возможность для развития логического мышления, внимательности и навыков решения сложных задач.

Разрабатываемое приложение ориентировано на широкий круг пользователей. Оно будет интересно как любителям логических головоломок, так и тем, кто хочет провести время с пользой для своего интеллектуального развития.

Далее приведём краткое описание разделов пояснительной записки.

Первый раздел носит название «Анализ задачи». В нем вы сможете ознакомиться с постановкой задачи, которая включает в себя: исследование предметной области поставленной задачи, определение её организационно-экономической сущности. Также в этом разделе вы сможете узнать о том, как данная задача решается в настоящее время. Все входные и выходные данные тоже будут описаны в первом разделе.

В подразделе «Инструменты разработки» будет рассмотрена среда, в которой создаётся данный курсовой проект. Здесь также будут установлены минимальные и оптимальные требования к аппаратным характеристикам, обеспечивающим правильное функционирование поставленной задачей.

В разделе «Проектирование задачи» будут рассмотрены основные аспекты разработки программного продукта. Здесь можно будет узнать об организации данных в контексте среды разработки. В данном разделе будет чётко описан пользовательский интерфейс, составлены алгоритмы процесса обработки информации, описана разработка системы справочной информации.

«Реализация задачи» – это третий раздел пояснительной записки, в котором описываются все элементы и объекты, которые будут использованы при реализации данного приложения. В этом разделе будут чётко описаны функции пользователя и их структура. Здесь можно будет найти таблицу, в которой будет представлена полная аннотация файлов, используемых в данном проекте.

Четвёртый раздел – «Тестирование». В нем будет описано полное и функциональное тестирование данной программы, т.е. будет оттестирован каждый пункт меню, каждая операция, которая выполняется приложением. Будут смоделированы все возможные действия пользователя при работе с программой, начиная от запуска до выхода.

					КП 2-40 01 01.35.40.11.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		4

В разделе «Применение» будет описано назначение, область применения, среда функционирования курсовой программы. Также в нем будет описано использование справочной системы.

«Заключение» будет содержать краткую формулировку задачи, результаты проделанной работы, описание использованных методов и средств, описание степени автоматизации процессов на различных этапах разработки.

В «Литературе» будет приведён список используемых при разработке источников.

В приложениях к пояснительной записке будет приведён листинг программы с необходимыми комментариями.

Схема работы системы будет представлена в графической части.

					КП 2-40 01 01.35.40.11.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		5

1 Анализ задачи

1.1 Постановка задачи

Тема данного курсового проекта заключается в разработке познавательного-интеллектуального приложения "Головоломка Хоши" для персональных компьютеров.

Основная цель создания этого программного продукта - предоставить пользователям возможность познакомиться с уникальной логической головоломкой "Головоломка Хоши", изучить ее правила и принципы, а также развивать логическое мышление, внимательность и навыки решения сложных интеллектуальных задач.

Предметная область проекта будет основана на детальном исследовании "Головоломки Хоши", ее истории, особенностей и методов решения, с целью создания максимально реалистичного и познавательного игрового опыта для пользователей.

Разрабатываемое приложение не будет иметь ограничений по частоте использования. Пользователи смогут играть в "Головоломку Хоши" в любое удобное для них время, будь то кратковременные игровые сессии или длительные погружения в решение головоломок.

Для создания приложения необходимо использовать максимально полную, достоверную и точную информацию о "Головоломке Хоши" из соответствующих источников. Разрабатываемый программный продукт позволит пользователям:

- играть в логическую головоломку "Головоломка Хоши";
- изучать различные стратегии и методы решения головоломок;
- развивать логическое мышление, внимательность и навыки решения сложных интеллектуальных задач.

Приложение будет иметь интуитивно понятный и эргономичный графический интерфейс для игрового процесса, включая удобные средства управления, отображения игрового поля, а также дополнительные. Для реализации проекта будут использованы различные типы данных, включая вводимые пользователем значения, результаты игры, текстовые и мультимедийные ресурсы.

1.2 Инструменты разработки

Для разработки проекта выбрана среда Delphi (Delphi 12), так как это среда объектно-ориентированного программирования, относящаяся к классу RAD – (Rapid Application Development – «Средство быстрой разработки приложений»), реализованная на Object Pascal. Используется для разработки визуализированного представления программного обеспечения.

					КП 2-40 01 01.35.40.11.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		6

Выгоды от проектирования в среде Windows с помощью Delphi 12:

- устраняется необходимость в повторном вводе данных;
- обеспечивается согласованность проекта и его реализации;
- увеличивается производительность разработки и переносимость программ.

Также для разработки программы необходимы:

- Help & Manual – редактор для создания справок;
- Ithernet-браузер – необходим для нахождения информации и фоновых изображений;

– Smart Install Maker: Удобный инструмент для создания инсталляторов. Обладает полным набором необходимых функций для создания профессиональных инсталляторов с минимальным размером и высокой степенью сжатия файлов;

- Adobe Photoshop 2021 – необходим для обработки фотографий;

При разработке данного программного продукта был использован компьютер со следующими характеристиками:

- процессор Intel(R) Core(TM) i3-7100U CPU @ 2.40GHz;
- ОЗУ: 16Gb;
- память: SDD 512Gb;
- ОС – Windows 11.

1.3 Требования к приложению

На этапе исследования предметной области был установлен целый ряд требований, предъявляемых к разрабатываемой игре.

При моделировании форм следует учесть следующие моменты:

- интуитивно понятный интерфейс и управление;
- небольшое количество информации на каждой форме;
- Оптимизация приложения для обеспечения быстрой отзывчивости при манипуляции с игровым полем;
- Минимизация задержек и лагов при переключении между режимами и загрузке новых головоломок;
- небольшие окна формы;
- адаптивность приложения под различные размеры экранов и устройств;
- наличие информации о функционале приложения;
- Стабильная работа приложения без сбоев и ошибок.

Минимальные системные требования для игры "Головоломка Хоши":

1.8 GHz процессор;

1 GB RAM;

От 200 MB доступного места на жестком диске.

					КП 2-40 01 01.35.40.11.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		7

Рекомендуемые системные требования:

Двухъядерный AMD Athlon или эквивалентный процессор Intel (или более производительный);

2 GB RAM;

От 500 MB доступного места на жестком диске.

При разработке интерфейса игры использованы преимущественно зеленые и песочные оттенки. Основные разделы игры доступны с первой страницы. Каждое окно игры имеет ясную визуальную иерархию своих элементов. Разработан понятный пользователю интерфейс, фрагменты текста располагаются на экране так, чтобы пользователь не испытывал дискомфорта в плане восприятия информации.

Ошибки программы вследствие некорректного взаимодействия пользователя при работе с игрой через графический интерфейс не должны влиять на работоспособность. Надежность игры должна быть на высоком уровне, не допуская ситуаций программных ошибок и некорректной работы приложения.

					КП 2-40 01 01.35.40.11.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		8

2 Проектирование задачи

2.1 Организация данных

Проектирование задачи – это очень важный и ответственный этап в разработке любого приложения.

Важным является он вследствие того, что методы, по средствам которых пользователь управляет формами, построены на высокой степени специализации каждого из компонентов.

Необходимым условием при разработке данного приложения является описание организации данных, т.е. логическая и физическая структура данных в контексте среды разработки. В разрабатываемой программе будут использоваться три вида данных.

Первым видом являются данные, которые будут введены разработчиком на этапе реализации задачи. Сюда можно отнести графические изображения, звуковые материалы текстовые данные и исходные коды теории в разделах.

Вторым видом данных, используемых в программе, является вводимая пользователем информация. К входной информации можно отнести вводимые пользователем значения, например выбор элементов на игровом поле, выбор уровня сложности.

Третьим видом данных являются результаты работы приложения - например, Текущее состояние игры (расположение элементов на поле). Результаты игры, такие как количество попыток. Эти данные также относятся к отдельному виду данных, так как они генерируются программой в процессе игры и не вводятся пользователем или разработчиком.

Таким образом, организация данных является важной задачей при разработке данной и любой программы.

2.2 Процессы

Согласно всем перечисленным требованиям и указаниям, главной задачей разрабатываемого приложения "Головоломка Хоши" является обучение пользователя логической игре "Хоши" и предоставление интерактивного игрового опыта.

Для реализации поставленных задач будут использоваться различные программные процедуры и функции. Например, с помощью них будет осуществляться генерация игрового поля, слежение за игровым процессом, проверка правильности решения головоломки.

На этапе проектирования важно уделить должное внимание разработке графического интерфейса приложения. Хотя проектирование интерфейса не является

					КП 2-40 01 01.35.40.11.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		9

принципиальным заранее, тщательно продуманная и реализованная форма улучшает общее восприятие и удобство использования программы.

Тем не менее, хорошо организованная форма улучшает внешний вид программы, поэтому эти вопросы будут более подробно рассмотрены далее.

2.3 Описание внешнего пользовательского интерфейса

Важным при выполнении курсового проекта является организация диалога между пользователем и самой программой. Во многом это зависит от того, как программист разработает данную программу, какие компоненты будут использованы и какие методы будут автоматизированы.

Особое внимание следует уделить интерфейсу. Разработчик должен так организовать внешний вид своей программы, что бы пользователь понял, что от него требуется. При решении задачи будут использованы компоненты для вывода информации, вставки видео и др.

Для организации эффективной работы пользователя нужно создать целостное приложение данной предметной области, в которой все компоненты приложения будут сгруппированы по функциональному назначению. При этом необходимо обеспечить удобный графический интерфейс пользователя.

Таким образом, для успешной работы всего проекта в целом следует обеспечить интуитивно понятный интерфейс с приятной гаммой цветов и шрифтами.

Структура навигации по проекту представлена на рисунке 1:



Рисунок 1 – Навигация между окнами программы

3 Реализация

3.1 Структура программы

Данный курсовой проект содержит 7 модулей. Далее рассмотрим назначение каждого модуля:

- Модуль LoadWnd – загрузочный экран;
- Модуль LevelSelection – главное меню приложения;
- Модуль menu – меню настроек;
- Модуль lvl1 – первый уровень;
- Модуль lvl2 – второй уровень;
- Модуль lvl3 – третий уровень;
- Модуль CustEdit – содержит вспомогательные функции;

Описание разработанных процедур находятся в таблице 1.

Таблица 1 – Процедуры и функции

Имя процедуры (функции)	В каком модуле находится	За каким компонентом закреплена	Назначение
1	2	3	4
procedure TForm4.FormCreate(Sender: TObject);	loadWnd.pas	Form	Установка цветов и значений переменных
procedure TForm4.Timer1Timer(Sender: TObject);	loadWnd.pas	Timer1	Изменение этапов заставки
Procedure TForm4.Timer2Timer(Sender: TObject);	loadWnd.pas	Timer2	Изменение прозрачности заставки
procedure TForm6.WMHotkey(var msg: TWMHotkey);	levelSelection.pas	Form	Обработка горячей клавиши
TForm6.FormClose(Sender: TObject; var Action: TCloseAction);	levelSelection.pas	Form	Отмена регистрации горячей клавиши
procedure TForm6.FormCreate(Sender: TObject);	levelSelection.pas	Form	Инициализация переменных, загрузка медиафайлов
procedure TForm6.FormShow(Sender: TObject);	levelSelection.pas	Form	Воспроизведение музыки при показе формы
procedure TForm6.Timer4Timer(Sender: TObject);	levelSelection.pas	Timer4	Воспроизведение анимации

Продолжение таблицы 1

1	2	3	4
procedure TForm6.Image6Click(Sender: TObject);	levelSelection.pas	Image6	Переключение видимости изображений
procedure TForm6.Image6MouseEnter(Sender: TObject);	levelSelection.pas	Image6	Установка курсора и запуск таймера с анимацией
procedure TForm6.Image6MouseLeave(Sender: TObject);	levelSelection.pas	Image6	Установка курсора и запуск таймера с анимацией
procedure TForm6.Image8Click(Sender: TObject);	levelSelection.pas	Image8	Закрытие приложения
procedure TForm6.Image8MouseEnter(Sender: TObject);	levelSelection.pas	Image8	Установка курсора и запуск таймера с анимацией
procedure TForm6.Image8MouseLeave(Sender: TObject);	levelSelection.pas	Image8	Установка курсора и запуск таймера с анимацией
procedure TForm6.MediaPlayer1Notify(Sender: TObject);	levelSelection.pas	MediaPlayer1	Воспроизведение музыки
procedure TForm6.Image12Click(Sender: TObject);	levelSelection.pas	Image12	Переключение видимости изображений
procedure TForm6.Image13Click(Sender: TObject);	levelSelection.pas	Image13	Показ меню настроек
procedure TForm6.Image14Click(Sender: TObject);	levelSelection.pas	Image14	Открытие справки
procedure TForm6.Image1Click(Sender: TObject);	levelSelection.pas	Image1	Переход на первый уровень
procedure TForm6.Image1MouseEnter(Sender: TObject);	levelSelection.pas	Image1	Смена курсора при наведении
procedure TForm6.Image1MouseLeave(Sender: TObject);	levelSelection.pas	Image1	Смена курсора при потере фокуса
procedure TForm6.Image2Click(Sender: TObject);	levelSelection.pas	Image2	Переход на второй уровень
procedure TForm6.Image3Click(Sender: TObject);	levelSelection.pas	Image3	Переход на третий уровень
procedure TForm6.Timer1Timer(Sender: TObject);	levelSelection.pas	Timer1	Воспроизведение анимации
procedure TForm6.Timer2Timer(Sender: TObject);	levelSelection.pas	Timer2	Воспроизведение анимации
procedure TForm6.Timer3Timer(Sender: TObject);	levelSelection.pas	Timer3	Воспроизведение анимации

Продолжение таблицы 1

1	2	3	4
procedure TForm8.FormCreate(Sender: TObject);	menu.pas	Form	Загрузка изображений
procedure TForm8.Image2Click(Sender: TObject);	menu.pas	Image2	Закрытие меню настроек
procedure TForm8.Image2MouseEnter(Sender: TObject);	menu.pas	Image2	Установка курсора при фокусе
procedure TForm8.Image2MouseLeave(Sender: TObject);	menu.pas	Image2	Установка курсора при потере фокуса
procedure TForm8.Image3Click(Sender: TObject);	menu.pas	Image3	Остановка и включение звуков
procedure TForm8.Image4Click(Sender: TObject);	menu.pas	Image4	Остановка и включения музыки
procedure TForm8.playThisSound(soundname:string);	menu.pas	-	Воспроизведение звуков
function checkEmpty(arr:arrCustomEdit):boolean;	CustEdit.pas	-	Проверка на заполненность игрового поля
procedure setNumbers(arr:arrCustomEdit; arrNum:mas; num:integer);	CustEdit.pas	-	Установка чисел на случайные свободные позиции игрового поля
function GetRandomInRange(const MinValue, MaxValue: Integer): Integer;	CustEdit.pas	-	Получение случайного значения из диапазона
procedure setNum(arr:arrCustomEdit; num:integer);	CustEdit.pas	-	Установка числа на сфокусированный объект игрового поля
procedure FillEdit(board:mas; crd:coordinates; form:Tform; var arr:arrCustomEdit; func:TNotifyEvent; br1, br2:TColor);	CustEdit.pas	-	Установка полей ввода значений в игровое поле
procedure drawTriangles(Canvas: TCanvas; x,y,SideLength:integer; Form:TForm; mode:boolean; Brush:TColor); var x1,y1:integer;	CustEdit.pas	-	Отрисовка игрового поля
procedure TForm5.Validate(Sender: TObject);	Lvl1.pas	TMyCustomEdit	Проверка на корректность ввода значения
procedure TForm5.SpeedButtonClick(Sender: TObject);	Lvl1.pas	SpeedButton	Установка значения в сфокусированное поле ввода
procedure TForm5.FormCreate(Sender: TObject);	Lvl1.pas	Form	Инициализация переменных и изображений
procedure TForm5.FormPaint(Sender: TObject);	Lvl1.pas	Form	Отрисовка игрового поля на форме

Продолжение таблицы 1

1	2	3	4
procedure TForm5.FormCloseQuery(Sender: TObject; var CanClose: Boolean);	Lvl1.pas	Form	Заккрытие игрового окна
procedure TForm1.FormCreate(Sender: TObject);	Lvl2.pas	form	Инициализация переменных и изображений
procedure TForm1.FormCloseQuery(Sender: TObject; var CanClose: Boolean);	Lvl2.pas	form	Заккрытие игрового окна
procedure TForm1.Validate(Sender: TObject);	Lvl2.pas	TMyCustomEdit	Проверка на корректность ввода значения
procedure TForm1.FormPaint(Sender: TObject);	Lvl2.pas	Form	Отрисовка игрового поля
procedure TForm1.SpeedButtonClick(Sender: TObject);	Lvl2.pas	SpeedButton	Установка значения в сфокусированное поле ввода
procedure TForm7.FormCreate(Sender: TObject);	Lvl3.pas	form	Инициализация переменных и изображений
procedure TForm7.FormCloseQuery(Sender: TObject; var CanClose: Boolean);	Lvl3.pas	form	Заккрытие игрового окна
procedure TForm7.Validate(Sender: TObject);	Lvl3.pas	TMyCustomEdit	Проверка на корректность ввода значения
procedure TForm7.FormPaint(Sender: TObject);	Lvl3.pas	Form	Отрисовка игрового поля
procedure TForm7.SpeedButtonClick(Sender: TObject);	Lvl3.pas	SpeedButton	Установка значения в сфокусированное поле ввода

Описание использованных компонентов приводится в таблице 2.

Таблица 2 – Использованные компоненты

Компонент	На какой форме расположен	Назначение
1	2	3
TImage	LevelSelection, lvl1, lvl2, lvl3, LoadWnd, menu	Используется как фон, фотографии, кнопки
TLabel	LoadWnd	Отображение надписей на форме
TMediaPlayer	LevelSelection	Используется для воспроизведения звуков и музыки
TSpeedButton	Lvl1, Lvl2, Lvl3	Используется для взаимодействия с игровым полем
TTimer	LevelSelection, LoadWnd	Используется для создания анимации
TMyCustomEdit	Lvl1, Lvl2, Lvl3	Используется для ввода значений пользователем

Точное название проекта и его состав приводится в таблице 3.

Таблица 3 – Спецификация программы

Имя файла	Назначение
1	2
Hoshi.exe	Исполняемый файл проекта
Hoshi.dproj	Файл проекта, связывает все файлы, из которых состоит приложение
loadWnd.pas	Файл программного модуля для формы TForm4
LevelSelection.pas	Файл программного модуля для формы TForm6
CustEdit.pas	Файл программного модуля
Lvl1.pas	Файл программного модуля для формы TForm5
Lvl2.pas	Файл программного модуля TForm1
Lvl3.pas	Файл программного модуля TForm7
Menu.pas	Файл программного модуля для формы TForm8
HoshiSetup.exe	Установочный файл
manualOfP.chm	Файл справки, содержит tutorial по работе с программой
loadWnd.dfm	Форма загрузочного экрана
LevelSelection.dfm	Главная форма
Lvl1.dfm	Форма первого уровня
Lvl2.dfm	Форма второго уровня
Lvl3.dfm	Форма третьего уровня
Menu.dfm	Форма меню настроек
Lvl1sel.png	Изображение главного меню для формы TForm6
Loadwnd2.png	Изображение заставочного экрана для фона формы TForm4
Settings2.png	Изображение для фона формы меню настроек TForm8
loadWnd.DCU	DCU (Компилированный файл)
LevelSelection. DCU	DCU (Компилированный файл)
Lvl1.DCU	DCU (Компилированный файл)
Lvl2.DCU	DCU (Компилированный файл)
Lvl3. DCU	DCU (Компилированный файл)
Menu. DCU	DCU (Компилированный файл)

4 Тестирование

При разработке данной программы многие возникающие ошибки и недоработки были исправлены на этапе реализации проекта. После завершения испытания реализации программы было проведено тщательное функциональное тестирование. Функциональное тестирование должно гарантировать работу всех элементов программы в автономном режиме.

Отчёт о результатах тестирования предоставлен в таблице 4.

Таблица 4 – Отчёт результатах тестирования

№ теста	Название теста	Ожидаемый результат	Результат тестирования
1	2	3	4
T1	Проверка кнопки «Настройки»	Переход к форме настроек	Переход к форме настроек
T2	Проверка кнопки «PLAY»	Переход к выбору уровня	Переход к выбору уровня
T3	Проверка кнопки «Вопросик»(справка)	Открытие справки	Открытие справки
T4	Проверка кнопки «Выход»	Выход из приложения	Выход из приложения
T5	Проверка кнопки «Звук»	Включение\выключение звуков	Включение\выключение звуков
T6	Проверка кнопки «Начать заново»	Создание игрового поля с башнями и дисками	Создание игрового поля с башнями и дисками
T7	Проверка кнопки «Выход из настроек»	Выход из формы настроек	Выход из формы настроек
T8	Проверка кнопка «1 уровень»	Переход на первый уровень	Переход на первый уровень
T9	Проверка кнопка «2 уровень»	Переход на второй уровень	Переход на второй уровень
T10	Проверка кнопка «3 уровень»	Переход на третий уровень	Переход на третий уровень

При разработке программного продукта было решено множество проблем, например, неверно отображались и перемещались изображения, звуки проигрывались неправильно, ошибки проверки ввода пользователем значений в игровом режиме, поэтому в будущем пользователь не столкнется с данными проблемами.

Элементы программы были проверены, и было установлено, что все они работают правильно и выполняют задачи, указанные в процедурах.

5 Применение

5.1 Общие сведения о программном продукте

Цель данного проекта данного проекта является создание игрового приложения «Головоломка Хоши», знакомство с ее правилами и стратегиями прохождения уровней, а также с интересными аспектами этой головоломки. Игра представляет собой одну из многих вариаций всеми известного «Судоку».

Данный программный продукт предназначен для любого рода пользователей. Применить данную программу смогут как любители головоломок, так и те, кто интересуется логическими играми.

Игра совершенно не требовательная, запускается и функционирует на различных компьютерах (даже на «Древних» ПК)

Тестирование проводилось на разных классах ПК, и работа с игрой была комфортной. Программа разработана для ПК со следующими характеристиками:

- процессор Intel(R) Core(TM) i3-7100U CPU @ 2.40GHz;
- графический адаптер INTEL UHD Graphics 620
- ОС – Windows 11.

5.2 Инсталляция

Для того чтобы установить программу необходимо запустить файл Hoshi-Setup.exe. Появится окно установки игры «Hoshi», следуем инструкции и устанавливаем приложение.

5.3 Выполнение программы

Игру "Hoshi" можно запустить несколькими способами. Первый способ - это запуск через ярлык на рабочем столе. Для этого дважды щелкните левой кнопкой мыши на ярлыке с названием "Hoshi".

Данную программу можно запустить из каталога, в который устанавливалось приложение (по умолчанию C:\Program Files (x86)\Company\Hoshi).

По подготовленным тестам будет осуществляться функциональное и полное тестирование программного продукта. Отчет о результатах тестирования будет представлен в 4 разделе пояснительной записки.

После запуска приложения и прохождения заставки на экране нас встречает главное окно.

					КП 2-40 01 01.35.40.11.24 ПЗ	Лист
Изм.	Лист	№ док.м.	Подпись	Дата		17

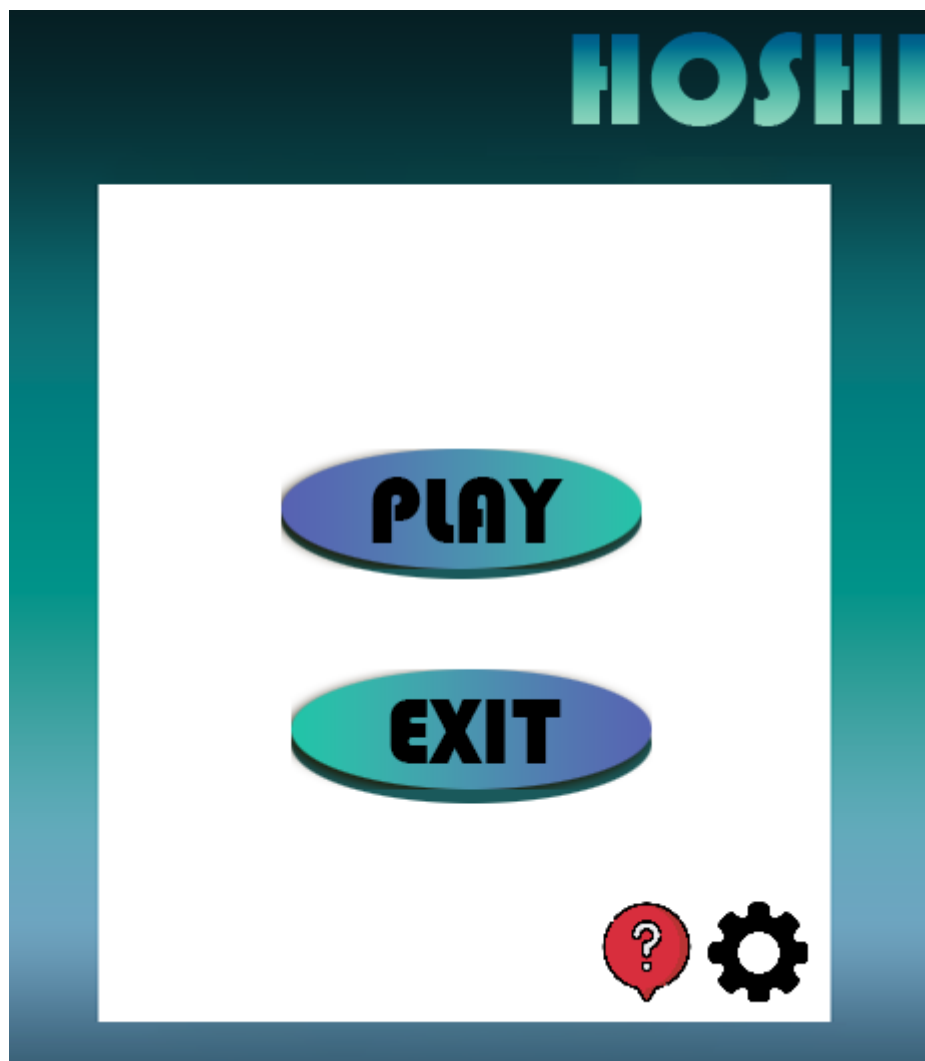


Рисунок 2 – Главное окно

В главном окне находится 4 кнопок: «Настройки», «PLAY», «EXIT», «Справка». При нажатии кнопки «EXIT» воспроизводится выход из приложения.

При нажатии кнопки «Настройки», появляется форма настройки приложения



Рисунок 3 – Меню настроек

При нажатии кнопки «PLAY» на главной форме нам предоставиться выбор уровней, после щелчка на который мы перейдем на форму выбранного уровня.



Рисунок 4 – Выбор уровней

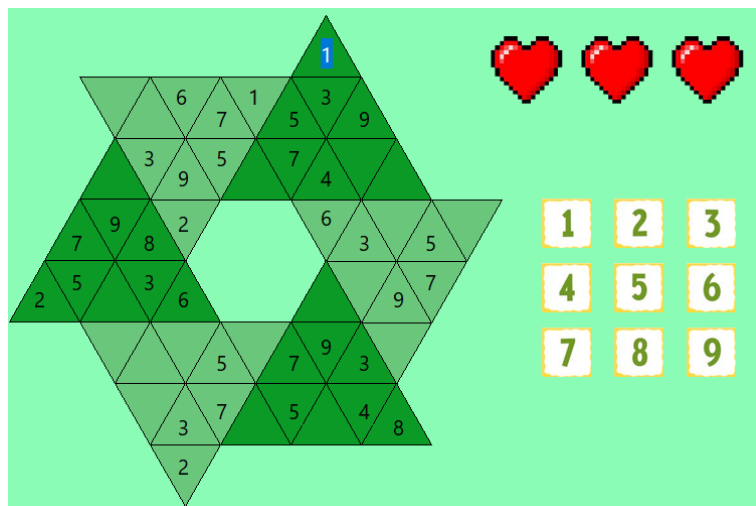


Рисунок 5 - геймплей

Цель игры – решить головоломку, а именно заполнить все пустые поля значениями не нарушая правила игры. Игровой процесс построен таким образом, чтобы постепенно усложнять задачи, развивая и тренируя интеллектуальные способности игроков. По мере продвижения по игре, головоломки становятся все более сложными, требуя от игроков все большей концентрации и логического анализа.

Справочная система запускается с помощью кнопки «Справка» на главном меню.

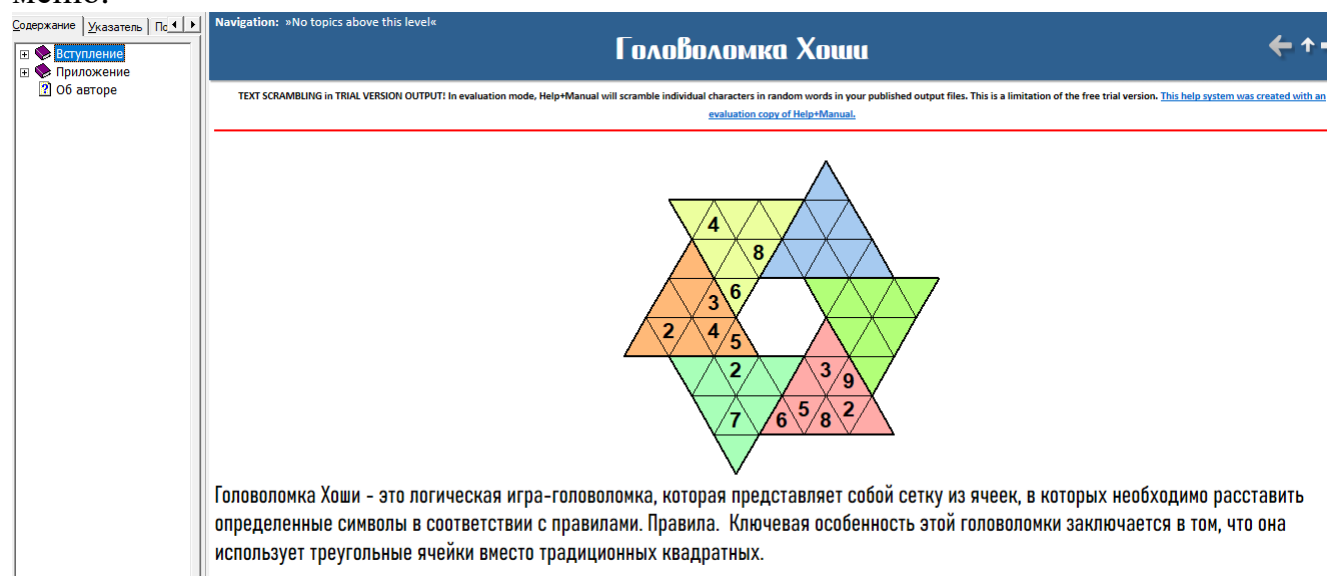


Рисунок 6 - Справка

Заключение

В ходе выполнения данного курсового проекта была разработана интерактивная логическая игра "Головоломка Хоши". Проект включал в себя все этапы создания программного продукта: от концептуализации и разработки дизайна до программной реализации и тестирования.

Разработанное приложение предназначено для широкого круга пользователей. Применить данную программу смогут как любители головоломок, так и те, кто интересуется логическими играми.

Основная сила Delphi - интегрированная среда разработки, которая значительно упростила процесс создания игры. Это подобно конструктору, где можно выбирать и перетаскивать подходящие элементы для создания игрового интерфейса.

Результатом проделанной работы стала игра "Головоломка Хоши" - увлекательный логический проект, сочетающий в себе интригующий геймплей, привлекательный дизайн и высокое качество исполнения. Игра, несомненно, найдет свою аудиторию среди любителей головоломок и логических задач, предоставляя им возможность развивать свои интеллектуальные способности и снимать стресс в увлекательной и творческой форме.

					КП 2-40 01 01.35.40.11.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		21

Список использованных источников

1. OpenGameArt [Электронный ресурс] – Режим доступа:
<https://opengameart.org/> - Дата доступа: 23.06.2024
2. FLATICON [Электронный ресурс] – Режим доступа:
<https://www.flaticon.com/> - Дата доступа: 20.06.2024
3. Pinterest [Электронный ресурс] – Режим доступа:
<https://www.pinterest.com/> - Дата доступа: 20.06.2024

					КП 2-40 01 01.35.40.11.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		22