**Introduction**

The goal of this project is to create a command-line implementation of the classic game Hangman using C. The game will allow users to guess words, track their progress, and display the hangman figure as they guess incorrectly.

**Project Overview**

The Hangman game will be implemented in C and will be played in the terminal. The game will have the following features:

* A random word will be selected from a pre-defined list of words (with themed categories)
* Users will have a set number of guesses to guess the word (until the hangman is complete)
* For each incorrect guess, a part of the hangman figure will be displayed.
* After each guess, the current state of the word will be displayed, with blank spaces for letters that have not been guessed yet.
* Users will have the option to enter a single letter or an entire word to guess.
* The game will end when the user guesses the word correctly or runs out of guesses.
* The game will allow the user to play again or quit.

Those are the root features we want implemented; with more time we will implement:

* A multiplayer mode to play with friends (on the same laptop).
* Power ups and hints.
* Time limit and difficulty levels (different word groupings for different difficulties).

**Contributors**

The following people are contributors to the project with their roles included.

* Cooper Hanson
  + Coding the logic behind the words and categories to be selected by the player.
* George Shea
  + Coding the logic behind the user input and checking if the guess is right or wrong.
* Keegan Gunkel
  + Project manager and documentation. Also coding the logic behind the in-game sounds.
* Anthony Steffl
  + Coding the U.I for the program and just any graphics involved in the program.
* Namduk Tsering
  + Coding the logic behind hints and “power-ups”

There will be potential other enhancements such as a multiplayer mode and different difficulties/time limits that will be contributed to by everyone.

**Technologies Used**

The following C libraries will most likely be used:

* stdio.h - for input/output operations.
* string.h - for string manipulation.
* time.h - for generating random numbers.
* stdlib.h - for memory allocation and random numbers.

**Project Plan**

Basic overview of how our process will go during creating the Hangman game. Other and more audacious features are not included in this plan.

* Stage 1
  + Create a basic structure for the game, including displaying the blank word and hangman figure (AKA the U.I of the game).
* Stage 2
  + Implement user input for guessing a letter or word.
* Stage 3
  + Add logic for checking if the user's guess is correct or incorrect, and updating the game state accordingly.
  + Add a mechanism for tracking the number of incorrect guesses and displaying the hangman figure.
* Stage 5
  + Implement end-game logic for when the user wins or loses.
  + Add a mechanism for playing again or quitting the game.

**Expected Outcomes**

The expected outcomes of this project are:

* A working implementation of the Hangman game in C.
* A better understanding of the C programming language and its libraries.
* Improved problem-solving skills and ability to translate requirements into working code.
* A deeper understanding of software design principles.

**Conclusion**

In conclusion, this project aims to implement the classic game Hangman using C. By creating this game, we hope to deepen our understanding of C programming, software design principles, and problem-solving skills.