Design Plan - Hangman

James Viner

Project Summary

Hangman is a game to be played interactively in the console with the user. The program reads in a word list, chooses a random one from it, and then allows the user to play the traditional word-guessing game of hangman, tracking various player stats across all games played with a save file.

Features Targeted

Allow Punctuation

Allow the chosen word to contain punctuation such as apostrophes and hyphens and print any punctuation in a given word for free without needing to be guessed for. This can be accomplished with an additional check against any given round's selected word, looking for special symbols, and marking them to be printed before the user's first guess.

Track Time

Track the total time spent playing the game alongside the other elements in the game's save file in the format of HH:MM:SS. This will require the starting and stoppage of the round's timer as well as the conversion from total seconds spent playing into minutes and hours. The timer persisting between rounds can be implemented with the save file.

Architecture

Data

The spec defines that the program must support up to at least 34 characters in a single word. It will be possible to 'turn on' which characters are to be displayed for a given game state by storing the chosen word along with a list true and false values corresponding to each character in the sequence and whether it should be displayed.

Significant Functions

char* load_save(void)

Loads the textual save information from file in user's home directory or creates one if it does not exist. Returns a pointer to a string (that will need to be freed after use) containing the loaded information.

void save_game(char *save_string, int game_score, int time_spent)

Saves the current game's statistics, pulling existing information from the save_string created by load_save.

int select_word(void)

Selects a word at the start of the game from either the specified file or the default file. Returns 0 if an error was encountered with reading the file or selecting a valid word. Returns 1 if successful.

User Interface

The game will be played using the CLI. The user will enter a guess of one letter at a time, being informed of if they have entered an invalid guess (more than one character or non-alphabetic character). Upon receipt of a valid guess, the program will inform the user of whether or not it was in the hangman word by either incrementing the incorrect guesses counter or by showing all instances of the character in the word. If the input stream is closed at any time, the program will exit.