2023年5月24日　提出

20-413

北野正樹

プログラミング演習IIIA

Hangman制作についての報告書

1. プログラムの概要
   1. Hangmanプログラムの説明

今回作成したhangmanプログラムはその名の通り規定数で未知の単語を当てるというクイズ形式のゲームとなっている。規定回数で単語を当てるというとてもシンプルな内容であるが、Wordleなどと名前を変えて世界で広く遊ばれているミニゲームである。今回の仕様では七回の間に単語を当てる仕様となっている。現在のプログラムでは7回で当てるということになっており、これはゲーム内では変えることができない。

英単語はTOEICの頻出単語1500からランダムに出題される仕組みになっており、ファイルは実行プログラム直下においてある。ここのファイルパスを変えることで出題される単語を変えることができる。

ゲームルールとしてはとても単純で一回のターンで一文字英小文字を入力することができ、それを七回以内にすべてのわからない文字を当てるというものである。

* 1. 本プログラムの開発環境

本プログラムの開発環境を下記に示す

・マシン：MacBook Pro2020M1

・OS：MacOS Ventura 13.4

・エディタ：Emacs

・言語：Clang

・コンパイラ：Apple clang version 14.0.3 (clang-1403.0.22.14.1)

* 1. 本プログラムの動作環境

MacOS Ventura 13.4以降　検証済

Debian GNU Linux（Macでの仮想環境） 検証済

Windows11（Macでの仮想環境）検証済

OSに依存しないヘッダーを使用して作成。

* 1. 本プログラムの使用方法

本ソースコードを上記のコンパイラのバージョンと同じ、またはそれ以上のバージョンでコンパイルし、実行ファイルを生成する。その実行ファイルと同じ階層に単語のファイルを置き実行すると、ゲームがスタートする。設定メニューなどはなく実行した瞬間からゲームが開始する。ゲームに表示されているメニューについて説明する。

・単語：------

ここには一致する文字を入力すると文字が現れていく。文字数と穴食いの場所から単語を予想することができる。

・使われた文字：

ここには今まで使われた文字が表示される。今までというのはそのゲームの文字であり、累計ではない。ゲーム中は同じ文字を入力することはできないようになっている。

・残り回数：

ここにはその名の通りあと何回回答できるかがカウントされる。0になるとゲームオーバーとなる。

・文字を入力してください：

ここのラインに予想した文字を入れる。予想した文字をタイプして、Enterキーを押すことで次のターンへ移ることができる。

ゲームオーバー、またはゲームクリアした際には正誤と正解単語が表示される。また、引き続きプレイするかをYまたはNで選択する。Yを押すとまた新しい単語がロードされ新しくゲームを始めることができる。Nを押すとゲームが終了する。

1.3 本プログラムの必要な変数

本プログラムの変数の管理は構造体などは用いずにグローバル変数として作成した。新規インスタンスを生成する必要がなく、変数を包括する必要がないと判断したためである。

char \*correctWord: 正解の単語を格納するポインタ配列。ゲーム開始時に毎回ランダムな文字列が代入される。サイズは動的確保。

int wordLength: 正解単語の文字列長を格納しておく変数。

int remainCount: 残りの回答数を格納しておく変数。

char usedChar[LIMIT\_TIMES+1]: 使用した単語を格納しておく配列。サイズは回答数を用いて静的確保。

char \*correctFlag: ゲーム内で出力するためのポインタ配列。ここの配列をその都度変更してゲームに出力している。サイズは動的確保

Boolean isCorrect: ゲームクリアかゲームオーバーかを判定する変数。

Boolean isRestart: ゲームを再スタートするかどうかを判定する変数。

* 1. 各関数についての説明
     1. main関数

本関数では、ゲーム全体の流れを司っている。入力に対して出力、判定などほとんどの処理をここで処理している。

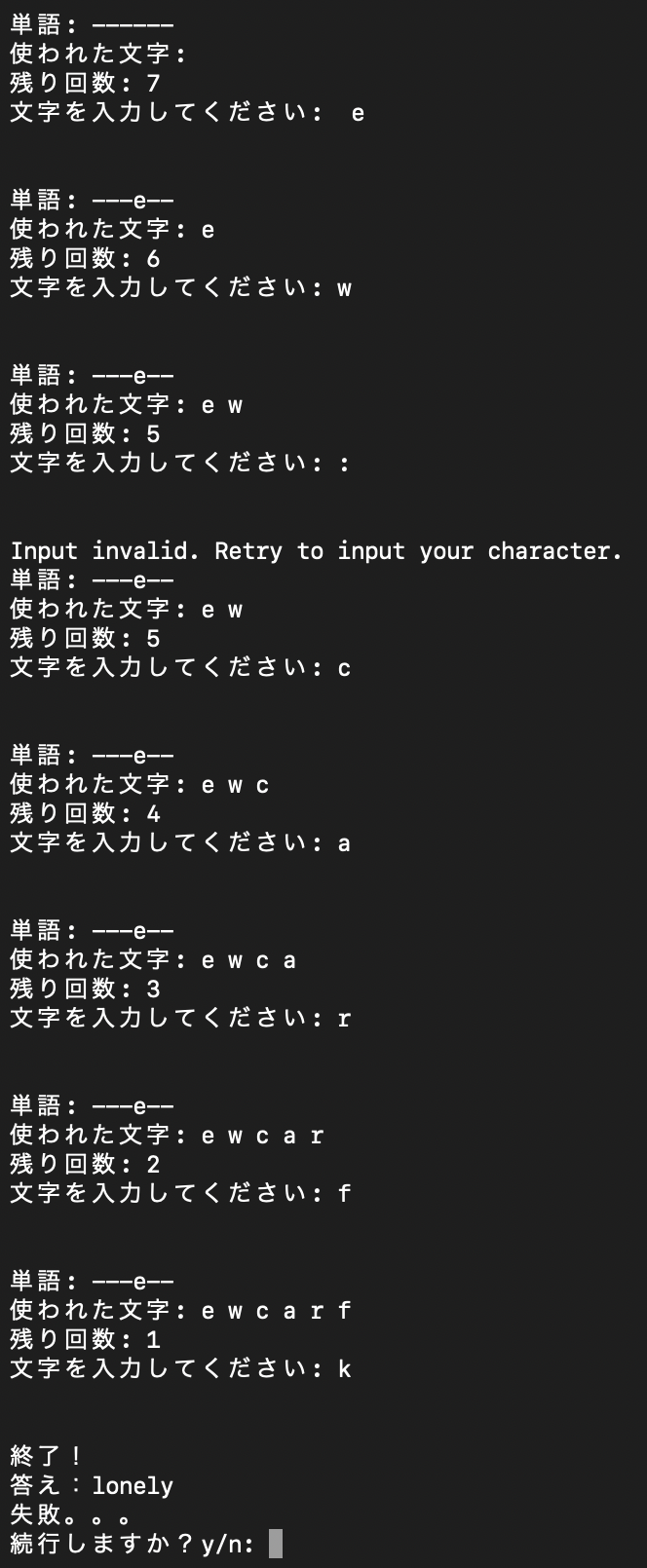
* + 1. input関数

本関数では、コマンドラインから一文字受け取る。エスケープシーケンスを用いて無駄な文字が入らないようになっている。

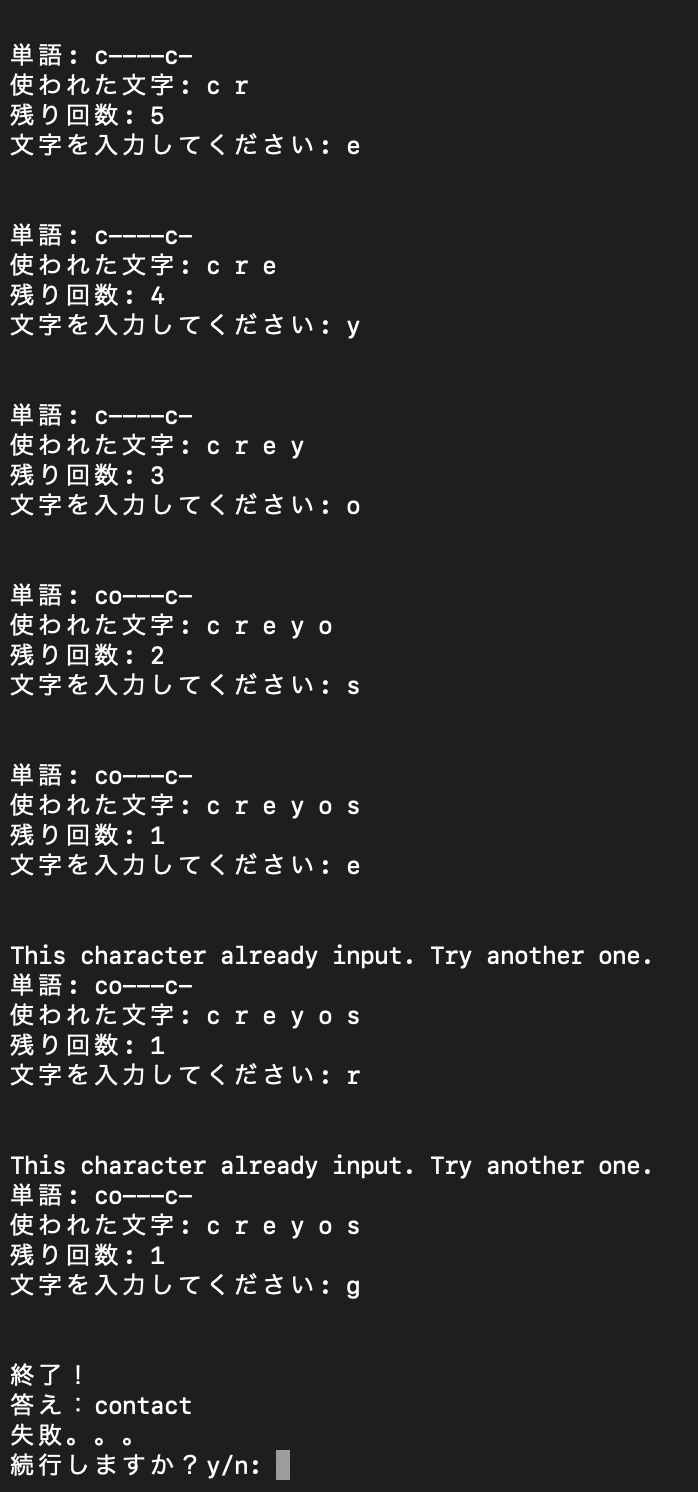
* + 1. initialize 関数

本関数では、ゲーム開始時のランダムシードの初期化や各変数の処理か処理を行っている。

* 1. 本プログラムの検証

本プログラムの検証を行ったため、下に示す。

このように失敗する旨の表示がある。また、英小文字ではない場合はもう一度入力させるようになっている。



このように同じ文字を入力するともう一度入力するように言われる。

* 1. 本プログラムの課題

本プログラムでは変数の動的割り当てなどにより少し処理が複雑化してしまった。また、ほとんどの処理をメイン関数内に収めたことで、少し可読性が悪くなってしまった。もう少し関数にし可読性をあげ、今後も拡張しやすいようなプログラムを書けるようにするべきだと感じた。

* 1. ソースコードの全容

今回作成したソースコードを以下に示す。

