

## Assignment 4

Due 6pm Sunday November 15

Recall that Hangman entails guessing letters to determine a secret word. Each time you guess a letter in the word, where the letter appears is revealed. Each time you guess a letter not in the word, you lose one life. You win if you guess the whole word before running out of lives. Your program should be called `hangman.c`

### Header file

You will be supplied with a header file called `hangman.h`. This should be saved to the directory, it **must** not be altered, and your program should include this file by

```
#include "hangman.h"
```

The header file creates some constants, some global variables and gives prototypes for the required functions, as well as some code for loading the dictionary.

### Recommended Functions

The functions you should use are:

- **getGuess**: this `char` function outputs the current progress, then reads and returns a lowercase letter from the user
- **isInGuess**: this boolean function tests whether the supplied letter is in the secret string
- **updateProgress**: this void function updates the information needed to display the progress
- **haveAllLetters**: this boolean function tests whether all letters in the secret string have been guessed
- **play**: this runs the game for the supplied string. The main part is a loop that might look like:

```
while( missesLeft>0 && !isDone) {
    next = getGuess...
    if( !isInGuess(...) ) {
        tell user;
        missesRemaining--;
    }
    else {
        updateProgress(...);
        if( haveAllLetters(...) )
            isDone=true;
    }
}
```

- **getRandomSecret**: this returns a random entry from the dictionary

The main function should check to see whether the user supplied an optional argument. If so, that should be used for the secret string; otherwise the dictionary should be loaded and then `getSecret` should be used to supply the secret. In either case, `play` is then called.

```
./a.out
Loaded 26 words into the dictionary
Let's go!
- - - - - Enter next guess: t
>>You've got 7 misses left
- - - - - Enter next guess: n
>>You've got 6 misses left
- - - - - Enter next guess: e
- - - - - e _ Enter next guess: r
- - - - - e r Enter next guess: d
>>You've got 5 misses left
- - - - - e r Enter next guess: s
- - - - - s _ e r Enter next guess: l
- - - l _ s _ e r Enter next guess: a
>>You've got 4 misses left
- - - l _ s _ e r Enter next guess: o
>>You've got 3 misses left
- - - l _ s _ e r Enter next guess: i
- - - l i s _ e r Enter next guess: h
- - - l i s h e r Enter next guess: c
>>You've got 2 misses left
- - - l i s h e r Enter next guess: m
>>You've got 1 misses left
- - - l i s h e r Enter next guess: p
p _ _ l i s h e r Enter next guess: u
p u _ l i s h e r Enter next guess: b
Well done! The secret word was publisher
```

Submit your source code using `handin.101.303 4 hangman.c`

You are to work independently, but can ask questions from the lecturer and lab instructors. Late submissions will be significantly penalized.