

# CS150 Assignment 6

## Hangman

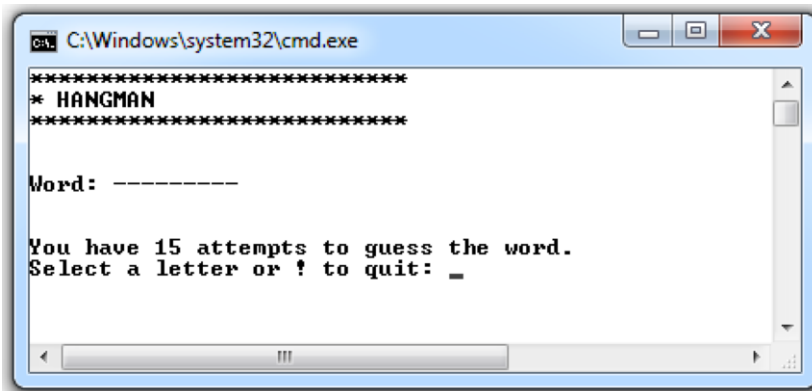
**Date assigned:** Wednesday, November 11, 2015

**Date due:** Monday, November 23, 2015, 9:15am (40 points)

For this assignment, you will write a complete C++ program that simulates the game hangman. Users are given 15 attempts to guess a word. During each turn, the user selects a letter. If that letter exists, then the location of those letters is shown to the user.

### Sample Run:

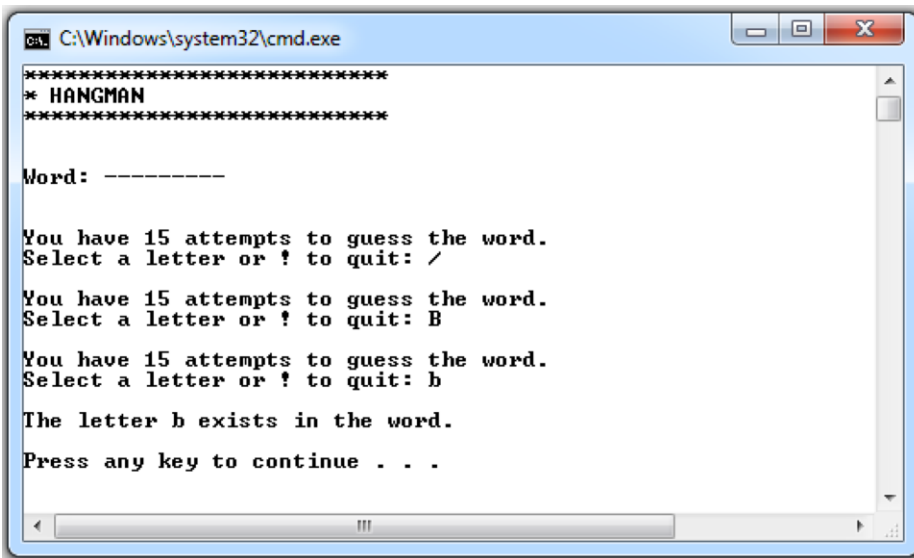
#### Run 1:



```
C:\Windows\system32\cmd.exe
*****
* HANGMAN
*****

Word: -----

You have 15 attempts to guess the word.
Select a letter or ! to quit: _
```



```
C:\Windows\system32\cmd.exe
*****
* HANGMAN
*****

Word: -----

You have 15 attempts to guess the word.
Select a letter or ! to quit: /

You have 15 attempts to guess the word.
Select a letter or ! to quit: B

You have 15 attempts to guess the word.
Select a letter or ! to quit: b

The letter b exists in the word.

Press any key to continue . . .
```

```
C:\Windows\system32\cmd.exe
*****
* HANGMAN
*****

Word: b----b---

You have 14 attempts to guess the word.
Select a letter or ! to quit:
```

```
C:\Windows\system32\cmd.exe
*****
* HANGMAN
*****

Word: b----b---

You have 14 attempts to guess the word.
Select a letter or ! to quit: z

The letter z does not exist in the word.
Press any key to continue . . . _
```

```
C:\Windows\system32\cmd.exe
*****
* HANGMAN
*****

Word: b----b---

You have 13 attempts to guess the word.
Select a letter or ! to quit: l

The letter l exists in the word.
Press any key to continue . . .
```

and keep guessing until ....

```
C:\Windows\system32\cmd.exe
*****
* HANGMAN
*****

Word: blackbir-

You have 7 attempts to guess the word.
Select a letter or ! to quit: d
The letter d exists in the word.

Word: blackbird

Congratulations! You solved the word in 9 guesses.
Press any key to continue . . .
```

**Run 2:**

```
C:\Windows\system32\cmd.exe
*****
* HANGMAN
*****

Word: -----

You have 15 attempts to guess the word.
Select a letter or ! to quit: !
Press any key to continue . . .
```

**Run 3:**

Guess the letters a, b, c, through n, and then:

```
C:\Windows\system32\cmd.exe
*****
* HANGMAN
*****

Word: blackbi-d

You have 1 attempts to guess the word.
Select a letter or ! to quit: o
The letter o does not exist in the word.

Sorry. You have run out of guesses.

Word: blackbird

Press any key to continue . . .
```

**Notes:**

- The words will only contain lower case letters. The word will not contain whitespace.
- A word will never be longer than 100 letters.
- Store the word and the size of the word as constants at the top of your main.
- Pause the game and clear the screen between turns using system (“pause”) and system (“cls”).
- You must use the functions listed on the next page. You may add more functions if you like.

**Remember**

- Code and test one function at a time.
- The debugger is your friend!

## You must use the following functions:

- `void printTitle (string title);`

Prints the `title` passed to the function, including the border of stars, to the screen.

- `void printWord (char theWord[], int wordSize);`

Prints the contents of the array `theWord` to the screen. `wordSize` is the number of characters in the array `theWord`.

- `void initializeUserGuess (char guess[], int wordSize);`

Sets the first `wordSize` elements in the array `guess` to dashes (-).

- `void getSelection (int numGuesses, char &selection);`

Displays a message to the user stating the number of guesses that they have remaining, then ask them to select a letter. Only accept values between a-z or !. If a user enters a different value, then ask them for their selection again.

- `bool checkLetter (char word[], char guess[], int size, char letter);`

Check if the letter (`letter`) exists in the array `word`. If the letter does exist, then change the appropriate dashes in the array `guess` to `letter`. For example, if `word` contains [did], `guess` contains [---], `size` contains 3, and `letter` contains d, then this function will change the contents of `guess` to [d-d] and return true;

- `bool checkWord (char guess[], int wordSize);`

Returns true if the array `guess` does not contain any dashes. Returns false if the array `guess` contains any dashes.

**To complete this assignment you must submit the following:**

**1. An electronic copy of your program on Grace**

- a. Add a new project named **06\_Hangman** to your previously created assignment solution called PUNetIDAssignments. It is *vital* that you name your project correctly!
- b. Type your program (fully documented/commented) into the project. We are now commenting each function in a program. You must follow the coding standards!
- c. Pay attention to the example output! Your program's output must look **exactly** like the example output! The spacing and newlines in your output must match exactly.
- d. Make sure that your program compiles and runs correctly. If you get any errors, double check that you typed everything correctly.
- e. Make sure that your program does not produce any warnings.
- f. Once you are sure that the program works correctly it is time to submit your program. You do this by logging on to Grace and placing your complete solution folder in the **CS150-02 Drop** folder. This solution folder must contain seven projects.
- g. The program must be in the drop folder by 9:15am on the day that it is due. Anything submitted after that will be considered late.

**2. A hard copy of your program**

- a. The hard copy must be placed on the instructor's desk by 9:15am on the day that it is due.
- b. The hard copy must be printed in color, double-sided, and stapled if necessary.
- c. Your tab size must be set to 2 and you must not go past column 80 in your output.

**Good luck! And remember, if you have any problems, come and see me straight away. ☺**