

CS306 – Introduction to Perl, Fall 2005
Extra Credit Homework Assignment #6
Due: December 5th, 2005 (start of class)

NOTE: I am aware that source code for both questions is readily available on the Internet. DO NOT USE IT. It is very obvious when a class member is not writing their own code. In addition, Perl is very good at text operations. This means it is very good at comparing two pieces of text and determining that they are, say, 74% similar. I'll let you draw your own conclusions from that. :-)

Hangman (5 pts). Implement the classic game of hangman. Implement the following features:

- Create a database of at least 20 words, each at least 6 characters long, that is stored on disk.
- Allow the user to play more than one time per run of the program. Ensure that you do not repeat words within the same playing session.
- Randomly choose a word from the list for each game.
- Present the user with an ASCII art representation of the game board (represent missing letters with a _ symbol. I'll let you decide how to make the hangman body.) Update this representation after each guess (so it will have the effect of scrolling in the terminal window).
- The game ends when all the letters of the word are guessed or the user has had six incorrect guesses (head, body, two arms, two legs)

Blackjack (5 pts). Implement a simplified game of blackjack. Create a data structure in memory to represent a single deck of cards (text like 9S is sufficient to indicate 9S). Randomly deal from this deck to two players (the user and the computer). The rules are simple:

- The closest hand to 21 without going over wins
- The Dealer deals two cards to the user (face up) and two to himself (face down)
- The user goes first. The user has the choice to Hit (get another card) or Stay (keep the current hand). If the user Busts (goes over 21) the dealer wins. If the user Stays, it is now the Dealer's turn. If the user gets exactly 21, the game is over and they win that round.
- The Dealer must hit until he has at least 17, and must stop as soon as he has at least 17.
- Numbered cards are worth their face value. Face cards are worth 10. Aces are worth either 1 or 11 (whichever leads to the better score for the player.) Example: A user has a hand of ace,six. This hand is worth 17 points. The user chooses to hit and gets an eight. If the ace were treated as an 11, the hand would bust with a value of 25, so the ace is now worth 1, and the hand is worth 15.
- For this simplified game, don't worry about betting, doubling down or splitting the hands.
- Reshuffle the deck (i.e. make all 52 cards available again) after each round.
- Display how many rounds the user and the dealer have won in the session.

Remember to use strict, comment thoroughly, and use modular programming techniques.