Assignment 1. Submission Deadline Aug 31, 2016. All programming should be done in C.

- Write a program for simulating a Turing machine that has a semi-infinite tape–a tape that extends to "infinity" in one direction. Read "infinity" as "as long as necessary". Use doubly linked lists for the implementation. Your program should take two inputs (i) (Q, δ, q_0, F) where Q is a finite set of states, q_0 is the special start state and $F \subseteq Q$ is a set of final states. The input alphabet is binary and the tape alphabet could include any other special symbol(s) in addition to the input alphabet. The transition function is specified as a series of table rows separated by a semicolon and terminated by a full-stop. For example: $q0$ 0 $q2$ 1 L; $q2$ 1 $q1$ 0 R; $q1$ b $q2$ 1 N. The last row says that when in state q1, if you see a "special" tape symbol b, go to state $q2$, write a 1 in the cell, and keep the tape head there itself. The previous rows are then self explanatory. (ii) a string $x \in \{0, 1\}^*$ that is input to the TM. At the beginning of computation, copy this string onto the tape.

  The instantaneous description of the Turing machine is defined by uqw, where the tape head is on the first symbol of the string w, u is the string to the left of the tape head, and the state of the machine is q. Print the instantaneous description of the tape after each step of simulation. The program should simulate the Turing machine on the input x until the computation halts. (20 marks)

- Construct binary search trees resulting from all possible permutations of N distinct real numbers. Use structures and pointers. Experiment with small N: calculate the average total depth as
  $D(N) = \frac{1}{N!} \sum_{\text{all BSTs}}$ (total depth of BST). Compare this with the expression $2(N+1)\sum_{i=1}^{N}(1/i) - 3N$ that we arrived at in the class. (10 marks)