Instructions

- Do not start this test until you are told to do so! You have 15 minutes for this quiz.
- This is a closed book exam. No notes or other aids are allowed.
- For partial credit, write neatly, show all of your work, and clearly indicate your answers.

1. (6 pts) OCaml Types and Type Inference
   
   a. (2 pts) Give the type of the following OCaml expression
      
      ```
      4::<[
      Type =
      ```
   
   b. (2 pts) Write an OCaml expression with the following type
      
      ```
      int * string
      Code =
      ```
   
   c. (2 pts) Give the value of the following OCaml expression. If an error exists, describe the error.
      
      ```
      let x = 3 in let x = 5 in x * x
      Value =
      ```

2. (4 pts) Regular expressions and NFA

   Given the following NFAs representing the regular expressions R and S, use the algorithm presented in class to add/modify states and transitions to produce a NFA representing the regular expression R*S.
3. (10 pts) NFA and DFA

Consider the following NFA

![NFA Diagram]

a. (2 pts) Does the NFA accept the string “ab”? If it accepts the string, list a sequence of state transitions (e.g., 3, 2, 1) that leads to acceptance of “ab”.

b. (8 pts) Convert the NFA to a DFA using the subset construction algorithm discussed in class. Be sure to label each state in the DFA with the corresponding state(s) in the NFA.