1. (Finite State Machine; 28 points)

(a) (Verification; 11 points) It is claimed that the following machine accepts a binary string in which every 0 is followed by a 1. Annotate the diagram and write the predicates which must be proven to verify this claim.

Let \( p(x) \) denote that in string \( x \) every 0 is followed by a 1. Let \( q(x) \) denote that in string \( x \) the last bit is a 0, and every 0, except the very last one, is followed by a 1.

(b) (Finite State Transducer; 11 points) Draw a finite state transducer whose input is a binary string ending with a special symbol \# . It outputs a binary string ending with \# according to the following rule: a contiguous sequence of 0s is replaced by a single 0 and a contiguous sequence of \( n \) 1s is replaced by a sequence of \( n + 1 \) 1s. Thus, 0010011# becomes 0110111#.

(c) (Regular Expression; 6 points) Write 6 of the shortest strings of \((000 \mid 1)^* (0 \mid 11)^*\).

2. (Recursion and Induction; 22 points)

(a) (4 points) Define a function, \( \text{between} \), of three arguments, \( x \), \( y \) and \( z \), each a number. The output is \text{True} iff \( x \) falls strictly between \( y \) and \( z \). \( y \) may be less than, equal, or greater than \( z \). Thus \( \text{between} \ 3 \ 5 \ 7 \) is \text{False}, \( \text{between} \ 3 \ 5 \ 2 \) is \text{True}, and \( \text{between} \ 3 \ 2 \ 3 \) is \text{False}.

(b) (6 points) Define a function which has a non-empty list of numbers as argument. Its output is \text{False} iff all numbers are identical.

(c) (6 points) Define function \( \text{zip} \) which takes a pair of lists of equal lengths as argument and returns a list of pairs of corresponding elements. So, 
\[
\text{zip} ([1,2,3], ['a','b','c']) = [(1,'a'), (2,'b'), (3,'c')]
\]

(d) (6 points) Define function \( \text{unzip} \) which is the inverse of \( \text{zip} \):

\[
\text{unzip} (\text{zip} (xs,ys)) = (xs,ys)
\]