1. (Recursive Programming; 28 points)

(a) 
\[
\text{tie}([], []) = []
\]
\[
\text{tie}(x:xs)(y:ys) = (x,y):(\text{tie}(xs,ys))
\]
\[
\text{untie}[] = ([], [])
\]
\[
\text{untie}(x,y):zs = ((x:xs),(y:ys))
\]
where \((xs,ys) = \text{untie}zs\)

(b) Define function \(gpos\) of two arguments \(k\) and \(n\), both integers and \(n \geq 0\), where \((gpos\ k\ n)\) returns a list of \(n\) consecutive integers starting at \(k\). Then,

\[
\text{pos } n = gpos\ 1\ n
\]

The definition of \(gpos\) is given by

\[
gpos\ k\ 0 = []
\]
\[
gpos\ k\ (n+1) = k: (gpos\ (k+1)\ n)
\]

(c) 
\[
\text{check } f\ g\ zs = \text{glt} (f\ zs)\ (g\ zs)
\]
\[
\text{glt}\ xs\ [] = \text{True}
\]
\[
\text{glt}\ []\ ys = \text{False}
\]
\[
\text{glt}\ (x:xs)\ (y:ys) = (x \geq y)\ \&\&\ \text{glt}\ xs\ ys
\]

2. (Finite State Machines)

(a) The following machine accepts strings with even number of zeroes.

![Machine diagram](image)

Figure 1: Machine that accepts strings with even number of zeroes
The following machine accepts strings ending in “11”.

![Figure 2: Machine that accepts strings ending in “11”](image)

Combine the two machines.

![Figure 3: Combination of the two machines given above](image)

(b) Define the following predicates.

- $emp(x)$: string $x$ is empty
- $p0(x)$: last symbol of string $x$ is 0
- $p1(x)$: last symbol of string $x$ is 1
- $eq(x)$: number of zero blocks in $x$ equals the number of one blocks
- $lt(x)$: number of zero blocks in $x$ is 1 less than the number of one blocks
- $gt(x)$: number of zero blocks in $x$ is 1 more than the number of one blocks

Annotate as follows:

- **A**: $emp \land eq$
- **B**: $p0 \land gt$
- **C**: $p1 \land lt$
- **D**: $p1 \land eq$
- **E**: $p0 \land eq$

Proof for transition:

- **C** to **E**: $p1(x) \land lt(x) \Rightarrow p0(x0) \land eq(x0)$
- **E** to **E**: $p0(x) \land eq(x) \Rightarrow p0(x0) \land eq(x0)$
- **E** to **C**: $p0(x) \land eq(x) \Rightarrow p1(x1) \land lt(x1)$

(c) $0(0|1)^*1$

2