Due: By 2pm on Friday 19 October. **Worth**: 15%

This assignment is to be completed in groups of no more than three students. Hand in a single paper for your group, with the information about each student filled in on the cover page.

Keep in mind that your proofs will be graded on their structure at least as much as on their content. In other words, when we ask you to prove a statement, it is to find out how well you can write proofs, not because we are interested in knowing whether or not that statement is true. So pay particular attention to writing your proofs properly.

1. [10 marks]

   A computer network consists of a number of "hubs", together with links between certain pairs of hubs. In this question, we assume that links are bidirectional (information can flow from either end to the other). Suppose that in a network, each hub knows how many other hubs it is linked with (this is called the hub's "connectivity").

   Make a conjecture about the relationship between the total number of links in a network, and the sum of the connectivities of all hub. Prove your conjecture using induction.

2. [10 marks]

   For all natural numbers n, let $S_0(n)$ be the sum of the even-numbered digits of n and $S_1(n)$ be the sum of the odd-numbered digits of n. In other words, if $n = d_k d_{k-1} \dots d_1 d_0$, then $S_0(n) = d_0 + d_2 + d_4 + \cdots$ and $S_1(n) = d_1 + d_3 + d_5 + \cdots$ (in decimal notation, of course). For example, $S_0(41232) = 2 + 2 + 4 = 8$ and $S_1(41232) = 3 + 1 = 4$.

   Use induction to prove that for all natural numbers n, n is divisible by 11 iff $S_1(n) - S_0(n)$ is divisible by 11. (HINT: Prove a stronger (i.e., more precise) statement.)

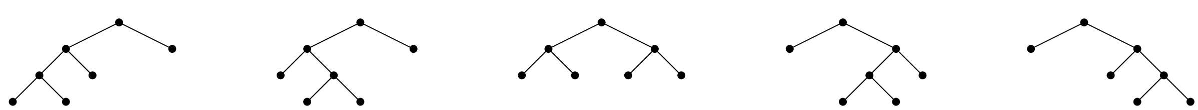
3. [5 marks]

   Give a recurrence relation for the number of sequences of n bits in which there are no consecutive 1s, and justify that it is correct.

4. [10 marks]

   Recall from the textbook the definition of "full binary tree" (a binary tree in which no node has exactly one child, *i.e.*, every node is either a leaf or has two children).

   Let $T(n)$ represent the number of different full binary trees on n nodes (assuming no distinguishing information is stored at the nodes). For example, $T(7) = 5$, as illustrated below.

   Give a recurrence for $T(n)$, including appropriate boundary conditions, and justify briefly that your recurrence is correct.

   Then, prove that $T(n) \geq \frac{1}{n} 2^{(n-1)/2}$ for all odd $n \geq 1$.