Discrete Mathematics MT16: Problem Sheet 3

Chapters 4 (Relations) and 5 (Sequences)

3.1

(i) Find a relation on $\mathbb{N}$ which is antisymmetric, irreflexive, transitive, serial, and not symmetric.

(ii) Find and draw a relation on $\{a, b, c\}$ which is neither symmetric nor antisymmetric, and neither reflexive nor irreflexive.

(iii) Find and draw a relation on $\{a, b, c\}$ which is *not* symmetric, but of which the transitive closure *is* symmetric.

3.2 How many different relations are there on the set $\{0, 1\}$? How many are reflexive? How many are symmetric? How many are antisymmetric? How many are transitive?

If $|A| = n$, how many different antisymmetric relations are there on A?

3.3 Define the relation $m \sim n$ on $\{1, 2, 3, 4, \ldots, 15, 16\}$ by $m \sim n$ if $n = 2^k m$ for some $k \in \mathbb{Z}$.

(i) Prove that $\sim$ is an equivalence relation,

(ii) Find all the equivalence classes.

Let N be a fixed positive integer and define the relation $m \sim n$ on $\{1, 2, 3, \ldots, 2N\}$ by $m \sim n$ if $n = 2^k m$ for some $k \in \mathbb{Z}$. How many different equivalence classes are there? (There is no need to give a proof that your answer is correct.)

3.4 Consider the Fibonacci sequence

$$F_0 = 0, \quad F_1 = 1, \quad F_{n+2} = F_{n+1} + F_n.$$

Prove, by induction on n, that

(i) $F_0 + F_1 + \cdots + F_n = F_{n+2} - 1$ for $n \in \mathbb{N}$;

(ii) $0 \cdot F_0 + 1 \cdot F_1 + \cdots + n \cdot F_n = n \cdot F_{n+2} - F_{n+3} + 2$ for $n \in \mathbb{N}$.

3.5

(i) Prove, by induction on n, that $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$ for $n \in \mathbb{N}_+$.

(ii) Prove, by induction on n, that $\sum_{i=1}^{n} i(i+1) = \frac{n(n+1)(n+2)}{3}$ for $n \in \mathbb{N}_+$.

(iii) Can you generalise these results?

3.6 Consider the recurrence $a_1 = 1$, $a_n = n a_{\lfloor n/2 \rfloor}$ for $n \geq 2$. Compute the first 6 terms of this sequence.

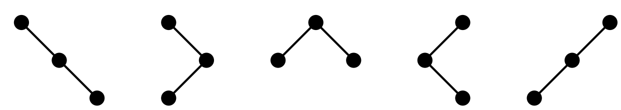
Prove, using strong induction, that $a_n \leq n^{\log_2 n}$.

3.7 This question is about counting the number of *binary trees* with n nodes. A binary tree consists of a set of nodes, with one node distinguished as the *root*. Starting with the root, every node leads to up to two futher nodes: a *left child*, a *right child*, or both, or neither. All nodes are eventual descendants of the root. Binary trees are considered different even if they only differ in whether a node is the left or right child of its parent.

We write b_n for the number of binary trees with n nodes, and conventionally set $b_0 = 1$. If we draw binary trees with the root at the top and children beneath, we can see that there is one tree with one node:

two with two nodes:

and five with three nodes:



(i) Explain carefully why b_n satisfies the recurrence

$$b_n = b_0 b_{n-1} + b_1 b_{n-2} + \cdots + b_{n-2} b_1 + b_{n-1} b_0 \text{ for } n > 1.$$

(ii) Use the recurrence to find b_5.

(There *is* a concise solution to this recurrence, but we have not covered the techniques needed either to find it or to prove its correctness.)