COL202: Discrete Mathematical Structures. I semester, 2020-21.
Amitabha Bagchi
Tutorial Sheet 2: Proofs.
8 October 2020

Important: The question marked with a ♠ is to be submitted via gradescope by 11:59PM on the day that you have your tutorial.

Note: Wherever the question asks you to "write a proof" or to "prove" you must follow the "Guidelines for writing a proof" linked on the course webpage. The emphasis of this tutorial sheet is on *writing proper proofs* so keep rough work and proofs separate.

Problem 1
Given two predicates $P, Q : \mathbb{N} \to \{T, F\}$, suppose we know that $\forall n \in \mathbb{N} : (n \geq 5) \Rightarrow P(n)$ and $\forall n \in \mathbb{N} : (n \geq 6) \Rightarrow Q(n)$ are true. Prove that $\exists n \in \mathbb{N} : P(n) \wedge Q(n)$. Can we prove or disprove that $\exists n \in \mathbb{N} : P(n) \wedge \neg Q(n)$?

Problem 2 ([1], Prob 7, pp 115)
Prove using the contrapositive method that for all real numbers x, if $x^2 - 2x \neq -1$ then $x \neq 1$.

Problem 3 ([1], Prob 8, pp 115)
Prove using the proof by contradiction method that for all real numbers x, if $x^2 - 2x \neq -1$ then $x \neq 1$.

Problem 4 ([1], Prob 15, pp 115)
Given function $f, g, h : \mathbb{Z}_+ \to \mathbb{R}_+$ Recall that $f(n) = O(g(n))$ by definition if

$$\exists c \in \mathbb{R}_+ : \exists n_0 \in \mathbb{Z}_+ : \forall n \geq n_0 f(n) \leq cg(n).$$

Prove that if $f(n) = O(g(n))$ and $g(n) = O(h(n))$ then $f(n) = O(h(n))$.

Problem 5
Let us assume that the postal department has only two denominations of stamps: Rs 3 and Rs 5. We call a number n *postal* if we can create postage worth Rs n using the two denominations that are available. Prove using the Well Ordering Principle that every $n \geq 8$ is postal.

Problem 6 ([1], Prob 13, pp 127)
Use mathematical induction to prove that the number of subsets of a set of size n is 2^n.

Problem 7 ♠

We are given an array $A[n]$ containing 0s and 1s only. We want to sort it so that all the 0s appear before all the 1s, e.g. if $A = [1, 0, 0, 1, 0, 0]$ our output should be $A = [0, 0, 0, 0, 1, 1]$. Prove by induction that the procedure "Sort 0-1" correctly achieves this.

Set $p_0 \leftarrow 0$, $p_1 \leftarrow n - 1$.
while $p_0 < p_1$ **do**
    while $p_0 < n$ and $A[p_0] = 0$ **do**
        $p_0 \leftarrow p_0 + 1$
    end while
    while $p_1 > -1$ and $A[p_1] = 1$ **do**
        $p_1 \leftarrow p_1 - 1$
    end while
    if $p_0 < p_1$ **then**
        Swap $A[p_0], A[p_1]$.
    end if
end while

Problem 8
Suppose we are given a linked list with integers stored in each node and suppose that the linked list is maintained in sorted order. Write an algorithm for inserting a new element ℓ in the linked list. Prove the correctness of your algorithm using mathematical induction on the number of elements in the linked list.

Problem 9
Integer trees are a recursively defined data type. Every tree is either an empty tree, we denote it $\mathsf{EmptyTree}$ or a tuple of the form (ℓ, T_1, T_2) where ℓ is an integer and T_1 and T_2 are trees which we call the left and right subtrees respectively. Write an algorithm for finding the minimum integer in the tree. Assume for simplicity that all integers stored are non-negative. Prove the correctness of your algorithm using mathematical induction.

Problem 10 ([1], Prob 14, pp 127)
Prove that the strong principle of mathematical induction implies the weak principle of mathematical induction, i.e., if we accept that the deduction rule of the strong principle is sound then the deduction rule of the weak principle is also sound.

References

[1] K. Bogart, S. Drysdale, C. Stein, *Discrete Math for Computer Science Students.* 2005.