COL202: Discrete Mathematical Structures. I semester, 2020-21.
Minor exam
9 November 2020
Maximum Marks: 29

Important: You are allowed to use only one side of one page in a reasonable sized handwriting for each question. Q1.1 and Q1.2 must be fitted within one single side of one page. **If we can't understand what you have written we don't give you any marks**.

Problem 1.1 (2 marks)
The *bandwidth* of a graph is defined as follows: Find a numbering of the vertices of a graph such that the maximum difference between the numbers of two vertices connected by an edge in the graph is minimized. This minimum value is called the bandwidth of the graph. Write the following as a predicate: The bandwidth of $G = (V, E)$ is at least k. You *must* use the following notation: $\mathcal{F}$ is the set of functions from V to $\{1, \dots, |V|\}$; $\texttt{bandwidth}(G, k)$ is the name of the predicate you define.

Problem 1.2 (2 marks)
If a graph $G = (V, E)$ and non-negative integer k satisfy the predicate given below, what property do they satisfy? Write it out in words. Your answer must be concise and use graph theoretic terminology.

$$\texttt{pred}(G, k) := \forall S : S \subset V \Rightarrow (\exists x : x \in S \Rightarrow (\forall S' : S' \subseteq S \wedge (\forall y \in S' : (x, y) \in E) \Rightarrow |S'| \leq k))$$

Problem 2 (4 marks)
Prove that $3n^2+5n+1$ is odd for all $n \in \mathbb{N}$. Your proof *must* explicitly use the Well-Ordering Principle.

Problem 3 (5 marks)
Show that $\mathbb{N}^3 = \mathbb{N} \times \mathbb{N} \times \mathbb{N}$ has the same cardinality as $\mathbb{N}$.

Problem 4 (9 marks)
Given a graph $G = (V, E)$ and a vertex $a \in V$, consider the following algorithm, normally called "depth-first search":

1. Set a counter i to 1 and set the *current vertex* to a. Set all vertices except a as *unvisited* and set a as *visited*. Create a graph T with vertex a in it and no edges.
2. While the current vertex has a neighbour that is unvisited
   (a) Add to T the edge between the current vertex and the unvisited neighbour. (This will also add the neighbour to the vertex set of T).
   (b) Mark the neighbour as visited.
   (c) Make the neighbour the new current vertex.
   (d) Increment the counter i.
3. Move along any path from the current vertex to a till you find a node (it may be a itself) that has an unvisited neighbour.
4. If you find such a node, set that node to the current vertex and go to step 2. If not, the algorithm is over.

Prove by induction on i that if G is connected then when the algorithm ends T forms a normal spanning tree of G rooted at a. Your solution *must* proceed by showing that T forms a normal tree in G at the beginning and end of each iteration of the loop in Step 2.

Problem 5 (7 marks)

Suppose $G = (V, E)$ is a connected graph and $T = (V, E')$ is a normal spanning tree of G. Then prove that $uv \in E$ such that $u \leq_T v$ is a bridge of G if and only if $uv \in E'$ and there is no edge $xy \in E \setminus E'$ such that $x \in \lceil u \rceil$ and $y \in \lfloor v \rfloor$.