MATH1005/MATH6005 Discrete Mathematical Models
Final Exam, Semester 1, 2021

Photo Credit: Amithi Aparekka Liyanagamage.

Throughout this exam, we write $\mathbb{N}$ for the set of positive integers and $\mathbb{N}^\star$ for the set of non-negative integers; that is, $\mathbb{N} = \{1, 2, 3, \dots\}$ and $\mathbb{N}^\star = \{0, 1, 2, \dots\}$.

Problem 1 (10 marks) (a) Give an example of a statement that has the logical structure of an implication. Then write down the contrapositive, the converse and the inverse of your statement. Clearly label which statement is which (original, contrapositive, converse, inverse).

(b) Suppose that A and B are non-empty sets and that $R \subseteq A \times B$. What else must be true about R before we can say that R is an injective function?

(c) Let P denote the set of prime numbers. Let s denote the following statement:

$$\exists k \in \mathbb{N} \ \ \forall n \in \mathbb{N} \ \ \big((2^n - 1 \in P) \rightarrow (n \leq k)\big)$$

The statement $\neg s$ is a famous conjecture in Number Theory. Write down a statement that is logically equivalent to $\neg s$ and in which the symbol $\neg$ does not appear.

(d) Let $B = \{0,1\}^8$; for convenience we shall write $b_1b_2\ldots b_8$ as shorthand for $(b_1, b_2, \ldots, b_8)$. Let $\eta : B \to \mathbb{Z}$ be defined by the following rule

$$\forall b_1b_2\ldots b_n \in B \;\; \eta(b_1b_2\ldots b_n) = \\ (-1)^{b_1} \times \left(b_2 \times 2^6 + b_3 \times 2^5 + b_4 \times 2^4 + b_5 \times 2^3 + b_6 \times 2^2 + b_7 \times 2^1 + b_8 \times 2^0\right).$$

Let $\tau : B \to \mathbb{Z}$ be the function that maps each element of B to the integer that it represents using the 8-bit signed integer method (also known as the "two's complement method", and the "toggle-plus-one" method).

(i) Evaluate $\tau(10100011)$.

(ii) Use set-roster notation to describe the range of η and the range of τ.

(iii) Give two reasons why τ may be preferred to η as a method for representing integers in a computer.

Problem 2 (10 marks) (a) We define a sequence $(a_n)_{n\in\mathbb{N}}$ by

$$\begin{cases} a_1 = 1 \\ \forall n \in \mathbb{N}\ a_{n+1} = a_n\left(1 - \dfrac{1}{(n+1)^2}\right). \end{cases}$$

Use mathematical induction to prove that

$$\forall n \in \mathbb{N} \ \ a_n = \frac{n+1}{2n}.$$

(b) Use the logical equivalences below and the definitions of set operations to prove, or provide a counterexample to disprove, each of the following statements.

(i) For any universal set U and for any $A, B, C \in \mathcal{P}(U)$, we have

$$A \cap (B \cup C^c) = (A \cap B) \cup (A \setminus C).$$

(ii) For any universal set U and for any $A, B, C, D \in \mathcal{P}(U)$, we have:

$$\text{If } C \subseteq A \text{ and } D \subseteq B, \text{ then } (A \times B) \setminus (C \times D) = (A \setminus C) \times (B \setminus D).$$

Given any statement variables p, q, and r, a tautology t and a contradiction c, the following logical equivalences hold.

1. Commutative laws:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
2. Associative laws:	$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	$(p \vee q) \vee r \equiv p \vee (q \vee r)$
3. Distributive laws:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
4. Identity laws:	$p \wedge t \equiv p$	$p \vee c \equiv p$
5. Negation laws:	$p \vee \neg p \equiv t$	$p \wedge \neg p \equiv c$
6. Double negative law:	$\neg(\neg p) \equiv p$	
7. Idempotent laws:	$p \wedge p \equiv p$	$p \vee p \equiv p$
8. Universal bound laws:	$p \vee t \equiv t$	$p \wedge c \equiv c$
9. De Morgan's laws:	$\neg(p \wedge q) \equiv \neg p \vee \neg q$	$\neg(p \vee q) \equiv \neg p \wedge \neg q$
10. Absorption laws:	$p \vee (p \wedge q) \equiv p$	$p \wedge (p \vee q) \equiv p$
11. Negations of t and c:	$\neg t \equiv c$	$\neg c \equiv t$

Problem 3 (10 marks) (a) You and your friend are playing a board game. Each turn involves rolling three six-sided dice. The faces of each die are numbered 1, 2, 3, 4, 5, 6. Your friend, who is very reasonable but has not taken MATH1005/MATH6005, makes the following remark

> "These dice may be unfair. With three dice we can roll 16 different totals, so each total should appear every 16-th turn or so. However, we have been playing for hours and the total 18 has hardly ever come up."

In no more than five sentences, respond to your friend's statement. An excellent response will either agree or disagree with the reasoning in the statement, and will justify the position taken so clearly that your friend is likely to agree with you.

(b) A **PIN** is a string of 4 digits. A PIN is said to be **non-repeating** if no digit appears twice. For example, 3137 is a PIN, while 0216 and 7935 are non-repeating PINs.

What is the probability that, when a PIN is selected at random, it is a non-repeating PIN in which the digits appear in strictly increasing order?

(c) For any $k \in \mathbb{N}$, an *XY-path* of length k in the Euclidean plane is a sequence of points

$$\big((x_n, y_n)\big)_{n \in \{0,1,2,\ldots,k\}} \subseteq \mathbb{Z} \times \mathbb{Z}$$

such that

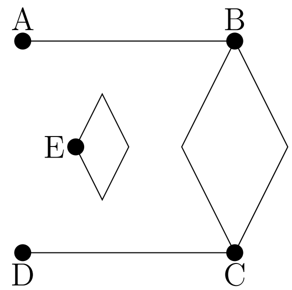
$$\forall n \in \{0,1,2,\ldots,k-1\} \;\; \Big((x_{n+1}, y_{n+1}) = (1 + x_n, y_n)\Big) \vee \Big((x_{n+1}, y_{n+1}) = (x_n, 1 + y_n)\Big);$$

we say that such a path *starts at* (x_0, y_0) and *ends at* (x_k, y_k).

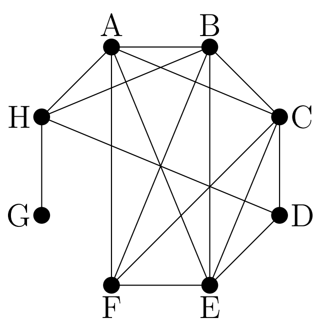
From all of the XY-paths that start at $(0,0)$ and end at $(4,6)$, one is chosen at random. What is the probability that the chosen XY-path visits the point $(2,0)$? Give your answer as a decimal, correct to two decimal places.

Problem 4 (10 marks) Graph Theory

(a) Let G be the graph shown below. Use set-roster notation to write down a set $V(G)$ and a multiset of size-2 multisets $E(G)$ that together describe G, and also write down an adjacency matrix that describes G.



(b) Let M be the graph shown below. Prove that M does not have a subgraph isomorphic to $K_{3,4}$.



(c) Let $n \in \mathbb{N}$. The hypercube of dimension n, denoted H_n, is the simple graph such that:

- $V(H_n)$ is the set of length-n bit strings;
- two length-n bit strings are adjacent if and only if the bit-strings differ in exactly one position.

(i) How many vertices does H_n have?

(ii) What is the degree of each vertex in H_n?

(iii) How many edges does H_n have?

(iv) If there exists a Hamilton circuit in H_4, write one down; if no Hamilton circuit exists in H_4, explain how you know.

Problem 5 (10 marks) (a) Describe the input and output of Dijkstra's algorithm.

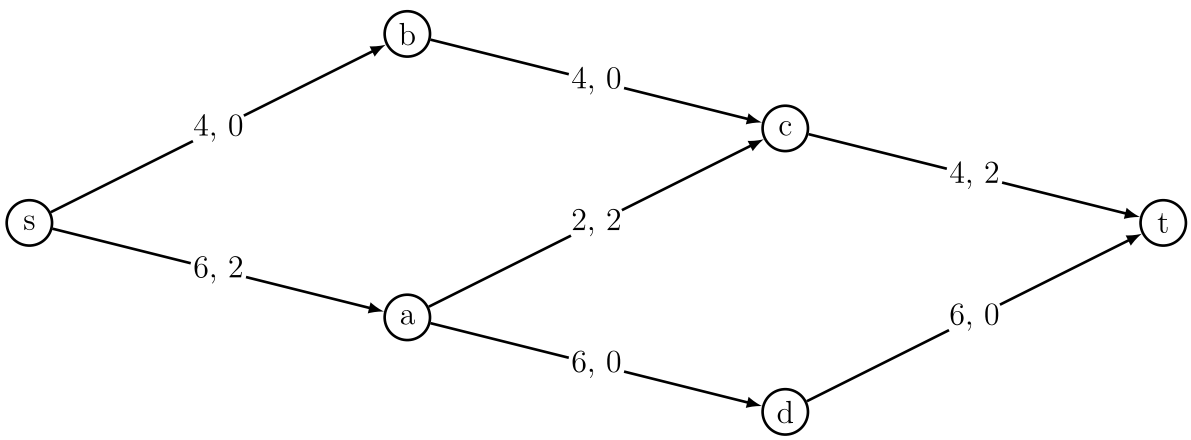
(b) Let S be the set of connected weighted simple graphs with 4 vertices. Prove or disprove the following statement:

$$\forall G \in S \;\; \exists!\, T \in S \;\; (T \text{ is a minimal spanning tree of } G).$$

(c) Suppose that you are given a weighted digraph G that represents a transport network, and your colleague says "At any time, we can make at most 20 units flow through this network." Your friend then shows you an example of a flow of volume 20 across the network.

One way to check your colleague's statement is to apply one of the algorithms we learned in the course. Describe another method by which you may determine whether your colleagues statement is true or false, and justify why your method will allow you to be confident in your answer.

(d) Use the vertex labelling algorithm described in the course to find a maximum flow function for the transport network shown below (pseudocode for this algorithm is given at the end of the exam paper). The first incremental flow f_1 is shown in the first row of the table at the bottom of the page, and the cumulative flow F_1 is shown in the graph. Write down the subsequent incremental flows in the table (use only as many rows as you need).



incremental flow label	path of incremental flow	volume of incremental flow
f_1	$s\ a\ c\ t$	2

Problem 6 (10 marks) (a) In no more than three sentences, explain how an internet is modelled by some type of graph (called a webgraph) for the purposes of the PageRank algorithm. An excellent answer will detail what type of graph is used, what the vertices represent, and what the edges represent?

(b) The PageRank algorithm may be understood to be following the movement of "The Random Surfer". In no more than five sentences, explain how The Random Surfer moves around the internet.

(c) Let $n \in \mathbb{N}$, let G be a webgraph with n vertices, let $\alpha = 0.15$ and let M denote the modified transition matrix (in the PageRank algorithm) determined by G and α. In the box below, write down an equation that completes the definition of the PageRank vector PR for G and α.

The PageRank vector PR is the unique $n \times 1$ matrix such that its entries sum to 1 and the following equation is satisfied

(d) Let G be the webgraph with the adjacency matrix A shown below. Draw a picture of the graph G represented by A, and then write down the basic transition matrix T associated to A as part of the PageRank algorithm.

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

THIS IS THE END OF THE EXAM. PSEUDOCODE FOR THE VERTEX LABELLING ALGORITHM IS OVER THE PAGE.

Vertex labelling algorithm for finding a maximum flow function for a transport network

Input: Transport network D with capacity function C.

Output: A maximum flow function $F_{\max}$ for the network.

Method: Initialise F to the zero flow F_0. Initialize i to 1.

For $i = 1, 2, \ldots$ carry out stage i below to attempt to build an incremental flow f_i.

If stage i succeeds, define $F_i = F_{i-1} + f_i$ and proceed to stage $i+1$.

If stage i fails, define $F_{\max} = F_{i-1}$ and stop.

Stage i:

1. If $i > 1$, mark up the amended edge flows for F_{i-1}.
2. Mark up the levels for F_{i-1}, as explained below.
3. If t is assigned a level, stage i will succeed, so continue.

   If not, then stage i fails, so return above to define $F_{\max}$ and terminate.
4. Mark up labels for F_{i-1} as follows until t is labelled:
   (a) Label each level 1 vertex v with sk_{v}, where $k_{\mathrm{v}} = S((s,v))$. (see below for definition of S)
   (b) If t has level 2 or more now work through the level 2 vertices in alphabetical order, labelling each vertex v with uk_u, where
      - u is the alphabetically earliest level 1 vertex with $(u,v) \in E(D)$ and $S((u,v)) > 0$,
      - k_{v} is the minimum of $S((u,v))$ and the value part of u's label.

   (c) If t has level 3 or more now work through the level 3 vertices in a similar manner and so on.
5. Let p_i be the path $\mathrm{u}_0\mathrm{u}_1 \ldots \mathrm{u}_n$ where $\mathrm{u}_n = \mathrm{t}$ and for $0 < j \leq n$
   u_j has label $\mathrm{u}_{j-1}k_j$.

   Define f_i to be the incremental flow on p_i with flow value k_n.

End of Method

Levels and labels: At each stage of the vertex labelling algorithm levels and labels are associated afresh with the vertices of the network.

The **level** of a vertex is determined iteratively as follows:

- The source vertex s always has level 0.
- If $e = (s, x)$ and $S(e) > 0$ then x has level 1.
- If x has level n and $S((x, y)) > 0$ then y has level $n + 1$ provided it has not already been assigned a lower level.

The **label** on a vertex v of level $n \geq 1$ has the form uk, where u is a vertex of level $n - 1$ and $(u, v) \in E(D)$ is an edge on the path for a potential incremental flow through v with flow value k.

The algorithm assigns labels in ascending order of levels, and in alphabetical order within levels.

The spare capacity function S
For vertices u,v of D, where D has capacity and flow functions C, F:

$$S((\mathrm{u,v})) = \begin{cases} C((\mathrm{u,v})) - F((\mathrm{u,v})) & \text{if } (\mathrm{u,v}) \in E(D) \\ F((\mathrm{v,u})) & \text{if } (\mathrm{v,u}) \in E(D) \\ 0 & \text{otherwise.} \end{cases}$$

When (v,u) $\in E(D)$, $S((\mathrm{u}, \mathrm{v}))$ is called a **virtual capacity**.