Duration: **50 minutes**
Aids Allowed: **One *single*-sided *hand*written *8.5"×11"* aid sheet.**

Student Number: |_|_|_|_|_|_|_|_|_|_|

Last (Family) Name(s): ____________________

First (Given) Name(s): ____________________

---

*Do **not** turn this page until you have received the signal to start.*
In the meantime, please read the instructions below *carefully*.

This term test consists of 3 questions on 8 pages (including this one), printed on both sides of the paper. *When you receive the signal to start, please make sure that your copy of the test is complete, fill in the identification section above, and write your name on the back of the last page.*

Answer each question directly on the test paper, in the space provided, and use one of the "blank" pages for rough work. If you need more space for one of your solutions, use a "blank" page and *indicate clearly the part of your work that should be marked*.

In your answers, you may use without proof any theorem or result covered in lectures, tutorials, problem sets, assignments, or the textbook, as long as you give a clear statement of the result(s)/theorem(s) you are using. You must justify all other facts required for your solutions.

Write up your solutions carefully! In particular, use notation and terminology correctly and explain what you are trying to do —part marks *will* be given for showing that you know the general structure of an answer, even if your solution is incomplete.

If you are unable to answer a question (or part of a question), remember that you will get 10% of the marks for any solution that you leave *entirely blank* (or where you cross off everything you wrote to make it clear that it should not be marked).

MARKING GUIDE

Nº 1: ______/12

Nº 2: ______/10

Nº 3: ______/ 8

TOTAL: ______/30

Good Luck!

Question 1. [12 MARKS]

Suppose that $G = (V, E)$ is an undirected graph and that each edge $e \in E$ is assigned a weight $w(e)$ such that $w(e) = 2^i$ for some $i \geqslant 0$. Suppose further that no two edges in E have the same weight.

Recall that a _matching_ for G is a set of edges $M \subseteq E$ such that no two edges in M share a common vertex. Consider the following greedy algorithm for finding a maximum weight matching in G, where $E = \{e_1, \ldots, e_m\}$.

> Sort the edges so that $w(e_1) \geqslant w(e_2) \geqslant \cdots \geqslant w(e_m)$.
> $M \leftarrow \varnothing$ # M is the current set of edges in the matching
> $W \leftarrow 0$ # W is the current total weight of M
> **for** $i = 1, \ldots, m$: # Loop Invariant: ...
>   **if** e_i does *not* share a vertex with any edge in M:
> $M \leftarrow M \cup \{e_i\}$
> $W \leftarrow W + w(e_i)$
> **return** M

Give a Loop Invariant, and use it to write a detailed proof that the output M is a maximum matching (following the proof format outlined in class).

HINT: Recall that $2^0 + 2^1 + \cdots + 2^{a-1} < 2^a$, for all $a \geqslant 0$.

Use the space on this "blank" page for scratch work, or for any solution that did not fit elsewhere.
Clearly label each such solution with the appropriate question and part number.

Question 2. [10 MARKS]

Give a dynamic program to solve the following problem:

- **Input:** Positive integers a, b, c, d, n.
- **Question:** Are there nonnegative integers (w, x, y, z) such that $n = aw + bx + cy + dz$?

Your solution should include the following steps:

(a) Define a suitable array indexed by parameters that define subproblems.

(b) Give a recurrence for the array values, based on the recursive structure of the problem, and argue that your recurrence is correct.

(c) Write an iterative algorithm, based on the recurrence, to solve the problem for arbitrary inputs a, b, c, d, n.

HINT: There is nothing to optimize—use an array that simply stores whether or not a solution exists.

Use the space on this "blank" page for scratch work, or for any solution that did not fit elsewhere.
Clearly label each such solution with the appropriate question and part number.

Question 3. [8 MARKS]

Suppose that we are given a flow network based on the directed graph $G = (V, E)$ with source node s and sink node t, and with integer capacities $c(u, v)$ for each edge $(u, v) \in E$. Suppose f is a maximum flow in the network, and we are given the residual network G_f. Give an efficient algorithm for finding a minimum cut in G, analyse your algorithm's running time and prove that it is correct.

Use the space on this "blank" page for scratch work, or for any solution that did not fit elsewhere.
Clearly label each such solution with the appropriate question and part number.

On this page, please write nothing except your name.

Last (Family) Name(s): ______________________________

First (Given) Name(s): ______________________________

Total Marks = 30