Homework 0
HW0 is a self-assessment covering topics and techniques which are assumed knowledge for this course. If you are unfamiliar with these topics we encourage you to refresh it during week 1. It does not need to be submitted and will not be graded.

`PART I: Big-O notation and functions.`
`Suggested reading: [DPV] Chapter 0.`

Problem 1

Part (a). DPV 0.1(c)
$f(n) = 100n + \log n, g(n) = n + (\log n)^2$.

Which holds: (Pick one)
- ○ $f(n) = O(g(n))$
- ○ $g(n) = O(f(n))$
- ○ $f(n) = O(g(n))$ and $g(n) = O(f(n))$ Correct Answer.

Part (b). DPV 0.1(d)
$f(n) = n \log n, g(n) = 10n \log 10n$.

Which holds: (Pick one)
- ○ $f(n) = O(g(n))$
- ○ $g(n) = O(f(n))$
- ○ $f(n) = O(g(n))$ and $g(n) = O(f(n))$ Correct Answer.

Part (c). DPV 0.1(k)
$f(n) = \sqrt{n}, g(n) = (\log n)^3$.

Which holds: (Pick one)
- ○ $f(n) = O(g(n))$
- ○ $g(n) = O(f(n))$ Correct Answer.
- ○ $f(n) = O(g(n))$ and $g(n) = O(f(n))$

Part (d). DPV 0.1(ℓ)
$f(n) = \sqrt{n}, g(n) = 5^{\log_2 n}$.

Which holds: (Pick one)
- ○ $f(n) = O(g(n))$ Correct Answer.
- ○ $g(n) = O(f(n))$
- ○ $f(n) = O(g(n))$ and $g(n) = O(f(n))$

Problem 2

(**[DPV 0.2]**) Show that $g(n) = 1 + a + a^2 + \ldots + a^n$ is $O(a^n)$ when $a > 1$ and $O(1)$ when $a < 1$. (*Hint*: You may try to prove $g(n) = \frac{a^{n+1}-1}{a-1}$ at first.)

Solution:

We have

$$a \cdot g(n) - g(n) = a^{n+1} - 1,$$

which means

$$g(n) = \frac{a^{n+1} - 1}{a - 1}.$$

Next, when $a > 1$, we can write

$$g(n) = \frac{a^{n+1} - 1}{a - 1} = \frac{a}{a - 1} \cdot a^n - \frac{1}{a - 1}.$$

We need to find a constant C and n_0, and show that $\frac{a}{a-1} \cdot a^n - \frac{1}{a-1} < Ca^n$ for all $n \geq n_0$. It is easy to check that $C = \frac{2a}{a-1}$ and $n_0 = 1$ satisfies the inequality, so $g(n) = O(a^n)$ when $a > 1$

When $a < 1$, we can write

$$g(n) = \frac{1 - a^{n+1}}{1 - a} = \frac{1}{1 - a} - \frac{a^{n+1}}{1 - a}$$

.

We need to find a constant C and n_0, and show that $\frac{1}{1-a} - \frac{a^{n+1}}{1-a} < C$ for all $n \geq n_0$. It is easy to check that $C = \frac{2}{1-a}$ and $n_0 = 1$ satisfies the inequality, so $g(n) = O(1)$ when $a < 1$.

PART II: Graphs fundamentals.

Problem 3

For all parts, $G = (V, E)$ represents an undirected, simple graph (i.e.: no multiple edges and no loops).

(a) Denote by $deg(v)$, the degree of vertex v, the number of edges incident to v. Check that

$$\sum_{v \in V} deg(v) = 2|E|.$$

(b) Review the concepts of path, cycle, connectivity.

(c) G is said to be a *tree* if it is connected and have no cycles. Think why the following three conditions are equivalent:

(i) G is a tree.

(ii) G is connected and $|E| = |V| - 1$.

(iii) G has no cycles and $|E| = |V| - 1$.

(d) A vertex is called a *leaf* if it has degree one. Show that every tree has at least two leaves. Think of an example of a tree with exactly two leaves.

Solution:

(a) Note that every edge is connected to exactly two vertices, so we can write

$$\sum_{v \in V} deg(v) = \sum_{e \in E} 2 = 2|E|.$$

(b) These concepts appear in Chapters 3 and 4 in the book. We will review it again when we talk about graph algorithms.

(c) *There are many ways to prove that these statements are equivalent. You won't be asked to prove it during the class, but the insight is valuable when studying graph algorithms. One possible proof is outlined below.*

$(i) \to (ii)$. Since G is a tree, it is connected. We show by induction on $|V|$ that the number of edges is $|V| - 1$. Remove any edge from the graph, you get a set of two disjoint trees with $|V_1|, |V_2|$ vertices satisfying $|V_1| + |V_2| = |V|$ (why we can't have a connected graph?), use the induction hypothesis to conclude that:

$$|E| = |V_1| - 1 + |V_2| - 1 + 1 = |V| - 1.$$

$(ii) \to (iii)$. Assume the graph has cycles. Delete one edge from every cycle until no cycles are left. Note that this operation does not disconnect the graph, thus, the resulting graph is a tree on $|V|$ vertices and by the previous implication it must hold that $|E| = |V| - 1$.

$(iii) \to (i)$. It suffices to check our graph is connected. Assume it is not, Then every connected component is a tree (since it is cycle free). Denote by $n_1, n_2, \ldots, n_c$ the number of vertices in each connected component. We have:

$$|E| = \sum_{i=1}^{c} n_i - 1 = |V| - c.$$

The RHS must equal $|V| - 1$ which yields $c = 1$, i.e., our graph has one connected component, as desired.

(d) Use the equality from part (a). If there are no leaves then every vertex has degree ≥ 2, hence

$$\sum_{v\in V} deg(v) \geq \sum_{v\in V} 2 = 2|V| > 2(|V|-1)$$

which contradicts $(c)-(ii)$.

`PART III: Boolean functions.`

Problem 4

A boolean function takes n variables $\{x_i\}_{1\leq i\leq n}$ with values `true, false` (sometimes we use the set $\{0,1\}$ instead) and outputs a boolean value. We will study boolean functions in *conjunctive normal form* (CNF), i.e.: our function is the conjunction of m clauses, each of which is made of the disjunction of distinct literals.

For each example below, decide which functions are in CNF and find an assignment of the variables such that the corresponding function evaluates to `true`, if such assignment exists.

- $(x \vee y \vee z) \wedge (x \vee w) \wedge (y \vee \bar{w})$ In CNF. It is enough to take $x = y = T$, and any assignment to z, w, for the function to be true.
- $(\bar{x} \vee \bar{y}) \wedge (x) \wedge (z \vee \bar{z})$ In CNF. x must be true, and this implies that y must be false for the first clause to be true. Any value of z will make the last clause true.
- $x \wedge (y \wedge (z \vee \bar{w}))$ Not in CNF. We must have $x = y = T$, and then $z = T$ suffices.