CS 331: Algorithms and Complexity (Spring 2022)

Unique numbers: 51310/51315

Discussion Section 1 - Solution

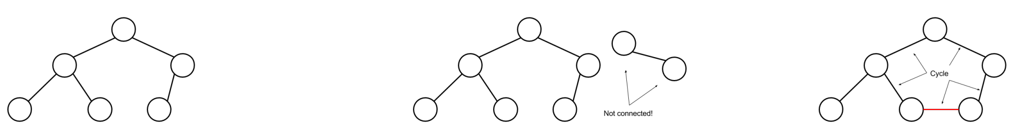
Problem 1

Recall that:

- A *graph* is a data structure with nodes and edges. Each edge connects two nodes.
- A *tree* is a connected acyclic graph.

In a connected graph, you can reach every node from every other node by following edges. In Figure 1b, the two nodes on the right are not reachable from the nodes on the left. Therefore, this is not a tree. Acyclic means that there are no cycles. In Figure 1c, a cycle is formed by the addition of the red edge.

We'll say that n is the number of nodes in a tree and m is the number of edges.



(a) This is a tree, with $n = 6$ and $m = 5$.

(b) The two nodes on the right hand side are disconnected. Not a tree.

(c) The addition of the red edge forms a cycle. This is not a valid tree.

Figure 1: Examples of trees and non-trees.

Prove that for any tree, the number of edges is one less than the number of nodes, i.e. $n - m = 1$.

Proof. By induction on the number of nodes, n.

Base case: $n = 1$. The tree has only one node. It cannot have any edges because the edge would have to go to itself, creating a cycle. Therefore, $m = 0$ and $n - m = 1 - 0 = 1$.

Induction Hypothesis: Assume that a tree with $n > 1$ nodes has m edges such that $n - m = 1$.

Inductive case Let T be a tree with $n' = n + 1$ nodes and m' edges. We have to prove that $n' - m' = 1$.

T can be decomposed into two parts: a tree with n nodes (call this tree S) and a single node with one edge to S (call this R). Such a decomposition exist by the following two lemmas.

Lemma 1. *In a tree T, there exist a node R that connects to exactly one edge.*

Proof. Proof by contradiction. Suppose that all the nodes in T have at least two edges. Then we can start at any node A and follow an edge to go to a new node B, then follow a new edge to go to a new node C. Since each node has at least two edges, we can repeat that until we go back to a node that we already visited. Therefore, T has a cycle, which is a contradiction since T is a tree. □

Lemma 2. *In a tree T, let R be a node that connects to exactly one edge, let S be the graph after removing R from T, then S is a tree.*

Proof. S is acyclic because T is acyclic and removing a node will not create a cycle. S is connected because T is connected and removing the node R will not break any paths in T, except for paths to R. □

By the induction hypothesis, S has $n - m = 1$ i.e, $m = n - 1$. R has exactly one node and one edge. Since T is just S with exactly one more node and one more edge, we have $n' - m' = (n + 1) - (m + 1) = (n + 1) - (n - 1 + 1) = 1$. This completes the proof.

□

Problem 2

Compression algorithms take in data and attempt to store it in a form that takes less space. These are sometimes separated into *lossy* algorithms, which can only return an approximation of the data, and *lossless* algorithms, which always return a perfect copy of the data.

Every once in a while, someone claims to have invented a "perfect" compression algorithm: one that is both lossless and guarantees compression (i.e. if you give it n bits, it will always give you a compressed form that is $n-1$ bits or smaller).

Prove that such an algorithm cannot possibly exist.

Hint: Remember that the algorithm is guaranteed to be both compressing *and* lossless. Start by assuming that the algorithm is always compressing, and show that there it cannot possibly be lossless (thus creating a contradiction).

Proof. By contradiction. Suppose such an algorithm did exist. Let us consider all two-bit files: 00, 01, 10, and 11. When compressed, we are guaranteed to get one-bit files back. The one-bit files are 0 and 1.

This yields a contradiction: there are only two compressed files, but four uncompressed files, and so the algorithm is not lossless. This contradicts our assumption that the algorithm is both lossless and guarantees compression.

□

Problem 3

Consider an $n \times n$ chess-board ($n \geq 1$). Let the bottom-left square be colored black. Prove that a bishop placed on that square can go to any black colored square on the chess-board.

Proof. Let a_{ij} represent the square occupying the i-th row and j-th column in the chess board. Let a_{11} be the square that the bishop is placed on. Then, $1 \leqslant i, j \leqslant n$.

This proof is by induction on dimension of the chess board, n.

Base case: $n = 1$. In this situation, the smaller chess-board is nothing but a_{11}, a single black square. So it is trivial to see that the bishop can go to all the black squares in the chess-board.

Induction Hypothesis: A bishop placed on a_{11} can go to all the black squares in the chess-board when $n = k$.

Induction Hypothesis: We have to prove that a bishop placed on a_{11} can go to all the black squares in the chess-board when $n = k + 1$.

> Let a_{xy} be a black square in the $(k+1)$-th row or column of the chess-board. I.e., either $x = k+1$ or $y = k+1$. Without loss of generality, let $x = k+1$. From the geometry of the board, if $a_{(x-1)(y-1)}$ and $a_{(x-1)(y+1)}$ exist, they will both be black colored squares.
>
> By the induction hypothesis, both of them can be reached by the bishop because both $a_{(x-1)(y-1)}$ and $a_{(x-1)(y+1)}$ (if exists) are part of $k \times k$ chess-board.
>
> From the rules of chess, it is possible for the bishop to move from both of those squares, provided they exist, to a_{xy}. Thus, any black square in the $(k+1)$-th row or column of the chess-board can be accessed by the bishop.

Thus the bishop placed on a_{11} can go to all the black squares for any $n \times n$ chess-board. $\square$

Problem 4

Suppose that you have a chessboard and a set of dominoes, where each domino covers two squares on the chessboard (one black and one white). A *tiling* of the chessboard is an arrangement of dominoes such that every square on the chessboard is covered, and every domino covers two adjacent squares: no dominoes are "hanging off the edge," and no dominoes are placed diagonally.

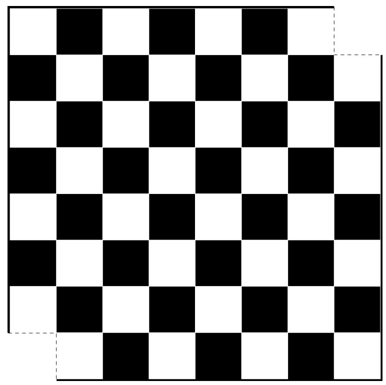


Figure 2: A chessboard with opposite corners removed.

Now take the chessboard and remove two corner squares of the same color, as seen in Figure 2. Prove that this new board cannot be tiled with dominoes—that is, any attempt to cover the chessboard with dominoes must always have either an uncovered square or a domino hanging off the edge.

Proof. By contradiction. We assume that this board can be tiled with dominos.

Each domino covers one black square and one white square. On the full chessboard, there are 32 black and 32 white squares. When we remove opposite corners, we end up with either 30 black and 32 white, or 32 black and 30 white squares. Without loss of generality, suppose we end up with 30 black and 32 white squares. In this case, we need 32 dominos to cover 32 white squares. Since each domino covers one white and one black square and since we have only 30 black squares, so there will be two dominos hanging off the edge. This contrdicts our assumption that this board can be tiled.

□