

DIS 3A

1 Trees

Recall that a *tree* is a connected acyclic graph (graph without cycles). In the note, we presented a few other definitions of a tree, and in this problem, we will prove two fundamental properties of a tree, and derive two definitions of a tree we learned from the note based on these properties. Let's start with the properties:

(a) Prove that any pair of vertices in a tree are connected by exactly one (simple) path.

(b) Prove that adding any edge between two vertices of a tree creates a simple cycle.

Now you will show that if a graph satisfies either of these two properties then it must be a tree:

(c) Prove that if every pair of vertices in a graph are connected by exactly one simple path, then the graph must be a tree.

(d) Prove that if the graph has no simple cycles and has the property that the addition of any single edge (not already in the graph) will create a simple cycle, then the graph is a tree.

2 Touring Hypercube

In the lecture, you have seen that if G is a hypercube of dimension n , then

- The vertices of G are the binary strings of length n .
- u and v are connected by an edge if they differ in exactly one bit location.

A *Hamiltonian tour* of a graph is a sequence of vertices v_0, v_1, \dots, v_k such that:

- Each vertex appears exactly once in the sequence.
- Each pair of consecutive vertices is connected by an edge.
- v_0 and v_k are connected by an edge.

(a) Show that a hypercube has an Eulerian tour if and only if n is even.

(b) Show that every hypercube has a Hamiltonian tour.

3 Graph Coloring

Prove that a graph with maximum degree at most k is $(k + 1)$ -colorable.