THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

CS 2312: Lab 10

SCHOOL OF ENGINEERING & APPLIED SCIENCE

Lab 10

- Show how TAs would solve past HW problems
- Review useful techniques and best practices in formal proofs
- Midterm Q&A

Single Cycle Graph

Consider a connected graph with $n \geq 3$ vertices.

Show that the graph has exactly one cycle if and only if it has n edges.

Key idea: Trees have one less edge and one less cycle.

THE GEORGE WASHINGTON UNIVERSITY
WASHINGTON, DC

Givens

1. The graph is connected
2. The graph has three or more vertices
3. Depending on the proof direction:
    a. The graph has exactly one cycle
    b. The graph has n edges

Forward Direction

- We are given a connected graph, *G*, with three or more vertices and exactly one cycle.
- There are two paths between neighboring vertices *a* and *b* in the cycle.
- Therefore, removing edge *(a, b)* does not disconnect the graph.
- The new graph, *G`*, is connected and has one less cycle; it is connected and acyclic.

Forward Direction

- ➢ *G`* is connected and acyclic, therefore it is a tree.
- ➢ *G`* is a tree, therefore it has *n-1* edges.
- ➢ Since *G`* is *G* with one edge removed, *G* must have one more edge and has *n* edges.

The forward direction is proven.

Reverse Direction

- We are given a connected graph, *G*, with three or more vertices and *n* edges.
- Assume: *G* does not have exactly one cycle.
  - Case 1: *G* has no cycles
  - Case 2: *G* has more than one cycle

Case 1

- ➢ There is no cycle, so *G* is connected and acyclic.
- ➢ Therefore, *G* is a tree and has $n-1$ edges.
- ➢ Contradiction!

Case 2

- *G* has more than one cycle, so *G* has at least two cycles.
- We can remove each cycle by removing an edge from the cycle as before.
- We will have removed at least two edges to make *G`*, therefore *G`* has at most *n-2* edges.
- *G`* is a tree, since it is connected and acyclic.
- *G`* has *n-1* edges, since it is a tree.
- Contradiction!

THE GEORGE WASHINGTON UNIVERSITY
WASHINGTON, DC

Reverse Direction

- ➢ All cases lead to contradictions, therefore our assumption that *G* doesn't have exactly one cycle is false.
- ➢ *G* must have exactly one cycle.

The reverse direction is proven.

Q.E.D.

Takeaways

- Comparing the graph to one we have worked with before helped.
- Remembering properties/lemmas shown in class was useful.

Odd Cycles

Every cycle in graph *G* has an odd length.

Show that at least one vertex in *G* has a degree of at most two.

We can use the contrapositive.

Show that if each vertex has a degree of three or more, *G* must have an even cycle.

Key idea: A maximal path can help us find cycles.

THE GEORGE WASHINGTON UNIVERSITY
WASHINGTON, DC

Givens

1. Each vertex in G has a degree of three or more.

Odd Cycles

- Consider a maximal path, a path that cannot be extended without repeating a vertex.
- Consider vertex *A* at one end of the path.
- *A* must have three neighbors, *B*, *C*, and *D*, since every vertex in *G* has a degree of three or more.
- The neighbors of *A* must be in the path, otherwise the path could be extended.
- We will label the neighbors such that they appear in the order *B*, *C*, then *D* in our path.

Odd Cycles

- Consider the two cases:
  - The length of either path $B \to C$ or $C \to D$ is even
  - The length of both paths $B \to C$ and $C \to D$ are odd

Case 1

- If the length of path $B \rightarrow C$ is even, then cycle $(A, B, \ldots, C, A)$ has an even length $(even + 2 = even)$.
- The same can be shown when path $C \rightarrow D$ is even.
- Therefore, the graph has an even cycle in this case.

Case 2

- If the lengths of both paths $B \to C$ and $C \to D$ are odd, then cycle $(A, B, \ldots, C, \ldots, D, A)$ has an even length *(odd + odd + 2 = even)*.
- Therefore, the graph has an even cycle in this case.

Odd Cycles

- ➢ In all cases we have shown that if each vertex in G has a degree of three or more, G must have an even cycle.
- ➢ By contrapositive, this proves that if each cycle in G has an odd length, then G must have a vertex with a degree of at most two.

Q.E.D.

Takeaways

- The contrapositive can give use a different perspective on a problem and potentially make the proof easier.
- Maximal paths can show that a cycle must exist if the degree of each vertex is two or more.
- Constructing paths and cycles in a graph can be a useful technique.

Creating a Directed Acyclic Graph (DAG)

Let G be a connected, undirected graph.

Show that we can assign a direction to each edge in G, turning it into a directed graph G', such that:

1. There exists a vertex, from which all edges only lead away.
2. There exists another vertex, toward which all edges only lead.
3. The directed graph G' should contain no directed cycles.

Creating a DAG

We can use a proof by construction.

We will show an algorithm which takes an arbitrary *G* and builds a *G`* which fits our conditions.

Key idea: We can avoid creating cycles by assigning a uniform direction to the edges in *G`*.

Givens

- G is a connected graph

Creating a DAG

- ➢ Assign some order to the vertices of G $(v_1, v_2, \ldots, v_n)$
- ➢ Make each directed edge point from the vertex closer to the beginning of the order to the vertex closer to the end.

We can now show that the algorithm produces a *G`* with the desired properties.

There Exists a Source

- The first vertex in our order, v_1, will only have outgoing edges because all edges point toward the end of the order.
- Therefore, v_1 is a source in G`.

There Exists a Sink

- The last vertex in our order, v_n , will only have incoming edges because all edges point toward the end of the order.
- Therefore, v_n is a sink in $G`$.

There Is No Cycle

- ➢ A directed cycle requires a path from some vertex *a* to some vertex *b* and a path from *b* to *a*.
- ➢ Take two vertices *a* and *b* in *G`*, such that there is a path from *a* to *b*.
- ➢ This implies that *b* is further towards the end of the order than *a*, since each edge in the path moves down the order.
- ➢ There cannot be a path from *b* to *a*, since no edge points up the order.
- ➢ Therefore, *G`* does not have a cycle.

Creating a DAG

➢ We have presented an algorithm which produces a DAG from any connected, undirected graph.

Q.E.D.

Takeaways

- We can use an algorithm to show that some graph must exist and prove properties about that graph.

Overall Takeaways

- Different proof methods can give you new perspectives on a problem. Think about the techniques you know and whether any of them would make your work easier.
- Keeping track of your givens and the statement you are proving can make a problem clearer.

Overall Takeaways

- Make sure that each statement logically follows from givens, lemmas and properties shown in class, or other statements in your proof.
- End problems with a statement which explains of why your work is sufficient to complete the proof.

Q&A Time!

- Midterm on 11/21
- Are there any topics you would like to review?