Final Exam Thursday 6/12 11:30am
Two sheets of notes allowed - no electronics
See the piazza post @630 with combined week 10 office hours.

44 Sample Final Questions are posted, solutions to follow

Week 10 Take Home quiz due Friday 6/5 - before class.

Extra Credit graph slides @631 due Thursday 6/4.

Wednesday (6/4) 6-7:30pm (EBU3B 4122)
Wednesday (6/4) 8-10pm (Homeplate)
Thursday (6/5) 8-10pm (Homeplate)
Friday (6/6) 12-1pm (EBU3B 4122)
Friday (6/6) 2-5pm at The Loft (with tutors / TAs)
Final Exam Study Session 1: Fri. 6/6 at 6-8pm (CENTER 115)
Saturday (6/7) 5-6pm (EBU3B 4122)
Final Exam Study Session 2: Sat. 6/7 at 6-8pm (PCYNH 106)
Final Exam Thursday 6/12 11:30am (CENTER 119)
Two sheets of notes allowed - no electronics
Greed can be good, can be bad
Minimum Spanning Trees
Hamiltonian Cycles
Traveling Salesman Problem - when greed is good enough
Identify the true statement

A. Greedy colorings always find proper colorings using the fewest possible colors

B. Finding $\chi(G)$ is an easy problem in general (please put lots of them on the final.)

C. If you choose an ordering and greedily color with $k$ colors, then $\chi(G) = k$

D. If you choose an ordering and greedily color with $k$ colors, then $\chi(G) \leq k$

E. If you choose an ordering and greedily color with $k$ colors, then $\chi(G) \geq k$
Greedy colorings vs Minimum Proper Colorings

Identify the true statement

A. Greedy colorings always (sometimes if you are lucky and choose a good ordering) find proper colorings using the fewest possible colors.

B. Finding $\chi(G)$ is an easy (hard) problem in general. Finding $\chi(G)$ is an minimization problem and is NP-hard in general. There are cases when it is easy - like find $\chi(T)$ where $T$ is a tree on one million vertices.

C. If you choose an ordering and greedily color with $k$ colors, then $\chi(G) = k$ (definitely NO)

D. If you choose an ordering and greedily color with $k$ colors, then $\chi(G) \leq k$
   If you find a proper coloring with $k$ colors, then $\chi(G)$ can be $k$ or it could be smaller!

E. If you choose an ordering and greedily color with $k$ colors $\chi(G) \geq k$ (also NO)
Spanning Trees

Given any graph, we can choose some of the edges to form a subgraph that is a tree. If all of the vertices are touched by at least one edge, then the tree is called spanning.

Does the green subgraph form a spanning tree?

A. No, It is a tree but not spanning  
B. Yes, it’s a tree  
C. Yes it is a spanning tree  
D. Its not a tree  
E. It is not a subgraph
Given any graph, we can choose some of the edges to form a subgraph that is a tree. If all of the vertices are touched by at least one edge, then the tree is called spanning.

Does the green subgraph form a spanning tree?

A. No, It is a tree but not spanning  
B. Yes, it’s a tree  
C. Yes it is a spanning tree  
D. Its not a tree  
E. It is not a subgraph
Given any graph, we can choose some of the edges to form a subgraph that is a tree. If all of the vertices are touched by at least one edge, then the tree is called spanning.

Does the green subgraph form a spanning tree?
A. No, It is a tree but not spanning  B. Yes, it’s a tree  C. Yes it is a spanning tree  D. Its not a tree  E. It is not a subgraph
Spanning Trees

Given any graph, we can choose some of the edges to form a subgraph that is a tree. If all of the vertices are touched by at least one edge, then the tree is called spanning.

Does the green subgraph form a spanning tree?

A. No, It is a tree but not spanning  
B. Yes, it’s a tree  
C. Yes it is a spanning tree  
D. Its not a tree  
E. It is not a subgraph
Spanning Trees

Given any graph, we can choose some of the edges to form a subgraph that is a tree. If all of the vertices are touched by at least one edge, then the tree is called spanning.

Does the green subgraph form a spanning tree?

A. No, It is a tree but not spanning  
B. Yes, it’s a tree  
C. Yes it is a spanning tree  
D. Its not a tree  
E. It is not a subgraph
Some graphs have weights, which could represent the cost of using it.
Minimum Spanning Trees

Some graphs have weights, which could represent the cost of using it. Can we find a spanning tree that uses the lowest total weight (add the weights of the edges in our tree)?
MST - Step 1. Sort the edges
MST - Step 2. Add an edge, unless it forms a cycle

AE 1
HI 2
CF 2
CG 3
FG 4
DG 5
GZ 6
BF 7
EH 8
AH 9
DZ 10
EF 11
JZ 12
AB 13
FI 14
FH 15
IJ 16
FJ 17
GJ 18
BE 18
BC 21
CD 23

Rubalcaba (rrrubalcaba@eng.ucsd.edu)
MST - Step 2. Add an edge, unless it forms a cycle

Rubalcaba (rrrubalcaba@eng.ucsd.edu)

6/4/2014 15 / 54
MST - Step 2. Add an edge, unless it forms a cycle
MST - Step 2. Add an edge, unless it forms a cycle

Rubalcaba (rrrubalcaba@eng.ucsd.edu)
MST - Step 2. Add an edge, unless it forms a cycle

Rubalcaba (rrrubalcaba@eng.ucsd.edu)
MST - Step 2. Add an edge, unless it forms a cycle
MST - Step 2. Add an edge, unless it forms a cycle

Rubalcaba (rrrubalcaba@eng.ucsd.edu)
MST - Step 2. Add an edge, unless it forms a cycle

Rubalcaba (rrrubalcaba@eng.ucsd.edu)
MST - Step 2. Add an edge, unless it forms a cycle

Rubalcaba (rrrubalcaba@eng.ucsd.edu)
MST - Step 2. Add an edge, unless it forms a cycle

Rubalcaba (rrrubalcaba@eng.ucsd.edu)
MST - Step 2. Add an edge, unless it forms a cycle

Rubalcaba (rrrubalcaba@eng.ucsd.edu)
MST - Step 2. Add an edge, unless it forms a cycle

Rubalcaba (rrrubalcaba@eng.ucsd.edu)
MST - Step 2. Add an edge, unless it forms a cycle
MST - Step 3. Stop when you have a tree
MST - Step 3. Stop when you have a tree (N-1) edges, connected - any new edge creates a cycle.

A E 1 ✓
H I 2 ✓
C F 2 ✓
C G 3 ✓
F G 4 ✓
D G 5 ✓
G Z 6 ✓
B F 7 ✓
E H 8 ✓
A H 9 ✓
D Z 10 ✗
E F 11 ✓
J Z 12 ✓
A B 13 ✓
F I 14 ✓
F H 15 ✗
I J 16 ✗
F J 17 ✗
G J 18 ✗
B E 18 ✗
A. Add the edge  
D. Don’t add the edge
A. Add the edge  
D. Don’t add the edge
Traveling Salesman Problem - Sorted Edges

B E 1 ✓
C F 2 ✓
C E 3 ✓
A C 4
B C 5
E F 6
A E 7
B D 8
A D 9
E D 10
D F 11
F B 12
A B 13
A F 14
C D 15

A. Add the edge      D. Don’t add the edge
### Traveling Salesman Problem - Sorted Edges

<table>
<thead>
<tr>
<th>Edge</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>B E</td>
<td>1</td>
</tr>
<tr>
<td>C F</td>
<td>2</td>
</tr>
<tr>
<td>C E</td>
<td>3</td>
</tr>
<tr>
<td>A C</td>
<td>4</td>
</tr>
<tr>
<td>B C</td>
<td>5</td>
</tr>
<tr>
<td>E F</td>
<td>6</td>
</tr>
<tr>
<td>A E</td>
<td>7</td>
</tr>
<tr>
<td>B D</td>
<td>8</td>
</tr>
<tr>
<td>A D</td>
<td>9</td>
</tr>
<tr>
<td>E D</td>
<td>10</td>
</tr>
<tr>
<td>D F</td>
<td>11</td>
</tr>
<tr>
<td>F B</td>
<td>12</td>
</tr>
<tr>
<td>A B</td>
<td>13</td>
</tr>
<tr>
<td>A F</td>
<td>14</td>
</tr>
<tr>
<td>C D</td>
<td>15</td>
</tr>
</tbody>
</table>

A. Add the edge  
D. Don’t add the edge
### Traveling Salesman Problem - Sorted Edges

<table>
<thead>
<tr>
<th>Edge</th>
<th>Weight</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>BE</td>
<td>1</td>
<td>✓</td>
</tr>
<tr>
<td>CF</td>
<td>2</td>
<td>✓</td>
</tr>
<tr>
<td>CE</td>
<td>3</td>
<td>✓</td>
</tr>
<tr>
<td>AC</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>BC</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>EF</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>AE</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>BD</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>AD</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>ED</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>DF</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>FB</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>AB</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>AF</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>CD</td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>

A. Add the edge  
D. Don’t add the edge
A. Add the edge  
D. Don’t add the edge
A. Add the edge  D. Don’t add the edge
A. Add the edge       D. Don’t add the edge
Traveling Salesman Problem - Sorted Edges

A. Add the edge
D. Don’t add the edge
A. Add the edge  
D. Don’t add the edge
A. Add the edge      D. Don’t add the edge
A. Add the edge

D. Don’t add the edge
A. Add the edge  
D. Don’t add the edge
A. Add the edge     D. Don’t add the edge
A. Add the edge       D. Don’t add the edge
Traveling Salesman Problem - Sorted Edges

A. Add the edge

D. Don’t add the edge
Traveling Salesman Problem - Sorted Edges

A. Add the edge
D. Don’t add the edge

Rubalcaba (rrrubalcaba@eng.ucsd.edu)
A. Add the edge    D. Don’t add the edge
Traveling Salesman Problem - Sorted Edges

A. Add the edge  D. Don’t add the edge
A. Add the edge  
D. Don’t add the edge
Traveling Salesman Problem - Sorted Edges

A. Add the edge  D. Don’t add the edge
A. Add the edge

D. Don’t add the edge
Traveling Salesman Problem - Sorted Edges

Options:

A. Add the edge

D. Don’t add the edge
Traveling Salesman Problem - Sorted Edges

A. Add the edge

D. Don’t add the edge

Rubalcaba (rrrubalcaba@eng.ucsd.edu)
For Q1-Q4, properly color the vertices of the graph using the *minimum* number of colors.

**Q1.** $P_2$

**Q2.** $\mu(P_2) = C_5$

**Q3.** $\mu(\mu(P_2)) = \mu(C_5) = \text{The Grötzsch graph}$

**Q4.** A random tree on 10 vertices.

**E.C.** $\chi[\mu(\mu(\mu(P_2)))] = \chi[\mu(\text{the Grötzsch graph})] =$