1. Use the order-requirement digraph to schedule the six tasks on two processors with the following priority lists:

![Diagram of task dependencies]

a. $T_1, T_2, T_3, T_4, T_5, T_6$

b. $T_6, T_5, T_4, T_3, T_2, T_1$
2. Use the order-requirement digraph to schedule the six tasks on three processors with the following priority lists:

```
  T_6  T_3  T_1  
  7    8    11   
  T_2  T_4  T_5  
  5    9    20   
T_5  
```

a. T_6, T_5, T_4, T_3, T_2, T_1

```

```

b. T_1, T_2, T_3, T_4, T_5, T_6

```

```

```
3. Use the order-requirement digraph to schedule the six tasks on two processors with the following priority list: \(T_1, T_2, T_3, T_4, T_5, T_6\)

![Diagram](image1.png)

4. Use the order-requirement digraph to schedule the six tasks on two processors with the following priority list: \(T_9, T_8, T_7, T_6, T_5, T_4, T_3, T_2, T_1\)

![Diagram](image2.png)
5. Use the order-requirement digraph to schedule the six tasks on three processors with the following priority list: $T_9, T_8, T_7, T_6, T_5, T_3, T_2, T_1$

Find the critical-path priority lists for the following:

6.
Independent Tasks & Bin-Packing Worksheet

1. Schedule independent tasks of length 8, 11, 17, 14, 16, 9, 2, 1, 18, 5, 3, 7, 6, 2, and, 1 on three processors, using the list-processing algorithm.

<table>
<thead>
<tr>
<th>4</th>
<th>8</th>
<th>12</th>
<th>16</th>
<th>20</th>
<th>24</th>
<th>28</th>
<th>32</th>
<th>36</th>
<th>40</th>
<th>44</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. Rearrange the tasks on problem #1 according to decreasing time and then schedule them on three processors, using the list-processing algorithm.

<table>
<thead>
<tr>
<th>4</th>
<th>8</th>
<th>12</th>
<th>16</th>
<th>20</th>
<th>24</th>
<th>28</th>
<th>32</th>
<th>36</th>
<th>40</th>
<th>44</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3. A radio station’s policy allows advertising breaks of no more than 135 seconds. Determine the minimum number of breaks into which ads of the following lengths will fit by applying each algorithm below: 80, 90, 130, 50, 60, 20, 90, 30, 30, 40. Use the bin templates found in your packet to show packings.
   a) next fit
   b) first fit
   c) worst fit
   d) next fit decreasing
   e) first fit decreasing
   f) worst fit decreasing
Coloring graphs: Extra Practice

Joanna is planning Thanksgiving dinner for her family. She is trying to create a seating arrangement that will prevent family squabbles from breaking out over dinner. Because Joanna just took Math 107 last year, she is able to create the chart below to help her: X’s indicate relatives who should not be sitting at the same table.

<table>
<thead>
<tr>
<th></th>
<th>Aunt Anne</th>
<th>Uncle Bert</th>
<th>Cousin Chelsea</th>
<th>Cousin Darrin</th>
<th>Grandma Elsie</th>
<th>Grandma Fran</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aunt Anne</td>
<td></td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Uncle Bert</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cousin Chelsea</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Cousin Darrin</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Grandma Elsie</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Grandma Fran</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

1) Create a graph that represents the family relations, using edges to denote members who shouldn’t sit together at dinner.

2. Color the graph and determine the fewest possible tables Joanne needs for her family dinner. How many tables are needed?

3. Give a seating chart that evenly spreads the family members among the required number of tables.
Homework Worksheet III

Schedule each of the following on two processors using the critical path scheduling algorithm:

1. [Diagram]

2. [Diagram]

3. [Diagram]
4. Schedule each of the following on three processors using the critical path scheduling algorithm:

5. Schedule each of the following on three processors using the critical path scheduling algorithm:

6. Schedule each of the following on three processors using the critical path scheduling algorithm: