- Homework 2 will be assigned before Thanksgiving Tuesday 11:59 pm (due date for projects)

- Proposal:
  - Basic idea (fixed)
  - Basic LP, ILP, ... (fixed)
  - Fundamental Questions/Motivation (can vary)
  - List of 3 or more questions/directions (can vary a lot over time)

E.g.,
  - Exam scheduling (basic idea) = fixed
  - Basic LP, ILP; graph coloring = fixed, will be refined
  - Fundamental Questions/Motivation: Given resources for exam (instructors time), students/faculty/TA
    hypothesis with shorter exam period, given policies that
    university can set, how does exam period lengths
    & conflicts can be achieved.
  - List of questions: (principles...)
    - might vary over time
  - What happens if large classes have individual section exams
  - Say we want to gauge for students between exam times
Practical Examples:

- LP resources products where each product has to be integer

   ILP, probably branch and bound, bounding is based on the LP relaxation?

- ILPs that can be solved directly from the LP relaxation:

  Weighted Bipartite Matching:
  \[
  \begin{align*}
  \text{max } & \sum c_i x_i \\
  \text{st } & A x \leq b, \ x \geq 0 \\
  \end{align*}
  \]

  (\( x \) components are integers)

  Claim: If we solve without integral constants LP, the simplex method always finds \( x \) optimal and integral.

  - Also: discuss bin packing, where take ILP and LP relaxation does nothing 😒

  Why do some problems:
  \[
  \begin{align*}
  \text{max } & \sum c_i x_i \\
  \text{st } & A x \leq b \\
  \end{align*}
  \]

  with \( x \in \mathbb{Z}^n \) \( \text{dictinaries} \)

  The simplex method:
  \[
  \begin{bmatrix} A | I \end{bmatrix} \begin{bmatrix} x_{\text{dec}} \\ x_{\text{slack}} \end{bmatrix} = b
  \]

  \[
  A_{B_{\text{slack}}} x_B + A_N x_N = b
  \]

  \[
  x_B = A_{B_{\text{slack}}}^{-1} (b - A_N x_N)
  \]

  and \( A_B, A_N \) some cols of \( [A_{\text{org}} | I] \)
Theorem: If \( I \) has \( k \) components, and \( A_B \) is any set of columns of \([A | I]\) that is square and \( A_B \) has an inverse, then \( A_B^{-1} I \) will have integer components.

If every minor of \( A = A_{ori} \) has determinant 0, 1, -1 ("totally unimodular")

\[
\begin{align*}
\text{max } & 4x_1 + 5x_2 \\
& x_1 + 2x_2 \leq 8 \\
& x_1 + x_2 \leq 5 \\
& 2x_1 + x_2 \leq 8 \\
\end{align*}
\]

\[
\begin{bmatrix}
12 & 1 & 0 & 0 \\
11 & 0 & 1 & 0 \\
21 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5
\end{bmatrix} =
\begin{bmatrix}
8 \\
5 \\
8
\end{bmatrix}
\]

Say basic \( x_1, x_2, x_3 \)

\[
A_B = \begin{bmatrix}
12 & 1 \\
1 & 0 \\
21 & 0
\end{bmatrix}
\]

from \( A \)

\[
\begin{bmatrix}
12 \\
11 \\
21
\end{bmatrix}
\]

from \( I \)

so \( \det A_B = 0, 1, -1 \) \( \implies \) \( \det \begin{bmatrix} 11 \\ 21 \end{bmatrix} = 0, 1, -1 \)

= 

Upshot: Look if some book/paper says your \( A \) is "totally unimodular." E.g. bipartite matching, network flows, etc.

Minors of \( \begin{bmatrix} 12 \\ 11 \\ 21 \end{bmatrix} \): 2x2 minors: pick 2 rows, 2 cols.
Bad news for bin pecking:

We have some bins

\[
\begin{array}{cccc}
& & & \\
& & & \\
& & & \\
& & & \\
& & & \\
20 & 20 & 16 & 25
\end{array}
\]

each bin has capacity

Have objects to put in bins \( \Delta \square \square \square \) \( \square \square \square \) \( \square \square \square \) \( \square \square \square \) \( \square \square \square \) \( \square \square \square \) each with size

Want to place each item in a bin.

Hospital Operating Rooms

- Room 1: 24 hours
- Room 2: 24 hours

Some operation to perform: 3 hours, 5 hours, 2 hours, etc.

Say we have 2 rooms, equal number of hours

Room 1: Capacity 1, object to \( a_1, a_2, \ldots, a_n \in [0, 1] \)
Room 2: Capacity 1, place room

Choose \( x_1, \ldots, x_n \)

\[ x_1 a_1 + \ldots + x_n a_n = 1 - \text{space} \]

min space

\[ x_1, \ldots, x_n, \text{space} \geq 0 \]

\[ x_1 \in \mathbb{Z}, x_2 \in \mathbb{Z}, \ldots, x_n \in \mathbb{Z} \]

What happens if?