

CARNEGIE MELLON UNIVERSITY 07-380

HOMEWORK 2

<https://www.cs.cmu.edu/~07380>

INSTRUCTIONS

- **Format:** Use the provided LaTeX template to write your answers in the appropriate locations within the *.tex files and then compile a pdf for submission. We try to mark these areas with STUDENT SOLUTION HERE comments. Make sure that you don't change the size or location of any of the answer boxes and that your answers are within the dedicated regions for each question/part. If you do not follow this format, we may deduct points.

You may also digitally annotate the pdf. Illegible handwriting will lead to lost points. However, we suggest that you try to do at least some of your work directly in LaTeX.

- **How to submit written component:** Submit to Gradescope a pdf with your answers. Again, make sure your answer boxes are aligned with the original pdf template.
- **Policy:** See the course website for homework policies, including late policy, and academic integrity policies.

Name	
Andrew ID	
Hours to complete all components (nearest hour)	

1 [10 pts] Planning

Consider a planning environment with six different operations (defined in the table below), starting state A , and goal condition $C \wedge D \wedge E$. Only one operation may be applied at a time, and we are trying to find the plan with the fewest number of operations.

	op1	op2	op3	op4	op5	op6
Precondition	A	B	A	A	A	A
Add	B	C, D, E	C	D	E	E
Delete						A

1. [4 pts] Run GraphPlan on this environment. Draw the **planning graph**, adding action levels and predicate levels until GraphPlan terminates.

Note: make sure to include the No-op actions for persistent states in your drawing. For your submission to this problem, you may do one of the following:

- Draw/annotate on top of the existing images in the pdf.
- Edit the `figures/graphplan.png` image file to add markings.

Hand drawing is acceptable, as long as it is clear and precise enough.

GraphPlan Graph

S_0

A_0

A

2. [2 pts] What plan is returned by GraphPlan?

Plan

3. [2 pts] Is that plan optimal?

☐ Yes ☐ No

4. [2 pts] List ALL pairs of exclusive operators in A_0 and ALL pairs of exclusive predicates in S_1 . Write “None” if none exist. Note: Remember that no-op counts as an action.

Exclusive Operators in A_0

Exclusive Predicates in S_1

2 [8 pts] Bayes the Bat's Day

Bayes the Bat (our new course's mascot!) is trying to organize his time in a day. Except for sleeping, eating, and (most importantly) playing 'Bayes-ball', he still has 8 hours to spare each day. Two important things in his life are *partying* and *doing homework*. In order to keep up with his classes, he has to spend **at least** 2 hours per day on homework. But he doesn't want to wear himself out, so the number of hours he spends partying should equal **at least** half the number of hours he spends on homework. On the other hand, he feels bad if he parties too much, so his party time **should not exceed** his homework time by more than 3 hours. Bayes the Bat knows that he will gain 1 unit of happiness for every hour he parties, and he will lose 0.5 unit of happiness for every hour he spends on homework. But he needs your help to figure out how much time he should spend on partying and how much time he should spend doing homework such that his happiness is **maximized**.

1. [3 pts] Write this problem as an LP in **inequality form** as defined in lecture. Define variable x_1 to be the hours spent partying, and variable x_2 to be the hours spent on homework. *Warning:* Be sure to strictly follow the inequality form, including the proper use of less than or equal, or you will lose points.

Inequality Form

2. [3 pts] Accurately plot the graphical representation of this linear program. Specifically:

- Plot the boundary of each halfspace as a line (no need to shade or draw normal vectors), and
- Plot the cost vector as an arrow with *magnitude one*, somewhere within the feasible region.

Do *not* draw; use a plotting tool such as Python matplotlib and include the resulting image here. Be sure to label the axes of your plot, including tick-marks. Display your plot with a *square* aspect ratio, e.g. in matplotlib: `plt.axis("equal")`. Additionally, zoom your plot to make the entire feasible region visible. Tip to plot a vector $[v_1, v_2]^T$ in matplotlib starting at some point (x_1, x_2) :

```
plt.quiver(x1, x2, v1, v2, angles="xy", scale_units="xy", scale=1)
```

Tip to properly control scaling using a specific width and height:

```
plt.figure(figsize=(width,height))
```

We have included some starter code for you in `figures/plot_graph.py`. You will need to modify the `plot_graph()` function and fill in code for `compute_unit_length()` yourself. **Note:** For the sake of grading please let your x_1 axis range over $[-2, 10]$ and your x_2 axis over $[-4, 8]$.



3. [2 pts] Find the optimal solution to the LP problem. Give the solution hours he should spend in party and homework respectively and his happiness.

Party	Homework	Happiness

3 [6 pts] Graphing LPs

For the inequality form of a linear program, and a given A matrix and $\mathbf{b}$ vector,

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & A\mathbf{x} \preceq \mathbf{b} \end{aligned}$$

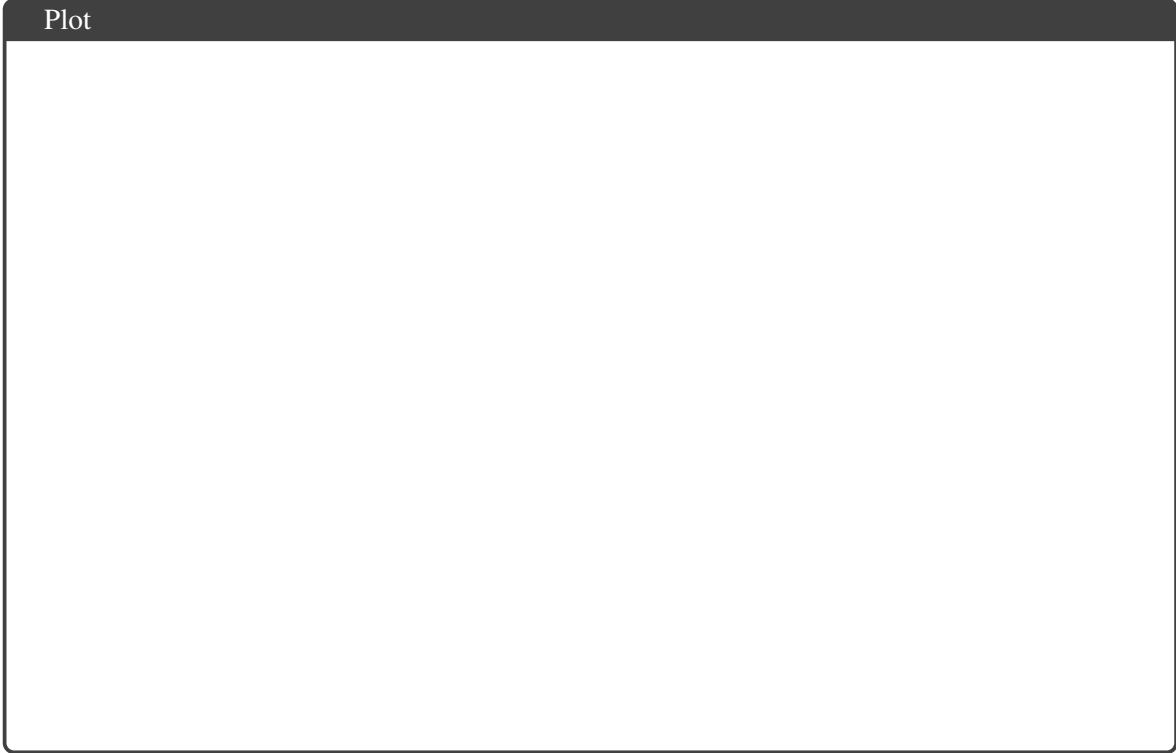
For each row i of A and $\mathbf{b}$, accurately plot 1) the line $a_{i,1}x_1 + a_{i,2}x_2 = b_i$ and 2) the vector $[a_{i,1}, a_{i,2}]^T$ as an arrow beginning at any point on its respective line with *magnitude one*. Tip to plot a vector $[v_1, v_2]^T$ in matplotlib starting at some point (x_1, x_2) : `plt.quiver(x1, x2, v1, v2, angles="xy", scale_units="xy", scale=1)`. Tip to properly control scaling using a specific width and height: `plt.figure(figsize=(width,height))`

Do *not* draw; use a plotting tool such as Python matplotlib and include the resulting image here. Be sure to label the axes of your plot, including tick-marks. Display your plot with a *square* aspect ratio, e.g. in matplotlib: `plt.axis("equal")`. Additionally, zoom your plot or adjust the axes such that all of the vectors are visible. You do not need to shade the feasible regions. We have included some starter code for you in `figures/plot_graph.py`. You will need to modify the `plot_graph()` function and fill in code for `compute_unit_length()` yourself.

1. [3 pts]

$$A = \begin{bmatrix} 3 & 5 \\ 7 & 6 \\ 12 & 6 \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} 10 \\ 17 \\ 27 \end{bmatrix}$$

Plot

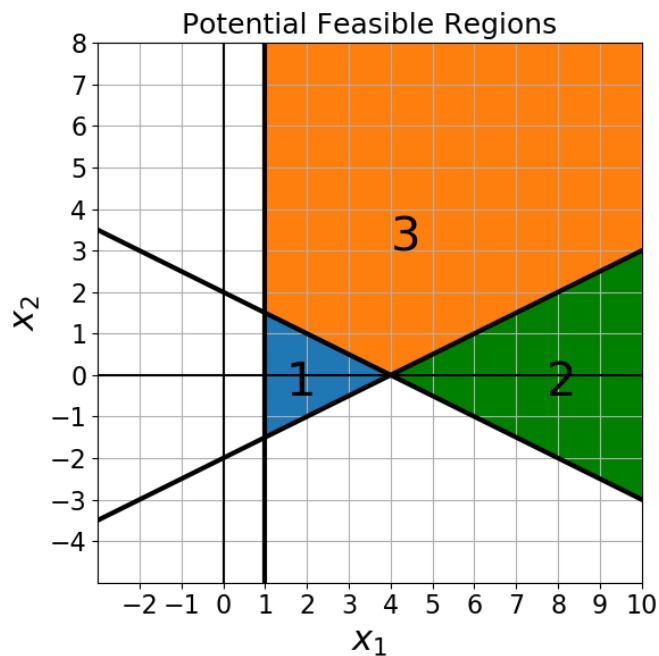


2. [3 pts]

$$A = \begin{bmatrix} -2 & -1 \\ 2 & 5 \\ 7 & 2 \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} -4 \\ 10 \\ 11 \end{bmatrix}$$

Plot

4 [9 pts] Feasible Regions



In this problem, you are given a graph with constraint boundary lines (**bolded**) and potential feasible regions. You may assume shaded regions at the edge of the plot continue to infinity. Provide a corresponding A and b using the inequality form below which defines each feasible region in the boxes below. **You must include all three constraints.**

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & A\mathbf{x} \preceq \mathbf{b} \end{aligned}$$

1. [3 pts] Feasible Region 1

A

b

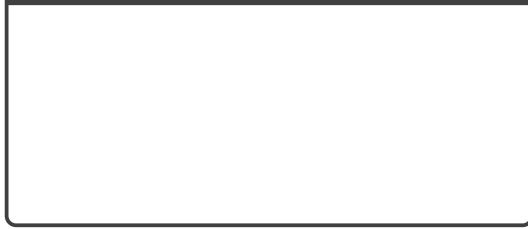
2. [3 pts] Feasible Region 2

A

b

3. [3 pts] Feasible Region 3

A



b



5 Collaboration Policy

After you have completed all other components of this assignment, report your answers to the following collaboration questions.

1. Did you receive any help whatsoever from anyone in solving this assignment? If so, include full details including names of people who helped you and the exact nature of help you received.

Your Answer

2. Did you give any help whatsoever to anyone in solving this assignment? If so, include full details including names of people you helped and the exact nature of help you offered.

Your Answer

3. Did you find or come across code that implements any part of this assignment? If so, include full details including the source of the code and how you used it in the assignment.

Your Answer