# Homework 3: Linear Programs, minimax and network flows

Due date: Friday February 18, 2022

1. **[2 points] Polyhedron modeling.** We saw that the set of $x$ such that $Ax \leq b$ where $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$ describes a polyhedron. For each polyhedron below, find a matrix $A$ and vector $b$ such that $Ax \leq b$ describes the polyhedron. **Hint:** since each inequality describes a different face, $m$ should be equal to the number of faces. Make sure the inequalities go the right way!
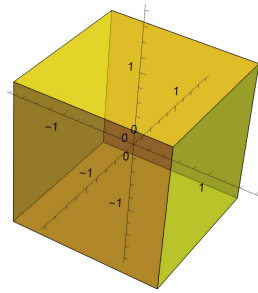


(a) Regular cube centered at the origin with vertices at $(\pm 1, \pm 1, \pm 1)$.
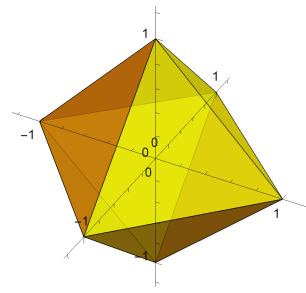
(b) Regular octahedron centered at the origin with vertices at $(\pm 1, 0, 0)$, $(0, \pm 1, 0)$, $(0, 0, \pm 1)$.

2. **[3 points] Museum site planning.** A site is being investigated as a potential location for a new museum. An aerial plan of the site is shown in the figure below (in units of feet). The museum will have a circular footprint and law mandates that there be at least 50 feet of clearance between the building and any edge of the site. If we want the largest possible museum, where should it be located, and what is its optimal radius?

   a) [1 point] Write down the mathematical model of this optimization problem, including the decision variables, constraints and objective function. Make sure to describe all problem parameters.

   b) [1.5 points] Implement the problem in Julia, and determine the optimal location and optimal radius.

   c) [0.5 points] Re-plot the figure below along with the optimally designed museum.
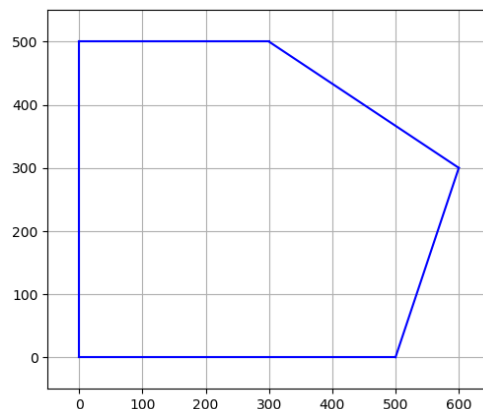
3. **[3 points] Doodle scheduling.** Doodle Inc. is interviewing a candidate for a software engineer position at their company. It works like this: the interview (10 AM to 3 PM) is divided into a number of 20-minute time slots that may be used for 1-on-1 meetings with the candidate. The first 20-minute slot at 10 am should include two employees, one of which needs to be either Mirjam or Matt. There is also a one-hour time slot in the middle of the day where 3 employees take the candidate out for lunch.

The goal is that all the senior employees to meet with the candidate at some point during the day, and that the candidate has someone to meet in every time slot (but never meets anyone more than once). However, everybody has a busy schedule so it's not clear whether this will be possible. A doodle poll (obviously) was sent to the senior employees to figure out their availability. Here is the data:

| | 10:00 | 10:20 | 10:40 | 11:00 | 11:20 | 11:40 | Lunch | 1:00 | 1:20 | 1:40 | 2:00 | 2:20 | 2:40 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mirjam | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| Matt | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| Manuel | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Luca | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| Jule | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| Michael | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| Malte | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| Chris | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| Spyros | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Florian | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Josep | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| Joel | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| Tom | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| Daniel | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Christian | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| Anne | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

In the table, a 1 means that the employee is available at the indicated time, while a 0 means that they are unavailable.

As Doodle Inc's hiring manager, it is your task to determine whether a feasible interview schedule exists.

a) [1 point] Write down the mathematical optimization model for this problem, including a desription of the decision variables, constraints and parameters of the problem.

b) [1.5 points] Implement the problem in Julia and determine whether there exists a feasible solution. See hw3data.ipynb for the data.

c) [0.5 points] If the problem is feasible, print out a calendar for the candidate that lists who they will be meeting at each time slot.