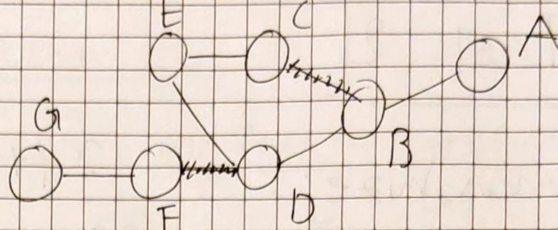
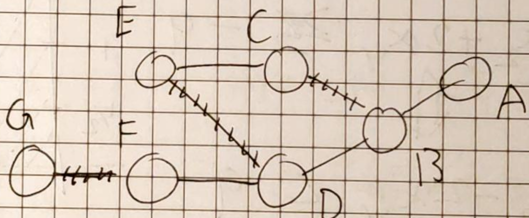
# Task 1




a) M is neither maximal nor maximum, since A is not covered (implying it is not maximal, but maximum $\Rightarrow$ maximal, so it cannot be maximum either).

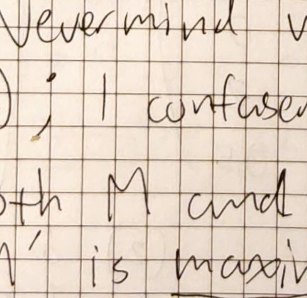
b) G — F — D — E is an alternating path starting and ending in nodes with no outward edges that are matched, thus this path is augmenting. Then:




$|M'| = |M| + 1$

c) A is still excluded from the matching, so neither.

d) Nevermind what I wrote in a) and c); I confused maximal for perfect. both M and M' are maximal, and M' is maximum, since $|M'| = 3 = \lfloor \frac{7}{2} \rfloor$.




Chop off A to achieve a bipartite graph. From Hall's Marriage Thm, this G' has a perfect matching.

$\boxed{\text{Task 2}}$

maximize:

$1.5x_1 + 2x_2 + 2x_3 + x_4$

$x_1 \leq 2x_2 + 1 \quad (1)$

$x_1 + x_3 \leq 7 \quad (2)$

$2x_4 + x_3 \geq x_2 - 9 \quad (3)$

$x_2 + 0.5x_4 \leq 13 \quad (4)$

---

a) let $\vec{x} = (x_1\ x_2\ x_3\ x_4)^T$, $\vec{c} = (1.5\ 2\ 2\ 1)^T$,

~~A = 1 0 0 0~~ rewrite into std. form:

(1) $x_1 - 2x_2 \leq 1$

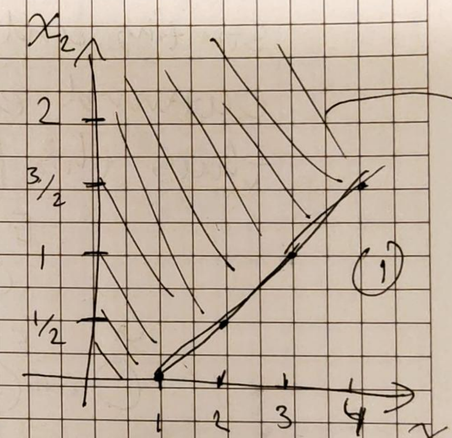
(2) $x_1 + x_3 \leq 7$

(3) $x_3 - x_2 + 2x_4 \geq -9$

$\Updownarrow$

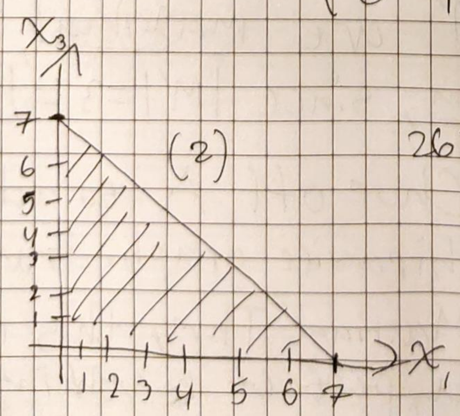
$x_2 - x_3 - 2x_4 \leq 9$

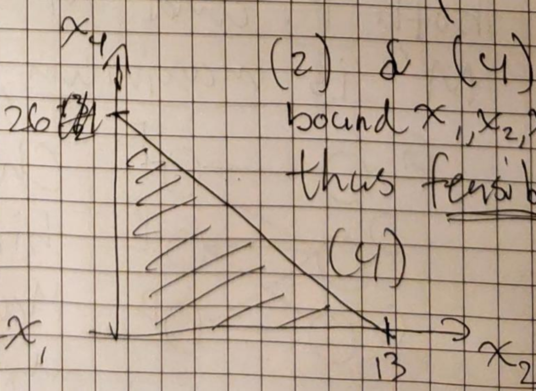
(4) $x_2 + 0.5x_4 \leq 13$




So let $A = \begin{pmatrix} 1 & -2 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & -1 & -2 \\ 0 & 1 & 0 & 0.5 \end{pmatrix}$, $\vec{b} = \begin{pmatrix} 1 \\ 7 \\ 9 \\ 13 \end{pmatrix}$

b)




(2) & (4) bound $x_1, x_2, x_3, x_4$ thus <u>feasible</u>

c) The dual $\overset{D}{\vee}$ of the primal P given by max $c^T x$ such that $Ax \leq b$ is:

minimize $b^T y$ such that $A^T y \geq c$

in other words:

minimize objective function

$y_1 + 7y_2 + 9y_3 + 13y_4$ such that

$$y_1 + y_2 \geq 1.5$$
$$-2y_1 + y_3 + y_4 \geq 2$$
$$y_2 - y_3 \geq 2$$
$$-2y_3 + 0.5y_4 \geq 1$$

$(y \geq 0)$

d) From the strong duality thm since P is bounded, D must also be bounded.

e) The maximum obtained by solving P should by the strong duality thm be the same as the minimum obtained by solving D.

**Task 3** given the following weight matrix

$$\begin{array}{c c} \begin{pmatrix} 20 & 28 & 19 & 13 \\ 15 & 30 & 31 & 28 \\ 40 & 21 & 20 & 17 \\ 21 & 28 & 26 & 12 \end{pmatrix} & \begin{array}{c} 13 \\ 15 \\ 17 \\ 12 \end{array} \end{array}$$

Find row-wise min.

$\downarrow$

Subtract from rows

$$\begin{array}{c} \begin{pmatrix} 7 & 15 & 6 & 0 \\ 0 & 15 & 16 & 13 \\ 23 & 4 & 3 & 0 \\ 9 & 16 & 14 & 0 \end{pmatrix} \\ \begin{array}{cccc} 0 & 4 & 3 & 0 \end{array} \end{array}$$

find column-wise min.

$\downarrow$

Subtract from columns

$$\begin{array}{c} \begin{pmatrix} 7 & 11 & 3 & 0 \\ 0 & 11 & 13 & 13 \\ 23 & 0 & 0 & 0 \\ 9 & 12 & 11 & 0 \end{pmatrix} \\ \begin{array}{cccc} 7 & 11 & 3 & 0 \end{array} \end{array}$$

Still no matching, so find column-wise min., then subtract from all unassigned vertices and add to the intersections.

$\downarrow$

$$\begin{pmatrix} 4 & 8 & \underline{0} & 0 \\ \underline{0} & 8 & 10 & 16 \\ 20 & \underline{0} & 0 & 3 \\ 6 & 9 & 8 & \underline{0} \end{pmatrix}$$

Find matching

$\downarrow$

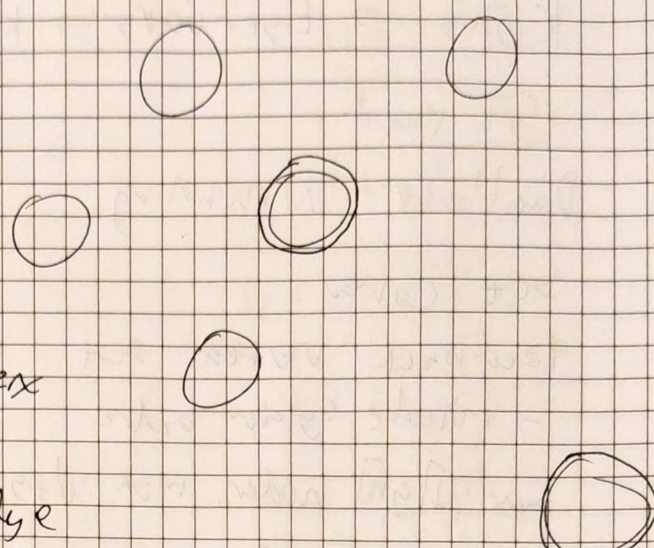
Calculate cost of matching

$$19 + 15 + 21 + 12$$
$$\parallel$$
$$67$$

cost

Task 4

Example
$n = 6$ villages
$K = 2$ clinics



Each ~~node~~ vertex
has ~~an~~ ~~edg~~ a
bidirectional edge
to every other vertex
describing the distance, such that

$$\deg v = n-1 \quad \forall v \in V$$

As I drew these nodes, I ended up guessing that I should place a clinic at the far node, and at the central node. This should be a decent compromise and could be implemented by averaging all the edges of each node, such that

$$\deg v' = 1 \quad \forall v' \in V',$$

then sorting and spreading the $K$ clinics out among the sorted vertices, starting at the minimum average, ending at the maximum. This collapses the problem into a $O(n \log n)$ scheme (due to sorting).

This scheme is heavily flawed.

$\boxed{\text{Task 4}}$ Let $v_1, \dots, v_n$ denote the villages. then, ~~$d(v_i$~~

$$d(v_i, v_j) \leq d(v_i, v') + d(v', v_j),$$

where $d$ is the standard euclidean distance metric, $v_i \neq v_j$ and $v'$ is some other village.

Let $c_1, \dots, c_k$ denote the health clinics.

Our objective is to minimize

$$z_i = \max\{d(v_i, c_j)\}$$

for all $1, \dots, i, \dots, n$.

~~This reminds me of the hungarian algorithm, where you must find the best "compromise" for most.~~

Let $d_{ij}$ denote $d(v_i, c_j)$, then

| | $v_1$ | $v_2$ | $v_3$ | $v_4$ | $\dots$ | $v_n$ |
|---|---|---|---|---|---|---|
| $c_1$ | $d_{11}$ | $d_{21}$ | $d_{31}$ | $d_{41}$ | | $d_{n1}$ |
| $c_2$ | $d_{12}$ | $d_{22}$ | $d_{32}$ | $d_{42}$ | | $d_{n2}$ |
| $\vdots$ | | | | | | |
| $c_k$ | $d_{1k}$ | $d_{2k}$ | $d_{3k}$ | $d_{4k}$ | | $d_{nk}$ |

~~Actua~~