Homework 8

Solution Key

Problem 1

variables
e_i = number of experienced workers at month i
l_i = number of labor hours needed at month i
u_i = number of unexperienced workers at month i

constraints
non-negativity: we cannot have negative amounts of workers or labor hours.

$$e_i, l_i, u_i \geq 0$$

labor done each month: 50 labor units for each experienced worker that is also training an unexperienced worker, of which you need one per each unexperienced worker, and 150 labor units for all the other experienced workers.

$$l_i = 50u_i + 150(e_i - u_i)$$

number of experienced workers working each month: the number of experienced workers from the previous month minus the 10% who left, plus the unexperienced workers from the previous month who are now experienced.

$$e_i = 0.9e_{i-1} + u_{i-1}$$

objective function
We want to minimize the total cost, which is the cost of paying the experienced workers and the unexperienced workers each month.

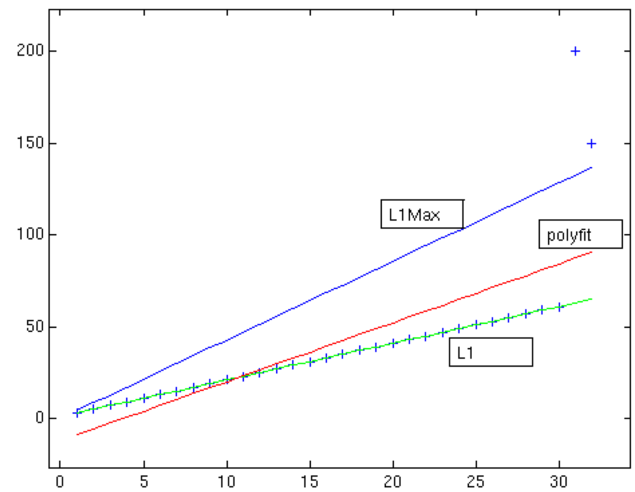
$$\min 3400 \sum_i e_i + 1800 \sum_i u_i$$

Problem 2

1. variables p, q, a_i
   The constraints $a_i \geq px_i + q - y_i$ and $a_i \geq -(px_i + q - y_i)$ together imply that $a_i \geq |px_i + q - y_i|$. The objective function is: $\min \sum_i a_i$. Since we are minimizing, each a_i will end up being equal to $|px_i + q - y_i|$, since there are no other constraints on each a_i.

2. variables p, q, a_i, m
   The same constraints as above, $a_i \geq px_i + q - y_i$ and $ai \geq -(px_i + q - y_i)$ for each i; also the new constraints that for each i, $m \geq a_i$, which will make m at least as large as the max value of a_i. The objective function is $\min m$, thus giving us the min of the max of a_i.

3.



Problem 3

1. **variables**
   c_i: whether or not a course is being taken

   constraints
   $0 \leq c_i \leq 1$ and c_i is an integer (meaning that c_i is either 0 or 1).

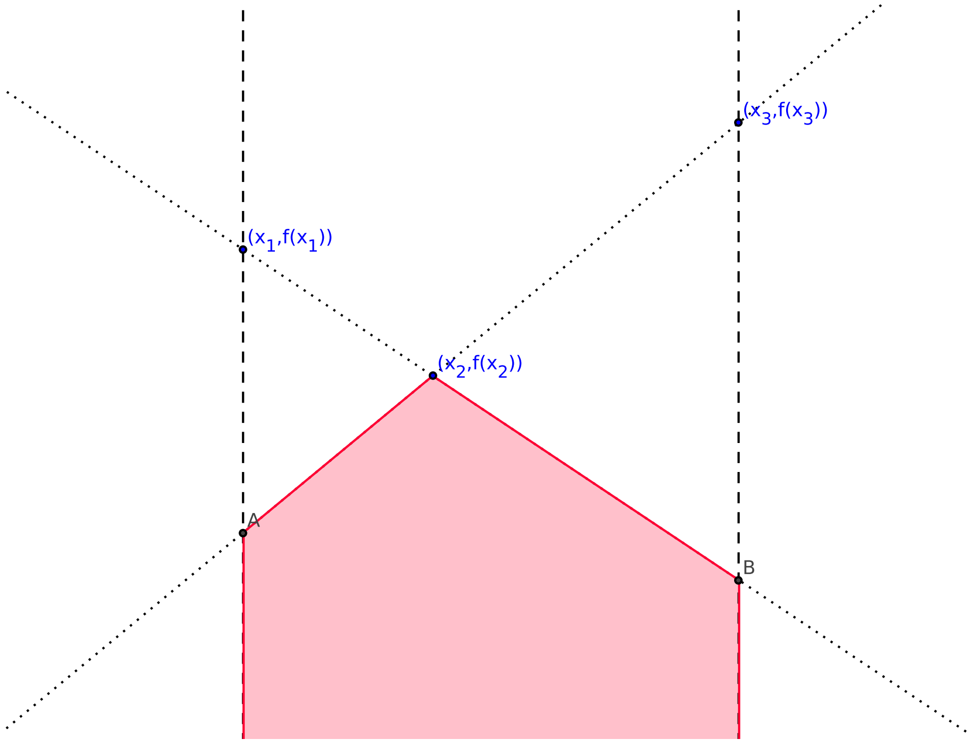
   If course j is a prerequisite for course i, then we add the constraint $c_j \geq c_i$, meaning that if $c_i = 1$, then we must have $c_j = 1$ as well.

   objective function
   $\max \sum_i c_i r_i$

2. The linear program is the optimization problem above, but without the constraint that the c_i variables are integers.

   The second hint reminds us that the set of optimal solutions of a linear program always includes a vertex of the constraint polytope. Recall that a vertex is defined by a subset of the constraints being at equality. Note that all the constraints, at equality, have one of the three forms 1) $c_i = 0$, 2) $c_i = 1$, or 3) $c_i = c_j$. Thus if several such equations have a unique solution, that solution must clearly have each coordinate equal to 0 or 1. Namely, the linear program always has an integer solution.

   (We can find such a solution by iterating through each i, seeing which one of the constraints $c_i = 0$ or $c_i = 1$ does not change the optimal objective value of the linear program, and then adding it to the linear program. Once we have gone through all i, the constraints will now specify a unique solution, which is integral, and which is optimal for the original problem.)



Problem 4

1. Consider the diagram above, illustrating the values of $(x_1, f(x_1))$, $(x_2, f(x_2))$ at some point in the golden ratio search algorithm.

   Our first claim is that there does not exist a convex function which passes through $(x_1, f(x_1))$, $(x_2, f(x_2))$, $(x_3, f(x_3))$ and any fourth point in the red region. Having a point in the red region whose x-coordinate is between x_1 and x_2 will cause the point $(x_2, f(x_2))$ of the function f to be located above the line segment connecting the point $(x_3, f(x_3))$ to the particular point. Also, having a point in the red region whose x-coordinate is between x_2 and x_3 will cause the point $(x_2, f(x_2))$ of the function f to be located above the line segment connecting the point $(x_1, f(x_1))$ and the particular point. Therefore, any other points of the function f must not be in the red region.

   Our next claim is that there exists a convex function which passes through $(x_1, f(x_1))$, $(x_2, f(x_2))$, $(x_3, f(x_3))$ and any fourth point outside the red region whose x-coordinate is between x_1 and x_3. However, we do not have to prove this claim for all possible points; we only need to consider the case where the fourth point is arbitrarily close to point A or point B, whichever is lower. It is obvious that such a convex function exists (one simple example is a piecewise function connecting the dots). So we only need to find value of the function at point A and B, and whichever is lower will be the lower bound.

   Let y_A and y_B be the function values at point A and B respectively. y_A and y_B can be computed using the following equation.

   $$y_A = f(x_2) - \frac{1}{\phi}(f(x_3) - f(x_2)) \qquad \text{and} \qquad y_B = f(x_2) - \phi(f(x_1) - f(x_2))$$

   Note that the answer above can be simplified into other formats, but they would be essentially the same answer.

The lower bound of the convex function f would be

$$\min\left\{f(x_2) - \frac{1}{\phi}(f(x_3) - f(x_2)), f(x_2) - \phi(f(x_1) - f(x_2))\right\}$$

2. The previous part would suggest the following stopping condition.

$$f(x_2) - \min\left\{f(x_2) - \frac{1}{\phi}(f(x_3) - f(x_2)), f(x_2) - \phi(f(x_1) - f(x_2))\right\} \leq \varepsilon$$

which can be rewritten as

$$\max\left\{\frac{1}{\phi}(f(x_3) - f(x_2)), \phi(f(x_1) - f(x_2))\right\} \leq \varepsilon$$

3. Let x_1, x_2 and x_3 be arbitrary x-coordinates such that $x_1 < x_2 < x_3$. Let $\lambda \in [0, 1]$ be such that $x_2 = \lambda x_1 + (1 - \lambda)x_3$ (note that arbitrarily choosing x_2 between x_1 and x_3 is equivalent to arbitrarily choosing λ between 0 and 1).

   Next, let y_1 and y_3 be y-coordinates which attain the minimum value of $f(x_1, y_1)$ and $f(x_3, y_3)$ for fixed x_1 and x_3 respectively. Note that by applying the definition of g, we get $g(x_1) = f(x_1, y_1)$ and $g(x_3) = f(x_3, y_3)$.

   Because f is a convex function, then the following inequality follows from the definition.

$$\lambda f(x_1, y_1) + (1 - \lambda)f(x_3, y_3) \geq f(\lambda(x_1, y_1) + (1 - \lambda)(x_3, y_3))$$

   By substituting each term on the left hand side with $g(x_1)$ and $g(x_3)$ respectively, and by rearranging the terms on the right, we get

$$\lambda g(x_1) + (1 - \lambda)g(x_3) \geq f(\lambda x_1 + (1 - \lambda)x_3, \lambda y_1 + (1 - \lambda)y_3)$$

   From the definition of g, it is always true that $f(x, y) \geq g(x)$ for any x and y; therefore, the right hand side of the inequality above must satisfy the following.

$$f(\lambda x_1 + (1 - \lambda)x_3, \lambda y_1 + (1 - \lambda)y_3) \geq g(\lambda x_1 + (1 - \lambda)x_3)$$

   Combining the last two inequalities above, we get

$$\lambda g(x_1) + (1 - \lambda)g(x_3) \geq g(\lambda x_1 + (1 - \lambda)x_3)$$

   which is true for any arbitrary x_1, x_3 and $0 < \lambda < 1$. Therefore, g must be a convex function.

4. The first time that the algorithm goes in the wrong direction, without loss of generality we assume that $\widetilde{g}(x_2) < \widetilde{g}(x_4)$ while $g(x_2) > g(x_4)$.

   First, we note that $g(x_4) \geq \widetilde{g}(x_2) - \varepsilon$, which follows from combining the inequalities $\widetilde{g}(x_4) > \widetilde{g}(x_2)$ and $g(x_4) \geq \widetilde{g}(x_4) - \varepsilon$.

   We can calculate the least possible true global minimum in this cases by using a similar argument to part 1, which is that

$$\begin{aligned}
\text{global minimum} &= g(x_4) - \phi(g(x_2) - g(x_4)) \\
&= (1 + \phi)g(x_4) - \phi g(x_2) \\
&= \phi^2 g(x_4) - \phi g(x_2) \\
&\geq \phi^2(\widetilde{g}(x_2) - \varepsilon) - \phi(g(x_2)) \\
&\geq \phi^2(\widetilde{g}(x_2) - \varepsilon) - \phi(\widetilde{g}(x_2)) \\
&= (\phi^2 - \phi)\widetilde{g}(x_2) - \phi^2\varepsilon \\
&= \widetilde{g}(x_2) - \phi^2\varepsilon
\end{aligned}$$

Therefore, the global minimum of g is at least $\widetilde{g}(x_2) - \phi^2\varepsilon$

Problem 5

1. (a) *Show that $\{y_i\}$ is a feasible solution to the dual linear program*:
   y has the property that $y_1A_1 + ... + y_mA_m = u$. But this means exactly $A^Ty = \begin{bmatrix} A_1 ... A_m \end{bmatrix} \begin{bmatrix} y_1 \\ ... \\ y_m \end{bmatrix} = u$, which is exactly the constraints we need. Also, by assumption, each y_i is nonnegative.

   (b) *Show that the objective value of $\{y_i\}$ in the dual linear program equals the optimal objective value of the original linear program, $u \cdot x^*$*:
   We have
   $$y_1A_1 + ... + y_mA_m = u$$
   $$y_1A_1x^* + ... + y_mA_mx^* = u \cdot x^*$$
   $$y_1b_1 + ... + y_mb_m = u \cdot x^*$$
   $$b^T \cdot y = u \cdot x^*$$
   where the left hand side of the third equation equals the left hand side of the second equation, term-by-term, because, by assumption, for each constraint i, either $y_i = 0$ or $A_ix^* = b_i$.

   (c) *Summarize what you have found:* Thus we have found a feasible point y for the dual linear program, whose objective b^Ty equals the optimal objective ux^* of the primal, proving the strong duality theorem.

2. (a) Linear program for the column player to choose moves to minimize the row player's expected payoff.
   Variables:
   r, p, s, m.
   r, p, s are the column player's probabilities of playing rock, paper, and scissors, respectively. m is the maximum of the rows of $A \cdot \begin{bmatrix} r \\ p \\ s \end{bmatrix}$, which corresponds to the row player's maximum expected payoff given the strategy defined by r, p, s.
   Constraints:
   $r + p + s = 1$, which states that the total probability of all 3 choices must equal 1.
   $r, p, s \geq 0$, because probabilities must be positive.
   $\begin{bmatrix} m \\ m \\ m \end{bmatrix} \geq A \begin{bmatrix} r \\ p \\ s \end{bmatrix}$, which together with the objective function will enforce the above definition of m.
   Objective Function: $\min m$
   We can summarize all these constraints with the following expression:
   $$\begin{bmatrix} -A_{1,1} & -A_{1,2} & -A_{1,3} & 1 \\ -A_{2,1} & -A_{2,2} & -A_{2,3} & 1 \\ -A_{3,1} & -A_{3,2} & -A_{3,3} & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} r \\ p \\ s \\ m \end{bmatrix} \begin{matrix} \geq \\ \geq \\ \geq \\ = \end{matrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

(b) This is the dual of the above problem:
Variables: r', p', s', m'.
Objective function: $\max m'$
Constraints:

$$\begin{bmatrix} -A_{1,1} & -A_{2,1} & -A_{3,1} & 1 \\ -A_{1,2} & -A_{2,2} & -A_{3,2} & 1 \\ -A_{1,3} & -A_{2,3} & -A_{3,3} & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} r' \\ p' \\ s' \\ m' \end{bmatrix} \begin{matrix} \leq \\ \leq \\ \leq \\ = \end{matrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

And $r', p', s' \geq 0$, with m' unconstrained.

Recall that the dual of an unconstrained variable is an equality constraint, which is why the last variable in each problem is unconstrained, and the last constraint in each problem is an equality constraint.

Clearly this linear program is identical to the one of the previous part, except with the roles of the players swapped—the payoff matrix A is transposed, and we are maximizing instead of minimizing payoffs.

(c) Let f be the optimal objective of both the primal and dual linear program. (By strong duality, as proven in the previous part, these objectives are equal.) Assume both players solve the corresponding linear programs from parts a and b, and play accordingly. Consider the row player. Her expected payoff when both players play like this must be at least the optimal objective value of the linear program from part b, because of the way that program was set up; this equals f. However, from the linear program of part a, she cannot achieve expected payoff greater than the optimal objective value from the linear program in part a, no matter how she plays, given the column player's strategy; this expected payoff is also f, by strong duality. Thus the row player gets expected payoff exactly f and cannot improve this no matter how she plays.

By symmetry, the corresponding argument also holds for the column player.