CS170 Discussion Section 8 Solutions: *3/15*

Change making

You are given an unlimited supply of coins of denominations $v_1, \ldots, v_n \in N$ and a value $W \in N$. Your goal is to make change for W using the minimum number of coins, that is, find a smallest set of coins whose total value is W.

1. Design a dynamic programming algorithm for solving the change making problem. What is its running time?

   For $0 \leq w \leq W$, define

   $$f(w) = \text{the minimum number of coins needed to make a change for } w.$$

   It satisfies

   $$f(w) = \min \begin{cases} 0 & \text{if } w = 0, \\ 1 + \min_{j:\, v_j \leq w} f(w - v_j) \\ \infty \end{cases}$$

   The answer is $f(W)$, where ∞ means impossible. It takes $O(nW)$ time.

2. You now have the additional constraint that there is only one coin per denomination. Does your previous algorithm still work? If not, design a new one.

   For $0 \leq i \leq n$ and $0 \leq w \leq W$, define

   $$\begin{aligned} f(i, w) = {} & \text{the minimum number of coins (among the first } i \text{ coins) needed} \\ & \text{to make a change for } w \text{, having one coin per denomination.} \end{aligned}$$

   It satisfies

   $$f(i, w) = \min \begin{cases} 0 & \text{if } i = 0 \text{ and } w = 0, \\ f(i-1, w) & \text{if } i > 0, \\ 1 + f(i-1, w - v_i) & \text{if } i > 0 \text{ and } v_i \leq w, \\ \infty \end{cases}$$

   The answer to the problem is $f(n, W)$. It takes $O(nW)$ time.

Chocolate Factory

You have a chocolate factory that makes dark and milk chocolate. You profit \$3 per gallons of dark chocolate and \$5 per gallons of milk chocolate. You wish to maximize your profit, but you do have some constraints. You cannot make negative amounts of anything, and you can make at most 400 gallons of chocolate combined.

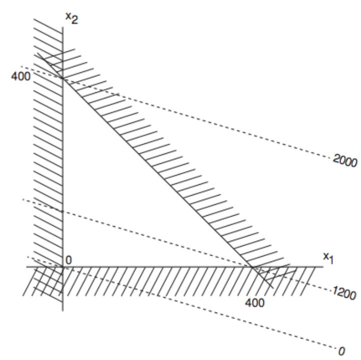
1. Give a constrained optimization formulation of the problem.

   Solution: Let x_1, x_2 respectively be the gallons of dark and milk chocolate.

$$\begin{aligned} &\max\ 3x_1 + 5x_2 \\ &\text{s.t. } x_1 \geq 0 \\ &\quad\ \ x_2 \geq 0 \\ &\quad\ \ x_1 + x_2 \leq 400 \end{aligned}$$

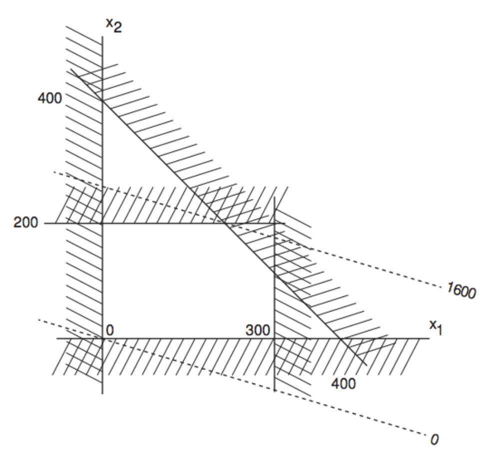
2. Draw the feasible region. Identify the optimal solution and draw the contour lines of the objective to demonstrate the optimality of the solution.

   Solution: The contours of the objective are $\mathcal{C}_c = \{(x_1, x_2) | 3x_1 + 5x_2 = c\}$. The optimal solution is $(0, 400)$.



3. Solve again with the additional constraint that you can't make more than 300 gallons of dark and 200 gallons of milk chocolate.

   Solution: $(200, 200)$.



Shipping

You have an unbounded number of products in different types P that you want to ship around the world. Each product of a given type $p \in P$ has some weight w_p and value v_p. You can choose to ship a product either by plane or by boat: shipping by plane costs \$10/kg, while shipping by boat costs a flat rate of \$1/unit. Shipping by plane is 20 times faster than shipping by boat. Construct a linear programming problem to maximize the total value of products you can ship per unit time while staying under your total budget B. Note that you can only ship an integral number of units of each product.

Solution: For each product type $p \in P$, introduce variables x_p and y_p representing number of units of that product we will ship by plane and boat respectively. First off we must ensure that we don't ship a negative amount of product, so for each $p \in P$ we add the constraints

$$x_p \geq 0 \text{ and } y_p \geq 0.$$

To ensure we don't go over budget, we impose the constraint

$$\sum_{p \in P} 10 w_p x_p + y_p \leq B.$$

To maximize total value per unit time, we maximize

$$\sum_{p \in P} v_p (20 x_p + y_p).$$

Midterm Prep

Best of luck next Monday! The staff will be holding a review session Saturday, March 18th at 2 PM in 100 GPBB. The worksheet will be posted on Piazza afterwards. If there is extra time at the end of discussion, you could ask the TA to review certain topics for the midterm.

Road Trip (Optional)

You are on a road trip, and there are n cities along a road, labeled 1 to n. Conveniently, these cities all lie on a single road, and the distance between two adjacent cities is one mile. We are currently at city 1, and would like to drive to city n. Each day, we can drive at most k miles, before we sleep for the night. We pay a_i for lodging at the city located at mile i. Each lodging cost is a positive number. Given that we can spend an arbitrary number of days on the road trip, determine a plan of driving to minimize lodging costs.

Input: An integer k, and a length n array of positive integers $a_1, \ldots, a_n$.
Output: The minimum lodging cost to complete the road trip starting from city 1 and ending at city n.

The running time of your algorithm should be $O(nk)$.

Solution:
Let $C(i)$ denote the minimum cost we need to spend on lodging, if we start at city 1 and ending, and sleeping, at city i.
So, the recurrence is

$$C(i) = a_i + \min_{i-k \leq j < i} C(j).$$

With base cases $C(0) = 0$, and $C(i) = \infty$ for all $i < 0$ (i.e. it's impossible to start from before mile zero). Our final answer is just $C(n)$ (or $\min_{n-k+1 \leq j \leq n} C(j)$, if we don't need to sleep at city n).