Written #5: Greed, LP, DP
Due: see Canvas

This is your last written homework. We are reading a last few topics, in Chapter 6 and also outside the book. Summer is coming! Write up solutions, and submit them on Canvas in `*.pdf` format. Also submit one `*.lp` file for the LP problem.

Problem 1. Flow Check. Suppose we have a flow network (G, s, t) with V vertices and E edges. Someone proposes a flow f (that is, a value $f(e)$ for each edge e of G). Show how to verify that f is a maxflow in $O(V + E)$ time. Be sure to check all the constraints.

Problem 2. Dual LP's. See the *bipartite matching* example on page 62 of slides/64MaxFlow.pdf, with ten vertices and twelve edges. File writ5/bm-max.lp formulates this as a "MAX" linear program, with a variable per edge and a constraint per vertex. (To review the "Brewer" example, see writ5/brew-max.lp and writ5/brew-min.lp).

2(a). Write out the dual "MIN" linear program, and submit it as file named `bm-min.lp`. You should reuse the variable and constraint names from the MAX lp, to make the duality explicit. Use the solver in writ5/ to verify that both LP's have (approximately) the same value.

2(b). For any bipartite graph, we could again write out these "MAX" and "MIN" LP's. (In fact both LP's attain their opitmum value at integer (0/1) values, which is not true of all LP's.) We interpret the "MAX" LP as selecting as many edges as possible, so that no two edges touch at a vertex. In other words, it finds a *matching* of maximum size. Give a similar interpretation for the "MIN" LP. Also, whose theorem states that these two things have the same size?

Problem 3. Greedy VC. Given a graph G, a *vertex cover* is a subset C of vertices, touching all the edges of G. We would like to find a vertex cover of minimum size. However, the exact minimum is "NP-hard" to find, so we consider this fast greedy approach instead:

GreedyVC(G):
    $C \leftarrow \emptyset$
    while G has edges:
        pick a vertex v in G with maximum degree
        add v to C
        remove all edges touching v from G
    return C

Trace out an example, showing that GreedyVC(G) can return a cover which is at least twice the minimum size. (In your example, when there is a tie, you may assume that the tie breaks in your favor.)

Problem 4. Numbers Game. We are given a sequence of positive integers $v_1, v_2, \ldots, v_n$. Players A and B will take turns "claiming" these integers, until none are left (A goes first). On each turn, a player may choose either the leftmost or the rightmost unclaimed integer. Each player wants to maximize their *income*: the sum of their claimed integers.

4(a). In a "greedy move", a player simply picks the larger available integer. Show an example where the greedy move is not optimal as A's first move.

4(b). Assuming both players play optimally, show how to predict A's total income, and how to choose A's best first move, in $O(n^2)$ time.

Problem 5. EMST. Suppose S is a finite set of points in the plane. Think of the points as vertices. Between every pair of points, add an edge whose weight equals the Euclidean distance between its two endpoints. The Euclidean MST (EMST) of S is the minimum weight spanning tree using these edges. Suppose e is an edge of the EMST, connecting points $p, q \in S$. Draw the circle with diameter e. Argue that no point of S is in the interior of the circle.