Homework Assignment #5
Due: March 2, 2018 at 2:30 p.m.

1. Hanmattan is the capital of Leutonia. It is situated on an island in the shape of a regular hexagon. The roads on the island all run north-south or east-west. (See figure.) At the centre of the island is the palace of the Grand Poohbah of Leutonia.

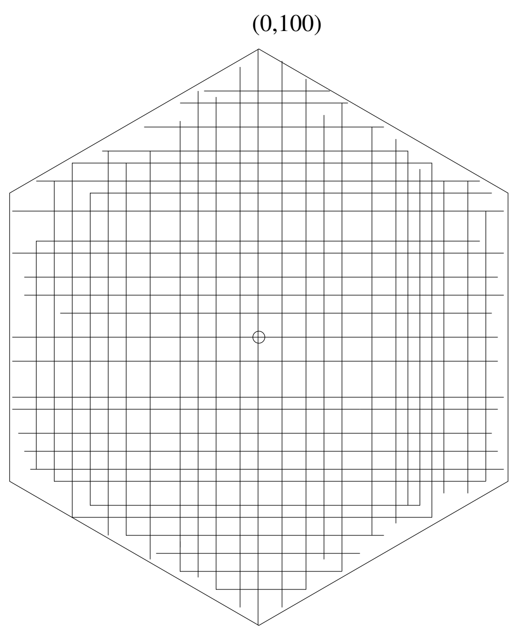
Real-estate listings for houses in Hanmattan are always listed in increasing order according to the distance from the Grand Poohbah's palace, since houses closer to the center of power are considered more desirable. Distance is measured along the shortest route along the streets.

Each week, a real-estate agent gets a list of n houses to be sorted. The Cartesian coordinates of each house are known. The Grand Poohbah's palace is at coordinates (0,0). The road network is set up so that the distance to a house at location (x, y) is $|x| + |y|$. One corner of the hexagonal island is at coordinate (0,100).

Assume that the locations of the houses are independent and uniformly randomly distributed over the island's territory. Thus, the probability of a house being in any region of the island is proportional to the area of that region. Assume that arithmetic operations on the coordinates can be done in O(1) time.

Give a *simple* algorithm to sort the n locations in expected $O(n)$ time. Explain why your answer is correct.

Hint: First try solving the problem for a diamond-shaped island (i.e., a square island whose shores run from southwest to northeast and from southeast to northwest). Then notice that essentially the same algorithm runs in expected $O(n)$ time for the hexagonal island (and *explain why* that is true).



2. Recall the canoe rental problem from class. A traveller wishes to travel down a river in rented canoes. There are n canoe rental agencies located along the river. We number them 1 to n in the order the appear going down the river. It costs p_{ij} dollars to rent a canoe at agency i and return it at agency j, where $j > i$. (You cannot paddle upstream; the current is too strong.)

In the original problem, you want to find the optimal cost of travelling from agency 1 to agency n. We saw in class that you can solve this problem by defining $A[i]$ to be the optimal cost of travelling from agency i to agency n (for $1 \le i \le n$). Then, the optimal cost $A[1]$ can be computed using the following recurrence.

$$\begin{aligned} A[n] &= 0 \\ A[i] &= \min_{j>i}(p_{ij} + A[j]), \text{for } 1 \le i \le n \end{aligned}$$

Now, suppose you want to solve this problem with an additional constraint that you want to rent a maximum of k different canoes to make the trip. (It takes a long time to stop at each rental agency, return one canoe, borrow the next one and transfer your gear into the new canoe, so you don't want to do this too many times.)

Let $B[i, \ell]$ be the optimal cost of travelling from agency i to agency n using at most ℓ different canoes (for $1 \le i \le n$ and $1 \le \ell \le k$).

(a) Give a recurrence that can be used to compute the entries of B.

(b) Provide pseudocode for computing $B[1, k]$ efficiently. In your pseudocode, you can store additional information that may be needed to solve part (d), below.

(c) What is the running time of the pseudocode in part (b)? Use Θ notation to state your answer in terms of n and/or k. Assume that all costs and indices can be stored in a single word of memory.

(d) After the computation in (b) has been done, you might want to print out the optimal itinerary that achieves the cost $B[1, k]$. Give pseudocode for printing this optimal itinerary.