CS200 - Problem Set 7

Due: Monday, Nov.6 to submission server before class

Please read the sections of the syllabus on problem sets and honor code before starting this homework.

1. Graph representations:

   (a) **[9 points total]** (See grading rubric for pseudocode grading scheme.) Fill in the pseudocode for the following algorithms. You can assume that the graph is not directed, and all edges have weight 1.

   Algorithm 1: `DegAdMat`(A, i)

   Input : Adjacency Matrix A of a graph with n vertices. An integer $i : 1 \leq i \leq n$.
   Output: Degree of the ith vertex.

   Algorithm 2: `DegAdList`(A, i)

   Input : Adjacency List A of a graph with n vertices. An integer $i : 1 \leq i \leq n$.
   Output: Degree of the ith vertex.

   Algorithm 3: `EdgeAdMat`(A, i, j)

   Input : Adjacency Matrix A of a graph with n vertices. Integers $i, j : 1 \leq i, j \leq n$.
   Output: 1 if edge between i and j, zero otherwise.

   Algorithm 4: `EdgeAdList`(A, i, j)

   Input : Adjacency List A of a graph with n vertices. Integers $i, j : 1 \leq i, j \leq n$.
   Output: 1 if edge between i and j, zero otherwise.

   (b) **[6 points]** What is the worst case time complexity for each of your algorithms? Explain.

2. **[6 points]** *Handshaking Lemma:* Suppose you have a graph $G(V, E)$ that has no self-loops. Let $deg : V \to \mathbb{N}$ be the degree function, so $deg(v)$ is the degree of vertex v. Then

$$\sum_{v \in V} deg(v) = 2|E|. \tag{1}$$

Please explain why.

3. **[11 points]** Use the Handshaking Lemma to prove that the number of vertices in a graph with odd degree is even.

4. We are currently studying the following algorithm in CS302.

Algorithm 5: `dynamic`(n)

Input: An $n \times 3$ array L containing natural numbers.
$A, B \in \mathbb{N}$, a rectangular array Q of size $A \times B$ with all 0's

```
for k=1 to A do
    for j=1 to B do
        for q=1 to k-1 do
            if Q[k, j] < Q[q, j] + Q[k − q, j] then
                Q[k, j] := Q[q, j] + Q[k − q, j];
            end
        end
        for r=1 to j-1 do
            if Q[k, j] < Q[k, r] + Q[k, j − r] then
                Q[k, j] := Q[q, j] + Q[k − q, j];
            end
        end
        for i=1 to n do
            if (k = L[i, 1] and j = L[i, 2]) or
              (k = L[i, 2] and j = L[i, 1]) then
                if Q[k, j] < L[i, 3] then
                    Q[k, j] := L[i, 3];
                end
            end
        end
    end
end
return Q[A, B];
```

(a) **[6 points]** Use summation notation to analyze the asymptotic runtime of the algorithm.

(b) **[6 points]** Explain how you can bound the asymptotic runtime without using summation notation, but instead using a worst-case analysis.

5. **[6 points each]** For many card games, it is helpful to know the probability of certain hands

occuring. In a standard deck, there are 52 cards. There are 13 different kinds of cards, and each kind appears 4 times, one for each of four suits (hearts, spades, diamonds, clubs).

(a) If you are dealt 5 cards, and all possibile hands of cards are equally likely, what is the probability that you get 4 of the same kind?

(b) If you are dealt 5 cards, and all possibile hands of cards are equally likely, what is the probability that you get a full house, where a full house is 3 cards of one kind, and 2 cards of a different kind?

6. **[6 points]** Suppose you create a graph on n vertices by looking at each pair of vertices, and then choosing to put an edge there with probability 2/3. What is the sample space of this problem? What is the size of the sample space?

7. How long did you spend on this homework?