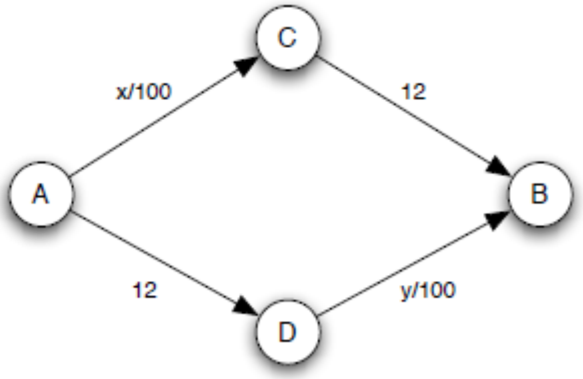
Big Data Algorithms and Analysis Final Exam

Name: No.:

1. Sampling is a simple but effective method in big data, and is adopted to estimate the median of streaming data where we cannot know the amount of data in prior. Now you have a storage that can keep fixed size of samples. Design an efficient sampling method for a stream of numbers such that each number has equal probability to be sampled at any time. Provide your justification.

2. There are 1000 cars which must travel from town A to town B. The directed graph in figure below indicates that travel time for each car on edge AC is x/100 if x cars go through edge AC, and similarly the travel time for each car on edge DB is y/100 if y cars go through edge DB. The travel time for each car on each of edges CB and AD is 12 regardless of the number of cars on these edges. Each driver wants to select a route to minimize his travel time. The drivers make simultaneous choices.



(a) At Nash equilibrium, how many cars go through route ACB?
(b) Now add a new (one-way) road from town C to town D. This new road from C to D has a travel time of 0 for each car regardless of the number of cars going through it. Identify the equilibrium of drivers. If the equilibrium is unique, prove the uniqueness, otherwise, present at least two equilibriums. What happens to total cost-of-travel (the sum of total travel times for the 1000 cars) as a result of the availability of the new road?
(c) Suppose now that the travel times on edges CB and AD are reduced to 5. The road from C to D that was constructed in part (b) is still available. Find a Nash equilibrium on the current network. What are the equilibrium values of x and y? What is the total cost-of-travel? What would happen to the total cost-of-travel if the road from C to D was removed?

3.
(a) In the page replacement algorithm, we learn about the FIFO (First In First Out) and LRU (Least Recently Used) replacement algorithms. Please verify whether the following statement hold, and provide your reasons. Statement: If the memory is larger, the page fault rate is always smaller for each algorithm.
(b) Prove for any deterministic online replacement algorithm, the competitive ratio is no less than

k, where k is the maximum number of pages that can fit into the cache.

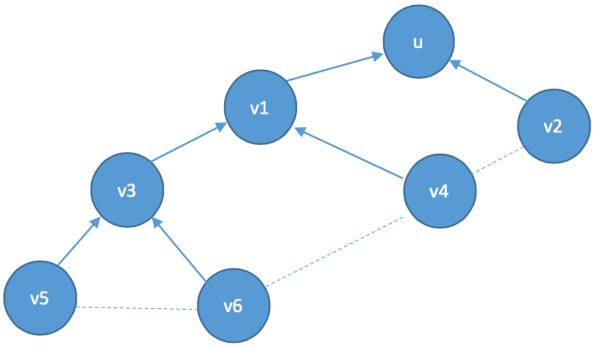
4. A Bloom filter is a probabilistic data structure to test whether an element is a member of a set. An empty bloom filter is a bit array of size m, all set to 0. There must also be k different hash functions to hash some set element to one of the m bit array positions. Bloom filter is space-efficient and has 100% recall rate.

(a) Suppose the false positive error rate a bloom filter is 0.19. What is the new false positive error if we double both the space and the number of hash functions of the bloom filter?

(b) Deletion is not allowed in the bloom filter in that it may delete other members of the set at the same time. Modify the bloom filter such that it can realize the deletion operation.

(c) Discuss the possible disadvantages of your approach?

5. Shortest path plays an indispensable role in large social network analysis. In general, distance in social network means hop-count distance. A tree structure method is proposed using landmarks to dynamically retrieve the shortest path in this setting. The main idea is simple: Firstly, find k landmarks by some criterions. Then, conduct k BFS search operations from each landmark and build k trees independently. For example, we give a structured tree for a landmark u.



The dotted lines refer to lines in actual graph but not selected in trees. Therefore, by triangle inequality, $d(v_1, v_2)$ can be bounded easily using

$$d(v_1, v_2) \le \min_k (d(v_1, u_k) + d(u_k, v_2))$$

where u_k is the k-th landmark, which is the root of k-th tree.

(a) The above method is just the simplest idea and gives the upper bound of shortest path distance. Can you give some improvements to get more accurate distance between nodes using k structured trees? And please give the time complexity (You may use the maximum length of paths in the graph or in the k trees as a parameter).

(b) One of advantages of this method is that it can be used in dynamic networks. Please analyze how it deals with the insertion and deletion of edges.

6. Let X_1, X_2, X_3 be a random sample from a distribution of the continuous type having pdf $f(x) = 2x, x \in [0,1]$.

(a) compute the probability that the smallest of X_1, X_2, X_3 exceeds the median of the distribution.

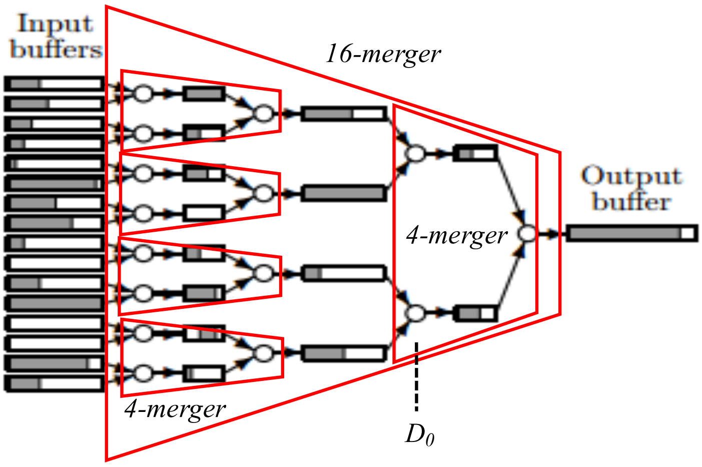
(b) If $Y_1 \le Y_2 \le Y_3$ are the order statistics, find the correlation between Y_2, Y_3.

7. Let $l(s)$ be the length of string s, and $K(s)$ be the Kolmogorov Complexity of string s. A string s is compressible by a number c if it has a description whose length does not exceed $l(s)$-c. And a string s is c-incompressible if $K(s)$>$l(s)$-c. Given a c-incompressible string x of length n, prove that any substring of x is 2log(n)+c+d- incompressible, where d is the length of the program that directly output any string it reads.

8. Just like Funnelsort introduced in class, Lazy Funnelsort is a sorting algorithm that operates on a contiguous array of N, and it performing the following:

1. Split the input into $N^{1/3}$ arrays of size $N^{2/3}$, and sort the arrays recursively.
2. Merge the $N^{1/3}$ sorted sequences using a $N^{1/3} - merger$

The difference is that Lazy Funnelsort uses a simplified version of *k-merger*, which "fold out" the original *k-merger* to a tree of binary mergers. An example of simplified 16-merger is shown in following figure.



The sizes of the buffers in *k-merger* are defined recursively: Let $D_0 = \lceil \log(k)/2 \rceil$ denote the middle level in the tree. Let the top tree be the subtree consisting of all nodes of depth at most D_0, and let the subtrees rooted by nodes at depth $D_0 + 1$ be the bottom trees. The edges between nodes at depth D_0 and $D_0 + 1$ have associated buffers of size $\lceil k^{3/2} \rceil$ and the sizes of the remaining buffers is defined by recursion on the top tree and the bottom trees. During sorting, only when a buffer is empty, the downstream part will be activated to fill the buffer as full as possible.

(a) Prove that the size of a *k-merger* (excluding its input buffers and output buffer) is bounded by ck^2 for a constant $c \ge 1$.

(b) Consider the I/O model shown in following figure. Suppose the size of Memory 1 is M and transfer block size is B. A *k-merger* performs $\frac{ck^3}{B}\log_M(k^3))$ I/Os during an invocation (that is, output k^3 element) and c is a constant. Prove that Lazy Funnelsort uses $\frac{3cN}{B}\log_M N$ I/Os to sort N elements.

9. The key step in Kirkpatrick–Seidel algorithm to find the convex hall is to find the tangent between two sub convex. Although the tangent can be found by linear programming, but the time complexity of solving general linear programming is not linear.

(a) Provide an example that the points with maximum y-coordinate of each sub convex are not the points in the grand convex.

(b) Following is an algorithm of finding tangent, which consists of the following steps:

1. Randomly pair the points of the input point set *P*, |*P*|=*n*, into $\left\lfloor\frac{\mathrm{n}}{2}\right\rfloor$ distinct line segments. Call this set *Q*, and name each of the $\frac{n}{2}$ line segments q_i. We name the endpoint of q_i with no bigger x-coordinate l_i and the other r_i. If n is odd, there may be one point which does not have a partner. This is acceptable. And let *M* be the median of x-coordinate of points in *P*.

2. For each q_i, if it is vertical, delete the endpoint with smaller y-coordinate and remove q_i from *Q*. Determine the median slope *m* of all remaining line segments in Q. For singleton points, consider slope to be zero. If |*Q*| is even, *m* is assumed to be the greater slope between two median segments.

3. Construct a line L with slope *m* across a point $p \in P$ such that none of the other points in *P* is in the upper side of L. Let $p_s, p_t \in P$ be the leftmost and rightmost points on L.

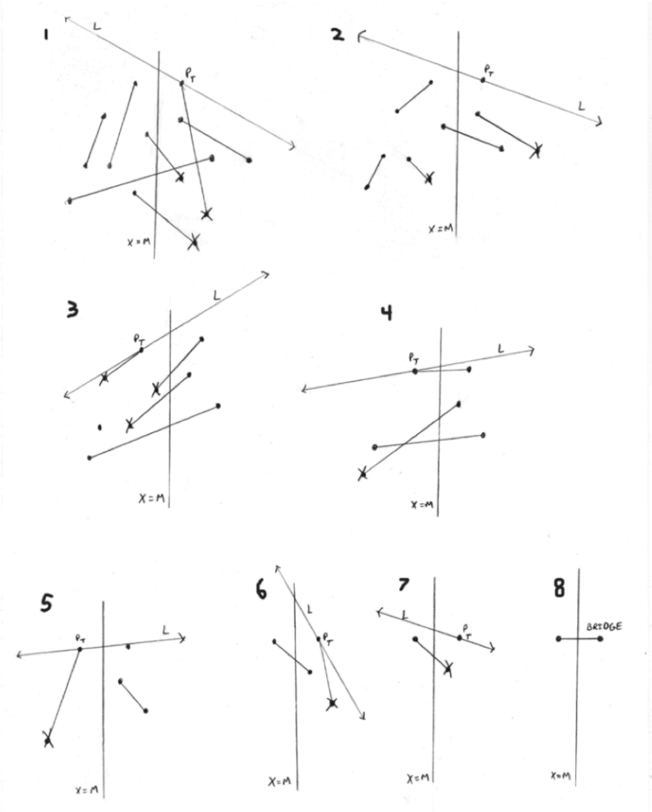
4. If p_s, p_t are already the points in each sub convex we want to connect, then $p_s p_t$ is the target tangent. Otherwise, prune the set of tangent point candidates as follows:

• If x-coordinate of $p_s > M$, for every line segment $q_i \in Q$ whose slope is $\leq m$, remove r_i

• If x-coordinate of $p_t \leq M$, for every line segment $q_i \in Q$ whose slope is $\geq m$, remove l_i

5. Go back to step 1 with the new point set *P*.

Following is an illustration figure.



Prove that the complexity of the algorithm is $O(n)$. Explain the correctness of the algorithm.