Quiz 1

- Do not open this quiz booklet until you are directed to do so. Read all the instructions first.
- The quiz contains 6 problems, several with multiple parts. You have 2 hours for this midterm.
- This quiz booklet contains 17 pages, including this one and three sheets of scratch paper. **Do not remove any of these pages!**
- **Do not write anything on the back of the sheets, we will not scan and therefore not grade it!**
- Write your solutions in the space provided. If you run out of space, continue on a scratch page and make a notation.
- Do not waste time deriving facts that we have studied. Just cite results from class.
- When we ask you to give an algorithm in this quiz, describe your algorithm in English or pseudocode, and provide a short argument for correctness and running time. You do not need to provide a diagram or example unless it helps make your explanation clearer.
- Do not spend too much time on any one problem.
- **Please write your name on every single page of this exam.**
- Good luck!

Problem	Title	Points	Parts	Grade	Initials
0	Name	1	1		
1	True/False	14	7		
2	Ordered Stack	15	2		
3	Find the Pairs	15	1		
4	Min-cut using MST	15	2		
5	Well Spaced Triples	20	2		
6	Bin Packing	20	3		
Total		100			

Name: ____________________

Circle your recitation:

F10	F11	F12	F1	F2	F3	F11	F12	F1
Katerina	Themis	Shankha	Maryam	Nishanth	Prashant	Manolis	Shalom	Ethan
R01	R02	R03	R04	R05	R06	R07	R08	R09

Quiz 1-1: [14 points] T/F Questions

Mark each statement as either true or false. You have to **provide a short explanation for each**.

(a) Consider $f(x) = \frac{x^3}{3}$ and $x^{(0)} \neq 0$. The gradient descent algorithm with the following iteration

$$x^{(t+1)} \leftarrow x^{(t)} - \frac{1}{2} f'(x^{(t)})$$

always converges to $-\infty$.

(b) We are given an array $A[1...n]$. We can delete its $n/2$ largest elements in time $O(n)$.

(c) The following definition is equivalent to that of a universal hash family:
Let $\mathcal{H}$ be a family of hash functions $h : \{0,1\}^u \to \{0,1\}^m$, then for any $h \neq h' \in \mathcal{H}$

$$\mathbb{P}_{k \in \{0,1\}^u}[h(k) = h'(k)] \leq \frac{1}{m}$$

(d) We define n random variables $X_i = \sum_{j=1}^{n} c_{(i,j)}$, where for every (i,j), $c_{(i,j)}$ is determined by an independent unbiased coin flip, so $c_{(i,j)}$ is one with probability 1/2 and zero otherwise. Then, for every $\beta \in (0,1)$

$$\mathbb{P}\left[\sum_{i=1}^{n} X_i > (1+\beta)\frac{n^2}{2}\right] < e^{-\beta^2 n^2/6}$$

(e) Let $G = (V, E)$ be a connected graph and let H be the subgraph of G induced by some set of vertices $V' \subset V$. That is, $H = (V', E')$ where E' consists of all edges both of whose endpoints are in V'. Then every MST of H is a subgraph of some MST of G.

(f) Consider a k-bit vector $v = x_1 \ldots x_k$ chosen uniformly at random from $\{0, 1\}^k$. For any $S \subseteq \{1, \ldots, k\}$ such that $S \neq \emptyset$, define $m_S(v) = \sum_{i \in S} x_i \pmod 2$. Then,

$$\mathrm{Var}\left[\sum_{S \subseteq \{1,\ldots,k\}, S \neq \emptyset} m_S(v)\right] = \frac{2^k - 1}{4}$$

(g) If an algorithm runs in time $\Theta(n)$ with probability 0.9999 and in time $\Theta(n^2)$ with the remaining probability, then its expected run-time is $\Theta(n)$.

Quiz 1-2: [15 points] Ordered Stack

An ordered stack is a data structure that stores a sequence of items and supports the following operations.

- ORDEREDPUSH(x) removes all items smaller than x from the beginning of the sequence and then adds x to the beginning of the sequence.
- POP deletes and returns the first item in the sequence (or returns Null if the sequence is empty).

(a) **[8 points]** Show how to implement an ordered stack with a simple linked list. Prove that if we start with an empty data structure and perform a sequence of n arbitrary ORDEREDPUSH(x) and POP operations, then the amortized cost of each such operation is $O(1)$.

(b) [7 points] Suppose we are given an array $A[1 \ldots n]$ that stores the height of n buildings on a city street, indexed from west to east. The view of building i is defined as the number of buildings between building i and the first building west of i that is taller than i. For example if $A = [1, 5, 2, 3, 1, 2, 1, 8]$, then the view of building 4 (which has height 3) is 1.

You want to compute the view of the n buildings of A. Show how you can modify the ordered stack in order to do that in $O(n)$ time.

Quiz 1-3: [15 points] Find the Pairs

Consider a set U of $2n$ distinct balls numbered from $1 \ldots 2n$ distributed evenly across n containers (each container has exactly 2 balls). You are not allowed to look inside the containers. But, we will answer your queries of the following form:

"*What is the smallest number of containers that contain all the balls in* S?" where $S \subseteq U$ is any subset of the $2n$ balls.

For example if $n = 4$ and the balls were arranged in the following way: $\{1, 4\}, \{2, 8\}, \{3, 6\}, \{5, 7\}$ in 4 containers, an example query would be "*What is the smallest number of containers that contain the balls* $\{1, 4, 2, 6\}$?" to which we would reply 3.

Two balls are said to be paired if they lie in the same container. Your task is to figure out for each ball the other ball it is paired with. Give an algorithm for figuring out all the pairings using $O(n \log n)$ queries. Note that we only care about the total number of queries your algorithm uses and not the running time.

Quiz 1-4: [15 points] Min-cut using MST

Consider the following algorithm that takes as input an *unweighted* graph and outputs a cut of this graph:

KCUT$(G(V, E))$:

Create weighted graph $G'(V, E, w)$, where w is a uniformly random assignment of distinct weights from $\{1, \ldots, |E|\}$ to edges in E.
Run KRUSKAL(G') to get a tree $T = (V, E_T)$.
Let e_f be the last edge that was added to E_T in the above execution of KRUSKAL(G').
$T' = (V, E_T \setminus \{e_f\})$ has exactly two connected components. Call them C and $V \setminus C$.
return the cut $(C, V \setminus C)$.

(a) [10 points] Prove that for any unweighted graph $G(V, E)$, the probability that KCUT(G) returns a minimum cut of G is at least $\Omega\left(\frac{1}{|V|^2}\right)$.

(b) [5 points] We want to find a min-cut with high probablity, namely with probability at least $1 - \frac{1}{n}$. Design an algorithm that uses the KCUT procedure to achieve this, and analyze its running time. Recall that Kruskal's algorithm has running time $O(m \log m)$, where $m = |E|$.

Quiz 1-5: Well Spaced Triples

Suppose we are given a bit string $B[1 \ldots n]$. A triple of distinct indices $1 \leq i < j < k \leq n$ is called a *well-spaced triple* in B if $B[i] = B[j] = B[k] = 1$ and $k - j = j - i$.

(a) **[5 points]** Describe an algorithm to determine whether B has a well-spaced triple in $O(n^2)$ time.

(b) **[15 points]** Describe an algorithm to determine the number of well-spaced triples in B in $O(n \log n)$ time.
(**Hint:** Use FFT)

Quiz 1-6: Bin Packing Let X_i, $1 \leq i \leq n$ be independent and identically distributed random variables following the distribution

$$\mathbb{P}\left(X_i = \frac{1}{2^k}\right) = \frac{1}{2^k} \quad \textit{for} \quad 1 \leq k \leq \infty$$

Each X_i represents the size of the item i. Our task is to pack the n items in the least possible number of bins of size 1. Let Z be the random variable that represents the total number of bins that we have to use if we pack the n items optimally. Consider the following algorithm for performing the packing:

BIN-PACKING$(x_1, ..., x_n)$:

Sort $(x_1, ..., x_n)$ by size in decreasing order. Let $(z_1, ..., z_n)$ be the sorted array of items
Create a bin b_1
For each $i \in \{1, ..., n\}$
  If z_i fits in some of the bins created so far
    Choose one of them arbitrarily and add z_i to it
  Else create a new bin

return the number of created bins.

(a) **[8 points]** Prove that the above packing algorithm is optimal for this problem.

(b) **[7 points]** Prove

$$\mathbb{E}[Z] \leq \frac{n}{3} + 1$$

(c) **[5 points]** Prove that

$$\mathbb{P}\left(Z \geq \frac{n}{3} + \sqrt{n}\right) \leq \frac{1}{e}$$

Scratch page

Scratch page

Scratch page