**Advanced Algorithms**

**Exercise for Lecture 4**

|  |  |  |  |
| --- | --- | --- | --- |
| **Student Name** |  | **Student ID** |  |
| **Problem 1** |  | | |
| **Problem 2** |  | | |
| **Problem 3** |  | | |
| **Problem 4** |  | | |
| **Total Score** |  | | |
| **Notes** | Deadline: **2023-09-22 24:00**  Submission Format: ‘**Lecture4\_Name\_Student ID.docx**’, and please send to: **[1914499454@qq.com](mailto:algorithms_23fall@163.com)**.  This assignment is meant to be an evaluation of your **individual** understanding coming into the course and should be completed **without collaboration** or outside help. | | |

**Problem 1.[25 points]** Consider a version of the division method in which ℎ(k)=k mod m, where m=2p−1 and k is a character string interpreted in radix 2p. Show that if we can derive string x from string y by permuting its characters, then x and y hash to the same value. Give an example of an application in which this property would be undesirable in a hash function.

**Problem 2.[25 points]** Suppose that we use double hashing to resolve collisions—that is, we use the hash function h(k,i)=(h1(k)+ih2(k))mod m. Show that if m and h2(k) have greatest common divisor d≥1 for some key k, then an unsuccessful search for key k examines (1/d)th of the hash table before returning to slot h1(k). Thus, when d=1, so that m and h2(k) are relatively prime, the search may examine the entire hash table.

**Problem 3.[25 points]** Suppose that we use an open-addressed hash table of size m to store n ≤ m/2 items. Assuming uniform hashing, show that for i = 1, 2..n, the probability is at most 2−k that the i-th insertion requires strictly more than k probes.

**Problem 4.[25 points]** Suppose that we have a hash table with n slots, with collisions resolved by chaining, and suppose that n keys are inserted into the table. Each key is equally likely to be hashed to each slot. Let M be the maximum number of keys in any slot after all the keys have been inserted. Your mission is to prove an O(lgn/lglgn) upper bound on E[M], the expected value of M. Argue that the probability Qk that exactly k keys hash to a particular slot is given by

Qk =