CS670: Assignment 3

Instructions

- Using LaTeX to typeset your solutions will fetch +5 bonus marks in the assignment.
- You can ask the instructor for help in the assignment during office hours or via appointment.
- All help will stop 48 hours before the assignment deadline.
- **Deadline: April 17th 2024, EOD**

1 Squareroot ORAM

1. Describe the square root ORAM construction in detail. In particular, describe the role of dummy items and stash (cache). Please make sure your answer is precise. Do not use words that are not necessary and add value to the answer. **20 marks**
2. Consider a memory of size n. Suppose we change the size of the stash (cache) to $n^{1/3}$. What would be the amortized cost to do one read operation. Show your work. **10 marks**

2 Oblivious Datastructures

We want to implement an Oblivious Min Heap. To do that, we will store the heap in an array, called, A, in the most obvious form. The 0^{th} index of the array is kept free. The first index stores the root. The $(2i)^{\text{th}}$ location stores the left child of the node stored in the i^{th} location. The right child is stored in location $2i + 1$. P_0 and P_1 hold A_0 and A_1 (each of size $n + 1$, i.e., it has n elements; remember, the 0^{th} index is kept free) respectively, such that $A_0 + A_1 = A$. Of course, the heap property is that: $A_0[i] + A_1[i] \leq A_0[2i] + A_1[2i]$ and $A_0[i] + A_1[i] \leq A_0[2i + 1] + A_1[2i + 1]$.

1. In the first part we will write a protocol such that P_0 and P_1 can do an Insert operation on this shared array (You can increase the size of A_0 and A_1 as new elements arrive). P_0 and P_1 get M_0 and M_1 respectively, such that $M_0 + M_1 = M$. In other words, the parties get additive shares of M. Their goal is to insert M in the secret shared heap.
   (a) In the first step, they both increase the size of the array by 1. What would P_0 and P_1 put in $A_0[n + 2]$ and $A_1[n + 2]$. **5 marks**
   (b) After the first step, why is the heap property not necessarily satisfied? **5 marks**
   (c) Suppose you have access to an *oblivious comparisons* black box, which has the following functionality: P_0 holds (x_0, y_0) and P_1 holds (x_1, y_1). After the end of the protocol, P_0 gets c_0 and P_1 gets c_1. They have the property that, if $(x_0 + x_1 < y_0 + y_1)$, then $c_0 + c_1 = 1$. Otherwise, $c_0 + c_1 = 0$. Use the oblivious comparisons black box to restore the heap property. **20 marks**
2. In the next part we will perform an Extract Min operation.
   (a) The Protocol begins with both P_0 and P_1 removing the element in $A_0[1]$ and $A_1[1]$ respectively. Which element of A_0 and A_1 would now be placed in $A_0[1]$ and $A_1[1]$ by P_0 and P_1 respectively? **5 marks**
   (b) After the first step, why is the heap property not necessarily satisfied? **5 marks**
   (c) Use the oblivious comparisons black box to restore the heap property. **20 marks**