

## CSC263H1: Problem Set 5

Due Tuesday November 5 before 10pm

12

### General instructions

Please read the following instructions carefully before starting the problem set. They contain important information about general problem set expectations, problem set submission instructions, and reminders of course policies.

- Your problem sets are graded on both correctness and clarity of communication. Solutions that are technically correct but poorly written will not receive full marks. Please read over your solutions carefully before submitting them.
- Each problem set may be completed in groups of up to three. If you are working in a group for this problem set, please consult the articles on collaboration and plagiarism on posted on quercus.
- Solutions must be typeset electronically, and submitted as a PDF with the correct filename. **Hand-written submissions will receive a grade of ZERO.**

The required filename for this problem set is **problem\_set5.pdf**.

- Problem sets must be submitted online through CrowdMark. If you haven't used CrowdMark before, give yourself plenty of time to figure it out, and ask for help if you need it! If you are working with a partner, you must form a group on CrowdMark, and make one submission per group. "I didn't know how to use CrowdMark" is not a valid excuse for submitting late work.
- Your submitted file(s) should not be larger than 9MB. You might exceed this limit if you use a word processor like Microsoft Word to create a PDF; if it does, you should look into PDF compression tools to make your PDF smaller, although please make sure that your PDF is still legible before submitting!
- Submissions must be made *before* the due date on CrowdMark.
- The work you submit must be that of your group; you may not use or copy from the work of other groups, or external sources like websites or textbooks.

1. [10 marks] **Amortized analysis: Binomial Heaps**

- (a) Using the accounting method, prove that in a binomial heap, the amortized cost of BINOMIAL-HEAP-INSERT is in  $O(1)$  and amortized cost of BINOMIAL-HEAP-EXTRACT-MIN is in  $O(\log n)$ , where  $n$  is the number of elements in the binomial heap. Only consider sequences containing INSERT and EXTRACT-MIN operations.

In your analysis, let  $\$1$  denote the maximum of:

- the time to create a binomial heap with one element,
- the time to compare the sizes of two binomial trees, the value of their roots, and link them together, and
- the time to update enough pointers to move a tree from one linked list to another.

Your solution should include (at least) a clearly stated *credit invariant*, and clear arguments that the invariant is maintained by the operations.

**Page allowance:** Please neatly present your solution using 1–2 pages. Any additional pages will not be marked.

**2. [15 marks] Amortized analysis: Sets**

Consider the following data structure for representing a dynamic set, supporting only SEARCH and INSERT operations. The elements of the set are stored in a singly linked list of sorted arrays, where the number of elements in each array is a power of 2 and the sizes of all the arrays in the list are different. Each element in the set occurs in exactly one array. The arrays in the linked list are kept in order of increasing size.

To perform SEARCH, perform binary search separately on each array in the list until either the desired element is found or all arrays have been considered.

To insert a new element  $x$  into the set (given the precondition that  $x$  is not already in the set),

- (a) create a new array of size 1 containing  $x$
  - (b) insert this new array at the beginning of the linked list
  - (c) while the linked list contains 2 arrays of the same size, merge the 2 arrays into one array of twice the size
- (a) [5 marks] What is the worst-case time, to within a constant factor, for performing SEARCH when the set has size  $n$ ? Justify your answer.
- (b) [5 marks] What is the worst-case time, to within a constant factor, for performing INSERT when the set has size  $n$ ? Justify your answer.
- (c) [5 marks] Use the accounting method to prove that the amortized insertion time in a sequence of  $n$  insertions, starting with an initially empty set, is  $\Theta(\log n)$ .

**Page allowance:** Please neatly present your solution using 1–2 pages. Any additional pages will not be marked.