

## Assignment 4

### 15 Points

This assignment parallels the second programming assignment - construction of algorithms resulting in implementations of the `MaxBucketSize` and `Analysis` methods of `THashTable<KeyType, DataType, HashType>`, as specified in the Project 2 document.

8. Use the results of the exercises in Assignment 3 to devise algorithms that calculate:
  - (a) The number of non-empty buckets
  - (b) The maximum bucket size
  - (c) The actual average search time (assuming equally likely queries)

Provide an asymptotic analysis of the runtime and runspace of your algorithms.

9. Use the results of the exercises above to devise an algorithm that calculates the actual bucket size distribution  $\{b(k)\}$  from the actual table instance. Provide an asymptotic analysis of the runtime and runspace of your algorithm.
10. Use the results of the exercises above to devise an algorithm that calculates the theoretical best-case bucket size distribution  $\{e(k)\}$  for “simple uniform hashing” based on the number of buckets  $b$  and the number of table entries  $n$ . Provide an asymptotic analysis of the runtime and runspace of your algorithm.

The answers to 8, 9, 10 may be in the form of C or C++ pseudo-code (where arrays index from 0) and should be recognizable from your actual code turned in for Project 2: Hash Analysis.

Assemble your solution paper as follows:

- For each question, repeat the question on the paper, and then provide your solution.
- Take some time to get your math typeset correctly.
- Convert to a pdf document and turn that document in via Blackboard under “Assignment 4”.