EECS 560 - Lab 5 Report

- Name: Jace Kline
- KUID: 2881618

Question Responses

Question 1 - Comparison Between Quadratic Probing and Double Hashing

Upon repeatedly generating the build time statistics of both the quadratic and double-hashing hash tables, there are two clear patterns that can be observed acress the multiple runs:

1. The build time of the quadratic probing table is consistently less for all input sizes than the build time for the double-hashing table. My hypothesis for this trend is that the chosen R-value of 773 (the alternative prime number less than m) is in some way not optimal for the hash table. Additionally, the collision function employed in the double-hashing table requires a greater number of computations. When considering hundreds of thousands of insertions (and therefore collision function calls), the extra complexity of the collision function in the double-hashing case could add up and create non-negligible differences between the performance of that and the less computationally-intensive quadratic probing case.
2. The relation between the increase in input size and the increase in build time is: (1) linear for quadratic probing, (2) clearly greater than linear for double-hashing, and in some cases quadratic. The reasoning for this discrepancy relates back to the reasons that explain pattern 1. The additional complexity of the collision resolution function for quadratic probing gets compounded on increased input size, and there are likely more collisions occurring in the double-hashing case due to a possibly non-optimal R-value.

Question 2 - Analysis of Searching

When observing the success of the search operations between the two different table types, it can be concluded that there are no statistically significant differences between the two. The find success rate appears very similar between both the quadratic probing and double-hashing over repeated runs of the program. Furthermore, it is evident that there exists no correlation between find success rate and input size on either of the tables. In other words, both randomly outperform eachother on different input sizes over many repeated attempts.

Question 3 - Theoretical vs Observed Worst-Case Complexity

To analyze the worst-case expirimental results, I ran the test a large number of times and picked the data set for each hash table type that produced the largest discrepancy between the build time for 100000 elements and 500000 elements. After getting the worst-case expirimental results, I used Excel to graph the build time as a function of the input size. Interestingly, after trying both linear and quadratic line fits for both data sets, it was found that the quadratic probing had a best-fit line that was linear (with R^2 = 0.99) and the double-hashing had a best-fit line that was quadratic (with R^2 = 0.99). These fits allow us to conclude that the worst-case expirimental time complexities of quadratic probing and double-hashing are O(n) and O(n^2) for our expiriment, respectively. From the theoretical perspective, I analyzed the worst-case complexities as follows: I assumed that regardless of the collision resolution technique used, each new insertion would have complexity of n, where n was the current number of elements in the hash table array at a specific time. This is because in the worst-case each element that has attempted to be inserted has collided with all other elements that exist in the table before being inserted. We can iterate this over all insertions and we get a complexity equal to O(n^2) for both the double-hashing and the quadratic probing cases. Hence, the worst-case quadratic probing performed better in practice than in the theoretical model, but the double-hashing scheme performed identically in practice and in theory.