**CSE 373 Homework 4 Write up**

CSE 373 A  
Homework 4  
Fangzheng Sun  
05/13/2015

1. **Who is in your group (Give name, UW NetID & student number of each person)?**

I do not have a group, I work myself.

1. **a) How did you design your tests & what properties did you test?**I test the wellness of both hashtables, whether the method getCount, size and incCount methods are true.

**b) What boundary cases did you consider?**

For some words that are the most frequent ones and some are very infrequent and some are impossible to have.

For incCount, I test both existing and new words. The size should be affected by new words and unaffected by the existing words.

1. **Conduct an experiment to determine which DataCounter implementation (HashTable\_SC, HashTable\_OP) is better for large input texts.**

**a) Describe your experimental setup:**

**1) Inputs used**

For every single test, I input 2 words to the hash tables for 100 times. Then I compute the average time.

**2) How you collected timing information**

Using the provided timing code and print the average time in ms.

**3) Any details that would be needed to replicate your experiments**

For the 2 words to be put in the table, one is an existing one, and another is new for the table (1-100 transfering to string). So The result will be fair .

**b) Experimental Results (Place your graphs and tables of results here).**

for hashtable\_SC, average running time is : 0.075 ms

for hashtable\_OA, average running time is : 0.025 ms

**c) Interpretation of Experimental Results**

**1) What did you expect about the results and why?**

I expect OA should be more efficient since probing avoid further running time in go through nodes in SC.

**2) Did your results agree with your expectations? Why or Why not?**

Yes. The OA works more efficient because probing avoid further running time in go through nodes in SC.

**3) According to your experiment, which Hashtable implementation, separate chaining or open addressing, is better?** Open addressing.

1. **Conduct experiments to determine if changing the hash function affects the runtime of your HashTable.**

**a) Brief description of your hash functions**

Make a new hasher. That is: instead of returning the sum of the ASCII value of the words, the new one returns the

Average value of all character. Then I compare the result.

**b) Experimental Results (Place your graphs and tables of results here).**

**Experiment with at least 2 hash functions (2 Hashing functions = 2 experiments depending on how you measured the runtime)**

**Don’t forget to give each graph a title and label the axes.**

|  |  |  |
| --- | --- | --- |
| Runtime table (ms) | SC | OA |
| The old hasher | 0.15 | 0.025 |
| The new hasher | 0.075 | 0.775 |

**c) Interpretation (Your expectations and why? Did it match your results? If not, why?)**

I expect the result should be unchanged. But the result does not mach. The big change in OA is possibly caused by the limited size of the experiment.

1. **Using Correlator, does your experimentation suggest that Bacon wrote Shakespeare's plays?**

**Show at least one (you can experiment with more texts if you want) correlation value for each of:**

**a) Shakespeare's work compared to Shakespeare's work**

hamlet to the metchant of Venis (act 1) : 2.6988673727921755E-4

**b) Bacon's work compared to Bacon's work**

the new atlantis to Bacon’s articles : 7.936387909302799E-4

**c) Shakespeare's work compared to Bacon's work**

hamlet to the new atlantis: 5.65727366923397E-4

**According to the results of your experiments, did Bacon write Shakespeare's plays?**

Since Bacon’s work to Bacon’s work’s variance is larger thant the result in c), in my experiment, I can not say that whether or not Bacon wrote Shakespeare’s work.

1. **Include a description of how your project goes "above and beyond" the basic requirements (if it does).**

I did not go this part.

1. **If you worked with a partner:**

**a) Describe the process you used for developing and testing your code. If you divided it, describe**

**that. If you did everything together, describe the actual process used (eg. how long you talked**

**about what, what order you wrote and tested, and how long it took).**

**b) Describe each group member's contributions/responsibilities in the project.**

**c) Describe at least one good thing and one bad thing about the process of working together.**

Not with a partner.

1. **a) Which parts of the project were most difficult?**

Implementing two hashtables, especially iterator and resizing.

**b) How could the project be better?**

Make the orientation clearer.

**Appendix**

**Place anything else that you want to add here.**