DEPARTMENT OF COMPUTER SCIENCE, UNIVERSITY OF CAPE TOWN.

Data Structures II: Practical Assignment I (2014)

Anne Kayem

March 25, 2014

1 FOR THIS ASSIGNMENT...

You may either work alone or with one other partner. If you do decide to work with a partner, please make one code and Readme file submission. Both your names should be on the Readme file and you must both agree on the submitted version.

2 OVERVIEW

In class we have just completed studying hash tables and have learnt that one of the applications of hash tables is in search engines. Using the data file (picture.xls) and a database of pictures (CapeTown), write a program to retrieve pictures from the database. The general idea is to use a hash table to store references to the pictures and use the principle of operation of the hash table to optimize looking up a picture.

3 BUILD A HASH TABLE — 10 MARKS

Using any one of the hashing algorithms that we studied in class (linear, quadratic, or separate chaining hashing) build a hash table to store the information in the picture.xls file. Use the following hashing function to do so:

$$\text{Hash(Key)} = (\text{Key mod Prime}) + 1$$

Here Prime is a prime number of your choice - that you can make larger or smaller to suit your needs and Key is basically the reference you would use to store and retrieve your pictures.

4 Creating a link to a Picture — 10 marks

In the second step, expand your hash table to include a reference (this can be just the path to the location of the file) to the picture associated with a given key. In cases where there is no picture, insert some text that will be returned to the user to indicate that there is no associated file.

5 Retrieving a file — 30 marks

Using a combination of the hashing function above and the hashing algorithm of your choice, write a small program to efficiently retrieve a picture from your database.

6 And now some questions.... — 40 marks

1. Explain why you opted to use the hashing algorithm that you have implemented. (5 marks)
2. Do you think a hybrid hashing algorithm that combines techniques from all three algorithms cited, would work better? Explain why or why not. (10 marks)
3. Calculate the number of collisions that occur during the information retrieval process. Explain your result. (10 marks)
4. Did you need to handle rehashing? If so, how did you do it and what was the impact on performance in terms of retrieval? If not, how did you solve the problem and why? (15 marks)

7 Readme File ... — 10 marks

Please provide a well documented readme file to enable us to run and mark your code as well as the supporting hand ins.