Comp 7720 - Online Algorithms

Assignment 2: Compression, Splay Trees, Caching, and $k$-server

Shahin Kamalli
University of Manitoba - Fall 2018

Due: Monday, October 22th at 11:59 pm

October 12, 2018

‘[Computer science] is not really about computers – and it’s not about computers in the same sense that physics is not really about particle accelerators, and biology is not about microscopes and Petri dishes...and geometry isn’t really about using surveying instruments. Now the reason that we think computer science is about computers is pretty much the same reason that the Egyptians thought geometry was about surveying instruments: when some field is just getting started and you don’t really understand it very well, it’s very easy to confuse the essence of what you’re doing with the tools that you use.’ Hal Abelson (CS professor at MIT)

Please pay attention to the followings when preparing/submitting your assignment:

• All problems are written problems. There are six problems with a total of 101 marks.

• The assignment is designed to be shorter than Assignment 1, and hopefully you will find it easier. Most questions require smart observations after which the solution is easy to derive and write.

• If you have any question related to the assignment, you are encouraged to post it on Piazza. Note that you can submit anonymously. Also, instead of emailing me, you can always write a private note in Piazza. It is likely that I drop hints when a question is posted publicly on Piazza (because all students can benefit from it). It is not the case when you ask questions in emails or during office hours.

• You are welcome to discuss the problems with your friends (or enemies). But you should write your answers individually. You might be interviewed about your answers. Be careful not to accidentally copy.

• Submit your answers electronically using UMLearn. You should submit a single pdf file. You are encouraged to prepare your assignment using $\LaTeX$. The $\LaTeX$ file for this assignment is posted.
Problem 1  Compression \( [5+5+5+5 = 20 \text{ marks}] \)

a) Apply the Burrows-Wheeler transform on the following string; show your work and the output

\[ \text{BARBAPAPA}\$ \]

Assume $ precedes all characters when you sort rotations.

b) Assume an initial list $ \rightarrow A \rightarrow B \rightarrow P \rightarrow R$, i.e., initially $ is at index 0, A is at index 1, etc. Assume we use Move-To-Front on the above list to encode the outcome of the BWT transform from part (a). Show what numbers will be encoded (you need to show how the list is updated).

c) Assume an initial list $ \rightarrow A \rightarrow B \rightarrow P \rightarrow R$, i.e., initially $ is at index 0, A is at index 1, etc. Assume we use Timestamp on the above list to encode $BA$\$APBP$. Show what numbers will be encoded (you need to show how the list is updated).

d) Assume an initial list $ \rightarrow A \rightarrow B \rightarrow C \rightarrow D$. A compressing scheme that uses Move-To-Front has encoded the following numbers for a text $T$. Show what the actual text is. The numbers are 3 0 1 1 2.

Problem 2  Splay Trees \( [7+7+7 = 21 \text{ marks}] \)

a) Apply the splay operation on the following splay tree when there is a request to node ‘30’. Show your steps.

b) Prove or disprove the following statement: “the root of a splay tree has always two children”.

c) Prove or disprove the following statement: “after a splay operation, the old root is always at depth 1 or 2 (i.e., it is at distance 1 or 2 of the new root)”.

Problem 3  Paging & Resource Augmentation \( [15 \text{ marks}] \)

Sometimes when we analyse online algorithms, we reduce the power of Opt to be more fair to the online algorithm; it is called resource augmentation. For example, for the paging algorithm, instead of comparing an online algorithm with an optimal algorithm which has the same cache-size, we assume the size of the cache of Opt is smaller than that of algorithm. Clearly, when comparing with a weaker optimal algorithm, the competitive ratio of algorithms is expected to improve.

Assume the cache of a marking algorithm A has size $k$ and the cache of Opt has size $k/2$. Prove that the competitive ratio of A is at most 2 under this setting (note that this is a big improvement over the competitive ratio $k$ in the classic setting without resource augmentation).

Problem 4  Double-Coverage-Algorith for Cycles \( [15 \text{ marks}] \)

Note that the double-coverage algorithm is well-defined for cycles: for each request to a node $x$, move the two servers on its two sides (in the cycle) with the equal speed towards $x$.

Show that DCA has unbounded competitive ratio for cycles. You might prove this by considering the problem for a cycle of length 5 with $k = 3$ servers in it.
Problem 5  Competitive k-server Algorithm for Cactus and Block Graphs [15 marks]

From the following two questions, choose one and submit a solution for it. There is no advantage in submitting solutions to both:

- A block graph is a graph in which every biconnected component (that is, any collection of vertices for which there are at least two paths between any pair of vertices) is a clique (consult Wikipedia to see a picture of a block graph). Prove the k-sever conjecture for block graphs, that is, describe that there is a deterministic algorithm that has a competitive ratio of k for these metrics (your solution should include a description on why the algorithm is k-competitive).

- A cactus graph is a graph in which every biconnected component (that is, any collection of vertices for which there are at least two paths between any pair of them) is a cycle. In other words, it is a graph in which every edge belongs to at most one cycle (consult Wikipedia to see a picture of a cactus graph). Provide a randomized k-server algorithm with competitive ratio of at most 2k for cactus graphs (your solution should include a description on why the algorithm is at most 2k-competitive).

Problem 6  Balance for Paths [15 marks]

We learned in the class that Balance is not a competitive algorithm. The example we saw in the class was based on the metric which was a cycle. A clever student comes to the conclusion that Balance might be competitive when the metric space is a tree or a path. Show that he is wrong. For that, consider a path of length m (assume vertices are labelled from 1 to m) and show a worst-case sequence for the 2-server problem which shows Balance is not competitive.