Assignment 3

Kyle Cheung

April 10, 2018

Question 1

$findChom(String)$
$s \leftarrow String$
for S[i,i+1] in S **do**
    if S[i,i+1] = LR/SR/LA/SS **then**
        $replace$
        $findChom(newString)$

String w = (())()= LLRRLR
$S \rightarrow SS \rightarrow SLR \rightarrow LALR \rightarrow LSRLR \rightarrow LLRRLR$

Question 2

$InputG$
G'
while there are vertices in G **do**
    $a \leftarrow SCC(G)$ ▷ where u,v is an element of a
    $G'.add(a, u, v)$
$TopologicalSort(G')$
for i less than number of Vertices **do**
    if $!G'.hasEdge(vi, vi + 1)$ **then**
        $false$
$true$

If the graph is semi connected for a pair (u,v) such that there is a path from u to v, Let V and U be there SCC. Since all nodes are strongly connected, there must be a path from v to u.

Question 3

Create a queue of buckets sorted by their distance value such that vertex v can be found in bucket d[v]. Scan the buckets, when a non-empty bucket is found, remove the first vertex and relax all adjacent vertices. Repeat the process until queue is done. Since we relax a total of E edges, the running time is O(V+E).

Question 4

Essentially we would like to create a minimally spanning tree. For example we could use Kruskals Algorithm, where n stations are represented as vertices and the path from one station to the next is represented as an Edge using the energy required to transfer data as the edges weight. Using this method we can easily calculate how to connect each station such that we use the minimal amount of energy.

Question 5

A-16.1. Allow all residents without a puppy to like all the dogs they prefer and add them to that residents priority list. If the puppy has not yet been adopted and no one has liked the puppy then the puppy is adopted by the resident and the round ends. If a puppy has multiple likes then they are adopted by whoever has the least amount of liked puppies This continues until each resident is matched.

A-16.2. Allow all residents without 3 puppies to like all the dogs they prefer and add them to the residents priority list. if the puppy has not yet been adopted yet and no one has liked the puppy then that puppy is adopted and this ends the round. If a dog has multiple likes they are adopted by whoever has liked the least amount of puppies. this continues until each resident has had 3 puppies.

Question 6

Using the same equation as the birthday problem just changing the number of days in a year to the possible number of combinations which in this case for a 3 digit lock would be 999. therefore we can calculate that the minimum number of apartments to have a greater than 50

Question 7

```
text: "Dostoyevsky.txt"
pattern: "the"
occurrences: 11411
Suffix Array: 0.000160932540894s
Brute Force: 0.200074911118s
Boyer-Moore: 0.27873301506s
text: "Dostoyevsky.txt"
pattern: "Raskolnikov"
occurrences: 784
Suffix Array: 0.00032901763916s
Brute Force: 0.185318946838s
Boyer-Moore: 0.0860641002655s
text: "Dostoyevsky.txt"
pattern: "dishevelled"
occurrences: 1
Suffix Array: 0.000158786773682s
Brute Force: 0.191564083099s
Boyer-Moore: 0.0762679576874s
vw CS3340 $ python patternMatching.py E.coli.txt E.coli.sa E.coli.pat
```

```
text: "E.coli.txt"
pattern: "AAA"
occurrences: 104915
Suffix Array: 0.000204086303711s
Brute Force: 0.827691078186s
Boyer-Moore: 1.1095290184s
text: "E.coli.txt"
pattern: "ACGT"
occurrences: 13733
Suffix Array: 0.000188112258911s
Brute Force: 0.751274108887s
Boyer-Moore: 1.87848210335s
text: "E.coli.txt"
pattern: "NNNNNNNNNNNNNNNNNNNNNNNNNNNN"
occurrences: 11406
Suffix Array: 0.000174999237061s
Brute Force: 0.70400595665s
Boyer-Moore: 0.176497936249s
text: "E.coli.txt"
pattern: "ATATCAGTTATATTTAAACTAAATTAAAGTCATGAATAAT"
occurrences: 1
Suffix Array: 0.000186920166016s
Brute Force: 0.822018861771s
Boyer-Moore: 0.646251916885s
```