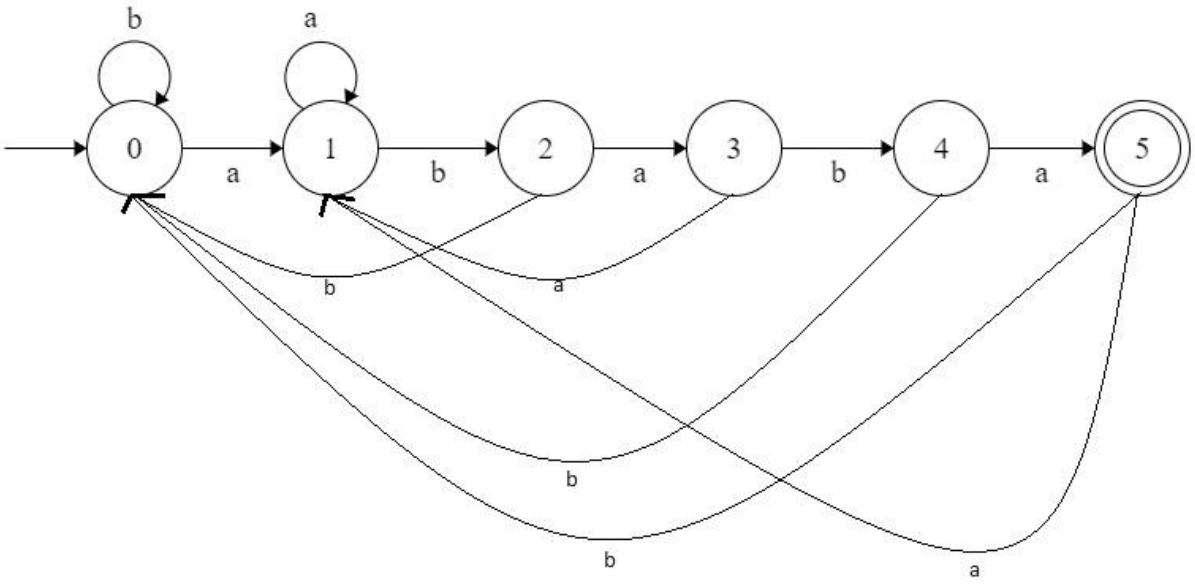
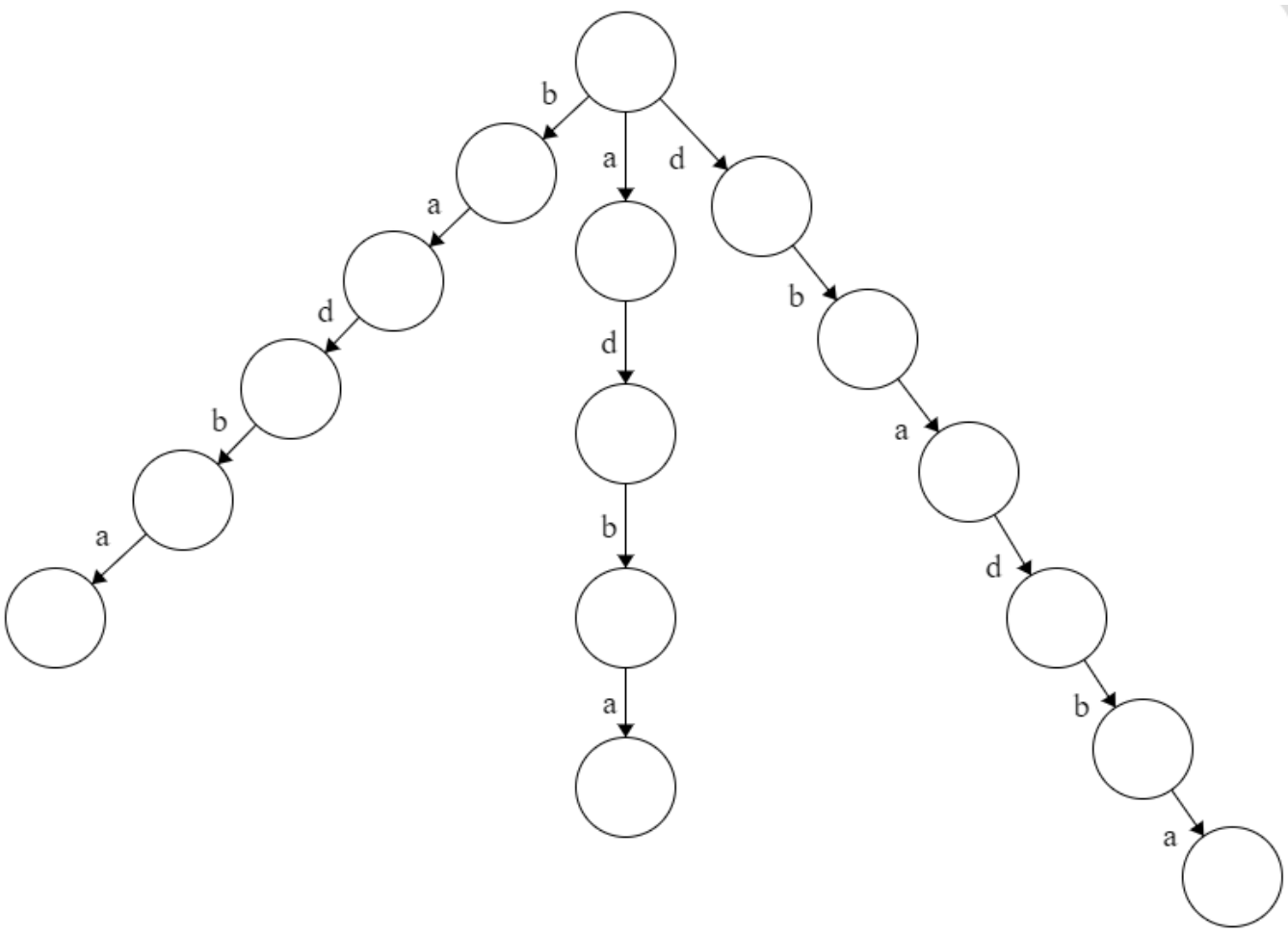
1. The string matching automaton:

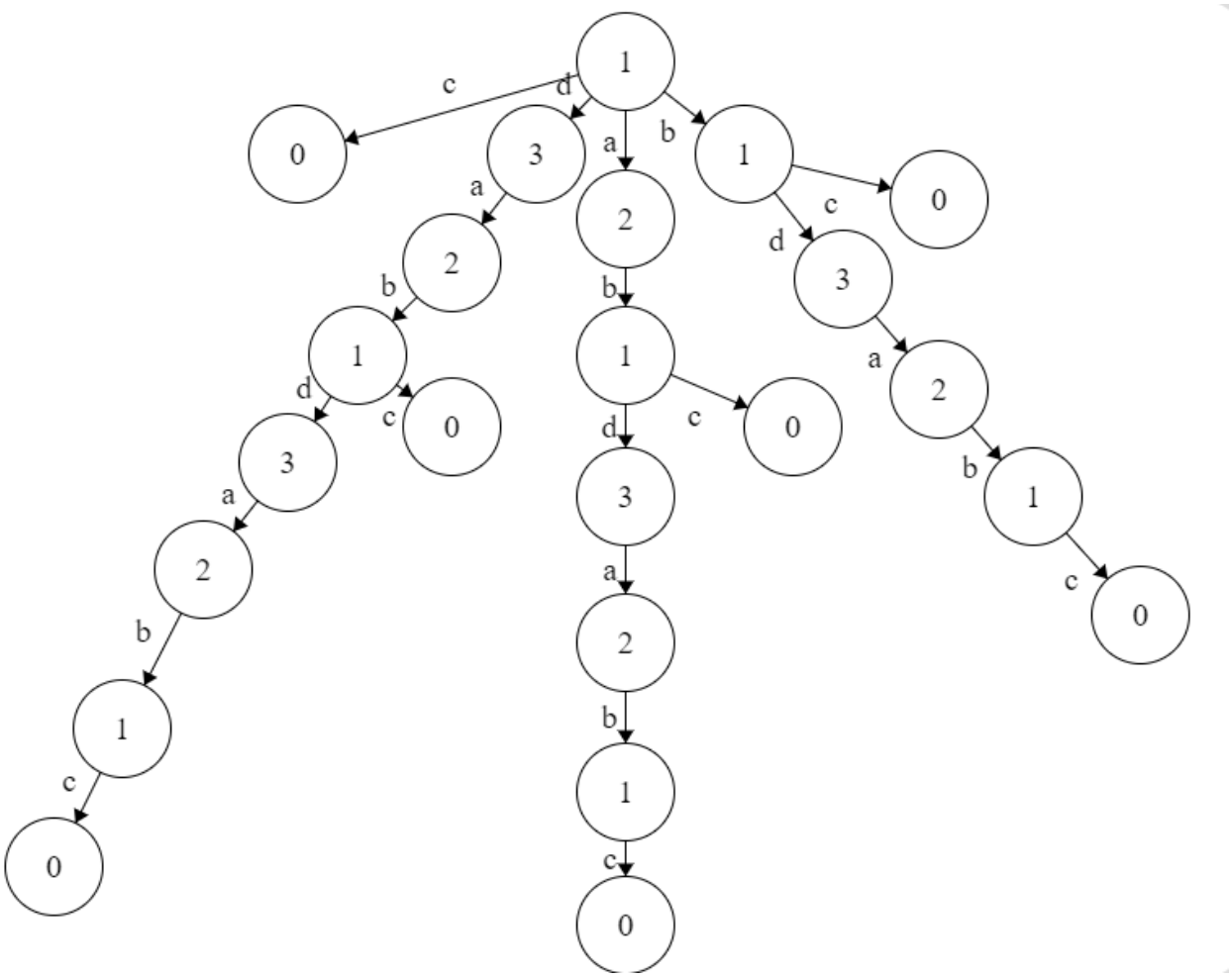


2. I assumed that the first character of the text is index 1, and not zero, just like the algorithm 2 of the notes. So, we trace the algorithm for each of the texts:
   - aababaa:
     state: 0 (initial)
     input: a, state: 1
     input: a, state: 1
     input: b, state: 2
     input: a, state: 3
     input: b, state: 4
     input: a, state: 5 (here we achieve the final state, so the algorithm stores 6-5+1)
     input: a, state: 1
     The output would be I = {2}, which is the index that our pattern starts in the string.
   - babaaa:
     state: 0 (initial)
     input: b, state: 0
     input: a, state: 1
     input: b state: 2
     input: a, state: 3
     input: a, state: 1
     input: a, state: 1
     The output would be I = {}.

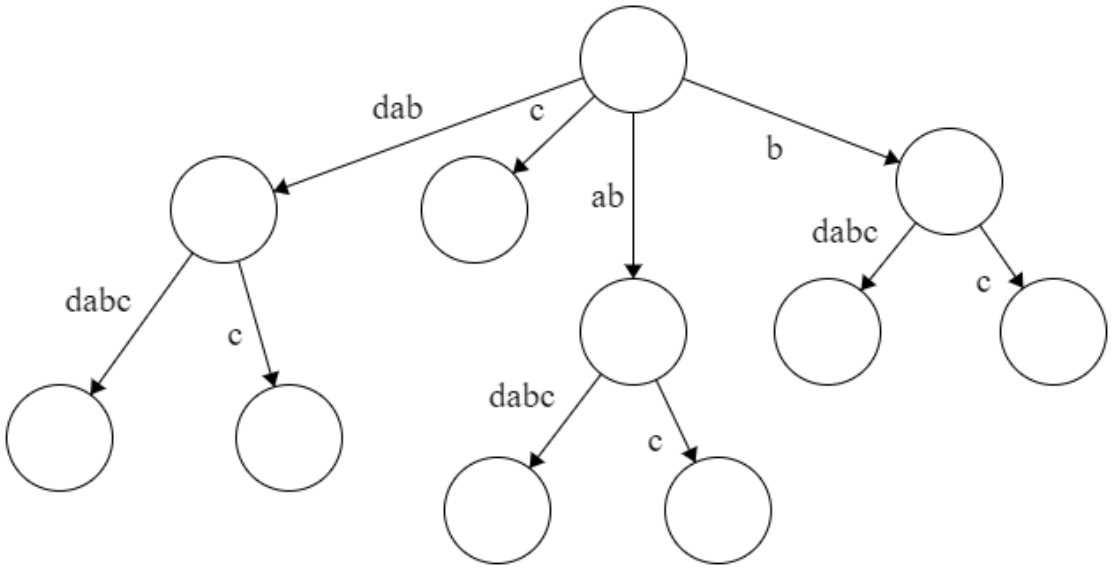
3. These are the suffixes: {a, ba, dba, adba, badba, dbadba}.
   So the suffix trie:



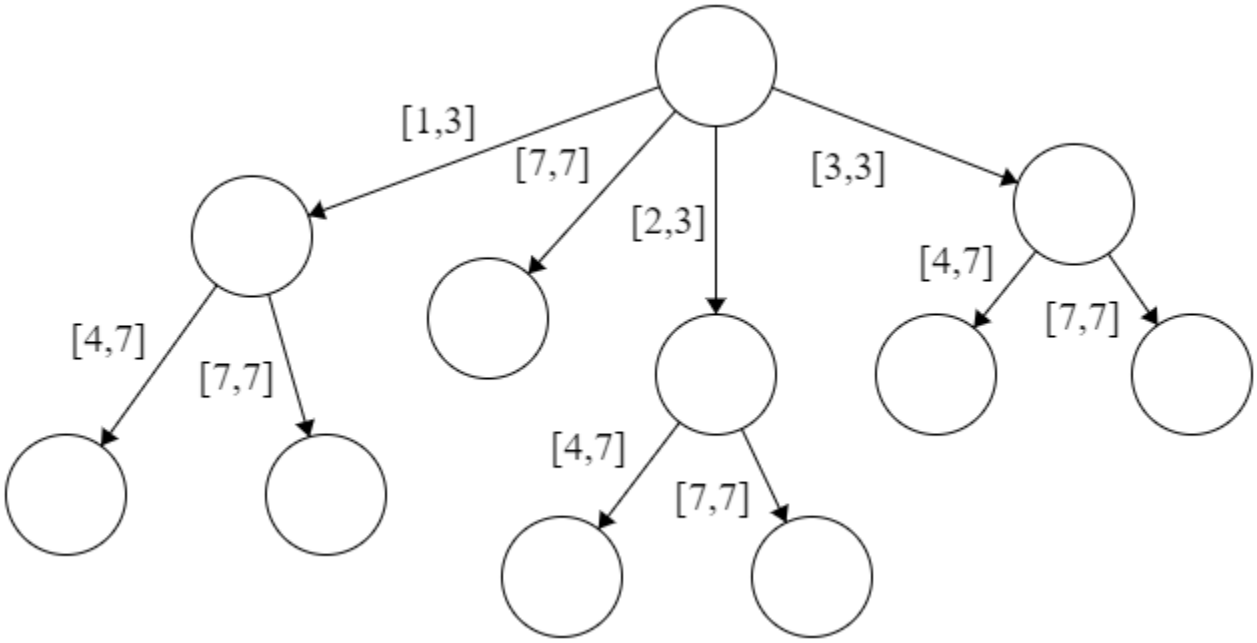
4. After step 1: (the values that I write on vertices, are suffix distance values that we compute in algorithm 3, called C_v)


1
c
d
a
b
0
3
1
2
0
c
d
a
2
b
3
b
1
1
d
c
0
a
2
3
c
0
b
1
d
3
c
0
a
2
b
1
2
c
0
b
1
c
0

After step 2:


dab
c
b
ab
dabc
dabc
c
dabc
c

After step 3: (assume the first index is 1)



5. I do every steps just like the question, and I've used the find_all method of the suffix tree to gather all the positions of a 9mer in the chromosome, then length of that list shows the number of occurrences of that 9mer. Running the code took about 2 minutes, so I just check how much of that is for which part of the code:

```
create string from fasta file time: 3.0997605323791504 seconds.
create suffix tree time: 107.90303993225098 seconds.
generate all 9mers time: 0.07405686378479004 seconds.
searching all 9mers in the suffix tree time: 62.2614803314209 seconds.
sorting results time: 0.0 seconds.
```

Here's the output, shows the most frequent 9mers in the first 10000000 characters of the chromosome:

AAAAAAAAA : 13859

TTTTTTTTT : 12784

CCAGCCTGG : 2875

CCAGGCTGG : 2784

GAGGCTGAG : 2523

GGAGGCTGA : 2430

AGGCTGAGG : 2374

AATCCCAGC : 2293

GTGTGTGTG : 2243

CCTGTAATC : 2235

TGTGTGTGT : 2228

TAATCCCAG : 2227

CTCAGCCTC : 2221

CTGTAATCC : 2217

TGTAATCCC : 2181

GTAATCCCA : 2172

CTGGGATTA : 2171

GCTGGGATT : 2165

GGATTACAG : 2161

GATTACAGG : 2152

TCAGCCTCC : 2142

GGGATTACA : 2135

CCTCAGCCT : 2134

ACACACACA : 2133

TGGGATTAC : 2103

CACACACAC : 2101

CAGCCTGGG : 2092

TTGGGAGGC : 2087

CCCAGGCTG : 2081

GCCTCCCAA : 1989

GGGAGGCTG : 1968

GGCTGAGGC : 1925

CTCCAGCCT : 1806

GCCTCAGCC : 1760

CAGCCTCCC : 1759

TCCAGCCTG : 1720

GCCTGTAAT : 1707

TTTGGGAGG : 1688

AGGCTGGAG : 1678

CAGGCTGGA : 1654

ACTCCAGCC : 1639

CCTCCCAAA : 1637

CCAGCTACT : 1617

AAATACAAA : 1608

TCCCAGCTA : 1606

CTTTGGGAG : 1605

TGCAGTGAG : 1603

ATTACAGGC : 1588

ACTTTGGGA : 1582

AAAAATTAG : 1581

GCTGAGGCA : 1579

AAAATTAGC : 1576

CTGAGGCAG : 1574

AGGCAGGAG : 1568

AAAATACAA : 1558

CACTCCAGC : 1557

CTCACTGCA : 1547

CCCAGCTAC : 1545

GCAGTGAGC : 1540

CACTTTGGG : 1540

AAAAATACA : 1538

AATACAAAA : 1538

CTCCCAAAG : 1535

TGCACTCCA : 1530

TGAGGCAGG : 1525

GGCTGGAGT : 1522

GCACTCCAG : 1512

TCCCAAAGT : 1511

TAGCTGGGA : 1500

CAAAAAAAA : 1496

TCCCAGCAC : 1492

GAGGCAGGA : 1491

TGCCTCAGC : 1486

CTGCCTCAG : 1479

GTGCTGGGA : 1476

GCTAATTTT : 1476

CTCCTGCCT : 1476

CCCAAAGTG : 1471

TGCTGGGAT : 1468

ATCCCAGCA : 1466

AGCACTTTG : 1460

TTTGTATTT : 1459

CCCAGCACT : 1459

GTAGCTGGG : 1458

AGTAGCTGG : 1454

TGGAGTGCA : 1454

GCTCACTGC : 1450

GCCCAGGCT : 1450

CTAATTTTT : 1444

CAAAGTGCT : 1442

AGTGCTGGG : 1440

CAGCACTTT : 1439

CTGGAGTGC : 1439

GCTGGAGTG : 1437

GCACTTTGG : 1435

AAAGTGCTG : 1431

AAGTGCTGG : 1428

CCTGCCTCA : 1427

TTTTTTTG : 1417

AAAAAAAAG : 1414

TTGTATTTT : 1411

TCCTGCCTC : 1411

CCAAAGTGC : 1410

ATACAAAAA : 1407

CCAGCACTT : 1407

CCACTGCAC : 1400

TTTTGTATT : 1390

AGCCTGGGC : 1378

TCTACTAAA : 1376

TGTATTTTT : 1374

CTACTAAAA : 1369

TACTAAAAA : 1356

ACCAGCCTG : 1331

GTGCAGTGG : 1327

CAGCCTGGA : 1316

CTCTACTAA : 1304

CTTTTTTTT : 1292

CACTGCACT : 1288

TTGCAGTGA : 1286

AGACCAGCC : 1285

CTAAAAATA : 1284

TTTTTGAGA : 1275

TTTTTAGTA : 1274

ACTAAAAAT : 1273

TCTCTACTA : 1267

GGCAGGAGA : 1267

GACCAGCCT : 1265

TCTCAAAAA : 1264

TCACTGCAA : 1261

CAGGCTGGT : 1261

TTTAGTAGA : 1256

TTTTTGTAT : 1256

TAAAAATAC : 1254

TTTTAGTAG : 1252

GCAGGAGAA : 1251

CTGCACTCC : 1239

GAGCATCTG : 1238

AGTGCAGTG : 1237

ACTGCACTC : 1232

TGGGAGGCC : 1224

GGCTGGTCT : 1224

AGGCTGGTC : 1222

AGCATCTGA : 1219

TCTCCTGCC : 1217

CAGGAGAAT : 1214

GGCCTCCCA : 1209

TGGGAGGCT : 1203

TTCTCCTGC : 1202

TTAGTAGAG : 1201

GGAGTGCAG : 1194

CTCAAAAAA : 1194

TTTTTTGAG : 1191

TATTTTTAG : 1186

ATTTTTAGT : 1178

TCAAAAAAA : 1177

ATTCTCCTG : 1170

TAGTAGAGA : 1169

AGCCTCCCA : 1167

GAGTGCAGT : 1160

AGGTCAGGA : 1153

ACAGCCTGG : 1149

GTATTTTTA : 1142

TTTTTTTGA : 1127

GGTCAGGAG : 1124

GTGGCTCAC : 1120

GACAGCCTG : 1106

CAGCTACTC : 1103

CAGTGAGCC : 1097

GGTGGCTCA : 1087

TCCTGACCT : 1085

TGCCCAGGC : 1077

GTGAGCCAC : 1071

AAAAAGAAA : 1070

GAGGTCAGG : 1055

CAGGTGAGC : 1055

TATATATAT : 1052

AGGTGAGCA : 1046

GCATCTGAC : 1043

ATATATATA : 1038

CTCCTGACC : 1036

TGAGCCACC : 1034

AAAAAAAGA : 1033

CAGCCTGGC : 1026

CCTGACCTC : 1025

CACCCCCAG : 1019

TCTCTCTCT : 1017

GGCTCACTG : 1012

GCCTGGCCA : 1012

GGTGAGCAT : 1007

GTGAAACCC : 1006

CAGCACCCA : 1004

GCCTGGGCA : 1003