# Συστήματα Ανάκτησης Πληροφορίας
## 2η σειρά ασκήσεων

*Φοιτητής:*
Ιωάννης Κανακαράκης p3060190

## Άσκηση 3

Για το σκοπό της άσκησης δημιουργήθηκε μια νέα γενική βιβλιοθήκη πάνω από τη βιβλιοθήκη Lucene με δομές και αντικείμενα κατάλληλα για την ανάλυση, επεξεργασία, αποθήκευση και εκτύπωση των επιθυμητών στοιχείων. Η βιβλιοθήκη αυτή χρησιμοποιήθηκε για τη δημιουργία ενός εκτελέσιμου που φέρει εις πέρας το σκοπό της άσκησης.
Ο κώδικας της εργασίας βρίσκεται εδώ, μαζί με τα αποτελέσματα και τα πηγαία αρχεία. Ο κώδικας διατίθεται ελεύθερα με την άδεια χρήσης GNU GPLv3.

Όπως φαίνεται και στο output (όπως δίδεται παρακάτω) αυτό που κάνει ο κώδικας του εκτελέσιμου αρχείου είναι:

1. Διαβάζει το αρχείο της συλλογής cacm και το φορτώνει σε κατάλληλη δομή.
2. Γράφει την συλλογή σε xml αρχείο.
3. Επιβεβαιώνει την ακεραιότητα της συλλογής διαβάζοντας την ξανά από το xml αρχείο.
4. Διαβάζει τις επερωτήσεις για τη συλλογή cacm και τις φορτώνει σε κατάλληλη δομή.
5. Δημιουργεί ένα ευρετήριο και ρωτά τις επερωτήσεις επιστρέφοντας και αποθηκεύοντας τα αποτελέσματα σε κατάλληλη δομή.
6. Αποθηκεύει τα αποτελέσματα σε αρχείο επεξεργασμένα κατάλληλα ώστε να διαβάζονται από το 'trec_eval'
7. Διαβάζει τις πληροφορίες σχετικότητας επερωτήσεων-κειμένων της συλλογής και τις φορτώνει σε κατάλληλη δομή
8. Γράφει τις πληροφορίες σχετικότητας επερωτήσεων-κειμένων σε αρχείο επεξεργασμένες κατάλληλα ώστε να διαβάζονται από το 'trec_eval'
9. Καλεί το 'trec_eval' αξιολογώντας τα αποτελέσματα των επερωτήσεων.
10. Αποθηκεύει τα αποτελέσματα της αξιολόγησης σε αρχείο

```
Parsing cacm documents from file: data/cacm/cacm.all
Writing cacm documents to xml file: data/results/cacm.all.xml
Loading cacm documents from xml file: data/results/cacm.all.xml
Parsing cacm queries from file: data/cacm/query.text
Searching cacm documents with cacm queries
Writing trec-formated results to file: data/results/trec_searchresults
Parsing cacm qrels from file: data/cacm/qrels.text
Writing treq-formated qrels to file: data/results/trec_qrels
Evaluating results with trec_eval
Evaluation results are in file: data/results/trec_results
```

Επιπλέον έχω προσθέσει δύο γραμμές σε σχόλια που εκτυπώνουν τα αποτελέσματα των επερωτήσεων στην οθόνη. Αν αφαιρέσετε τα σχόλια μπορείτε να δείτε το αποτέλεσμα σχετικότητας, το κωδικό του κειμένου και το τίτλο του, για τα πρώτα 20 αποτελέσματα για κάθε επερώτηση.

```
//          System.out.println("Printing results to output");
//          printSearchResults(results);
```

→ *παράδειγμα:*

```
[...]
-----------------------------------
Searching for: 55 - Anything dealing with star height of regular languages
or regular expressions or regular events.
-----------------------------------
  2650 - 1.760452      Order-n Correction for Regular Languages
  1739 - 1.118815      Regular Expression Search Algorithm
  2921 - 1.118815      Regular Right Part Grammars and Their Parsers
  1355 - 0.839111      Regular Coulomb Wave Functions (Algorithm 292 )
  1896 - 0.839111      Regular Coulomb Wave Functions (Algorithm 292 S22])
  1898 - 0.839111      Regular Coulomb Wave Functions (Algorithm 292 [S22])
  1993 - 0.839111      Regular Coulomb Wave Functions (Algorithm 292 $S22))
  2889 - 0.134752      Performance of Height-Balanced Trees
  1694 - 0.101064      An Algorithm for the Probability of the Union of a
Large Number of Events
  1846 - 0.101064      On Simulating Networks of Parallel Processes in
Which Simultaneous Events May Occur
  2839 - 0.101064      An Insertion Technique for One-Sided Height-Balanced
Trees
  3009 - 0.101064      Insertions and Deletions In One-Sided Height-
Balanced Trees
  3056 - 0.101064      Counting Large Numbers of Events in Small Registers
[...]
```

Όπως παρατηρείτε όλη η διαδικασία γίνεται μέσα από το κώδικα της Java.
Το 'trec_eval' είναι εκτελέσιμο αρχείο:

```
/usr/bin/trec_eval: ELF 64-bit LSB executable, x86-64, version 1 (SYSV),
dynamically linked (uses shared libs), for GNU/Linux 2.6.18, stripped
```

Η Java μέσω της βιβλιοθήκης 'Runtime' μας δίνει τη δυνατότητα να τρέξουμε άλλα εκτελέσιμα αρχεία. Έτσι το 'trec_eval' καλείται μέσα από το εκτελέσιμο αρχείο της Java. Δημιουργούνται δύο νέα streams που παρακολουθούν τα std::err και std::out streams του 'trec_eval'. Το std::out παράγει τα αποτελέσματα και τα αποθηκεύει σε αρχείο, ενώ το std::err παράγει μηνύματα λάθους και τα εκτυπώνει στην οθόνη.

→ Τα αποτελέσματα του 'trec_eval' φαίνονται παρακάτω. Έγιναν δύο δοκιμές:

- Με χρήση stemming/stopwords

```
runid                   all   RUN
num_q                   all   43
num_ret                 all   860
num_rel                 all   719
num_rel_ret             all   143
map                     all   0.1786
gm_map                  all   0.0471
Rprec                   all   0.2191
bpref                   all   0.2998
recip_rank              all   0.5786
iprec_at_recall_0.00    all   0.6102
iprec_at_recall_0.10    all   0.4399
iprec_at_recall_0.20    all   0.3152
iprec_at_recall_0.30    all   0.2397
iprec_at_recall_0.40    all   0.1567
iprec_at_recall_0.50    all   0.1024
iprec_at_recall_0.60    all   0.0754
iprec_at_recall_0.70    all   0.0638
iprec_at_recall_0.80    all   0.0631
iprec_at_recall_0.90    all   0.0631
iprec_at_recall_1.00    all   0.0631
P_5                     all   0.3023
P_10                    all   0.2488
P_15                    all   0.1984
P_20                    all   0.1663
P_30                    all   0.1109
P_100                   all   0.0333
P_200                   all   0.0166
P_500                   all   0.0067
P_1000                  all   0.0033
```

- Χωρίς stemming/stopwords

```
runid                   all   RUN
num_q                   all   43
num_ret                 all   860
num_rel                 all   719
num_rel_ret             all   116
map                     all   0.1323
gm_map                  all   0.0237
Rprec                   all   0.1765
bpref                   all   0.2281
recip_rank              all   0.5040
iprec_at_recall_0.00    all   0.5299
iprec_at_recall_0.10    all   0.3896
iprec_at_recall_0.20    all   0.2547
iprec_at_recall_0.30    all   0.1307
iprec_at_recall_0.40    all   0.0729
iprec_at_recall_0.50    all   0.0588
iprec_at_recall_0.60    all   0.0465
iprec_at_recall_0.70    all   0.0465
iprec_at_recall_0.80    all   0.0465
iprec_at_recall_0.90    all   0.0465
iprec_at_recall_1.00    all   0.0465
P_5                     all   0.2233
P_10                    all   0.1860
P_15                    all   0.1504
P_20                    all   0.1349
P_30                    all   0.0899
P_100                   all   0.0270
P_200                   all   0.0135
P_500                   all   0.0054
P_1000                  all   0.0027
```

Δείτε όλα τα αρχεία ολόκληρα στον ιστότοπο της εργασίας.
Ενδεικτικά παραδείγματα υπάρχουν στον φάκελο `output`.