Coursework #1

Instructor: Walid Magdy

Required

- Implement a simple IR tool, that includes
  - Preprocessing of text
    - Tokenisation
    - Stopping
    - Stemming
  - Positional inverted index
  - Search execution module that allows:
    - Boolean search
    - Phrase search
    - Proximity search
    - Ranked IR (TFIDF)
**CW1 depends on**

- **Lectures:**
  - Lecture 4: Preprocessing
  - Lecture 5: Indexing
  - Lecture 7: Ranked IR
- **Labs:**
  - Lab 1: Preprocessing
  - Lab 2: Indexing and Query execution
  - Lab 3: Ranked IR
- **Note:** By implementing Lab 3, you should have CW1 almost ready

---

**Deliverables**

- **Code ready to run:**
  - Preferred: Perl, Python
  - Allowed: Java
  - Other languages are fine, but please ask for approval first
- **Report (2-4 pages):**
  - Includes: modules implemented and role of each
  - Why you selected to do each step in this way?
- **Search Results file:**
  - Files containing the search results of provided queries
**Assessment**

- To be considered:
  - Search results (automatic marking)
  - Quality of report and explanation for code

- Not highly considered:
  - Speed of the system (unless unreasonably slow!)
  - Quality of code

**Allowed/not allowed**

- Allowed:
  - Use libraries for Porter stemming
  - Use ready code for optimisation
  - Discuss some functions with your friends
  - Use Piazza to ask question on implementation

- Not Allowed:
  - Copying code from each other!
  - Share results by any mean!
**Timeline**

- **05 Oct 2020**
  *Initial announcement of CW1*
  *Full details of CW1 to be released*

- **22 Oct 2020**
  *Test Set Release: the test data collection to run your code on and submit the results*

- **Sunday, 25 Oct 2020, 11:59:59pm**
  *Submission deadline*

---

**Notes**

- CW1 weight = 10% (only)
- Effort is high, but ..
- Full support through labs 1, 2, and 3
- Less details = more flexibility
- Good practice to build a system from scratch
- Once done: you built a search engine
- Next CW: will be not covered by labs (hence higher weight)
Advices

- Lab 2 + Lab 3 = CW 1
- Implement carefully
- Write efficient & clean code
- Change preprocessing & observe change!
- Test & test & test
- Keep your system as a project to add on as we go in the course