Lucene

* framework used to allow rapid development of IR tools. It is a set of Java classes.
* Two main classes: Analyzer and IndexWriter.
* Analyzer: Used to tokenize the data. There are several tokenizers build into the framework to use.
* IndexWriter: Used to create an index, functions to add new documents and functions to optimize index.
* There are functions to also search through an index:

1. Open Index
2. Create Searcher
3. Assign Analyzer: works with the query by indexing it the same way as the documents
4. Create and parse query
5. Determine which documents match

The web structure gives a document more dimension. It is used to determine which document is likely to be more relevant.

Some nodes are visited more than others. Due to having more in links for example. Page Rank calculates a score for each page.

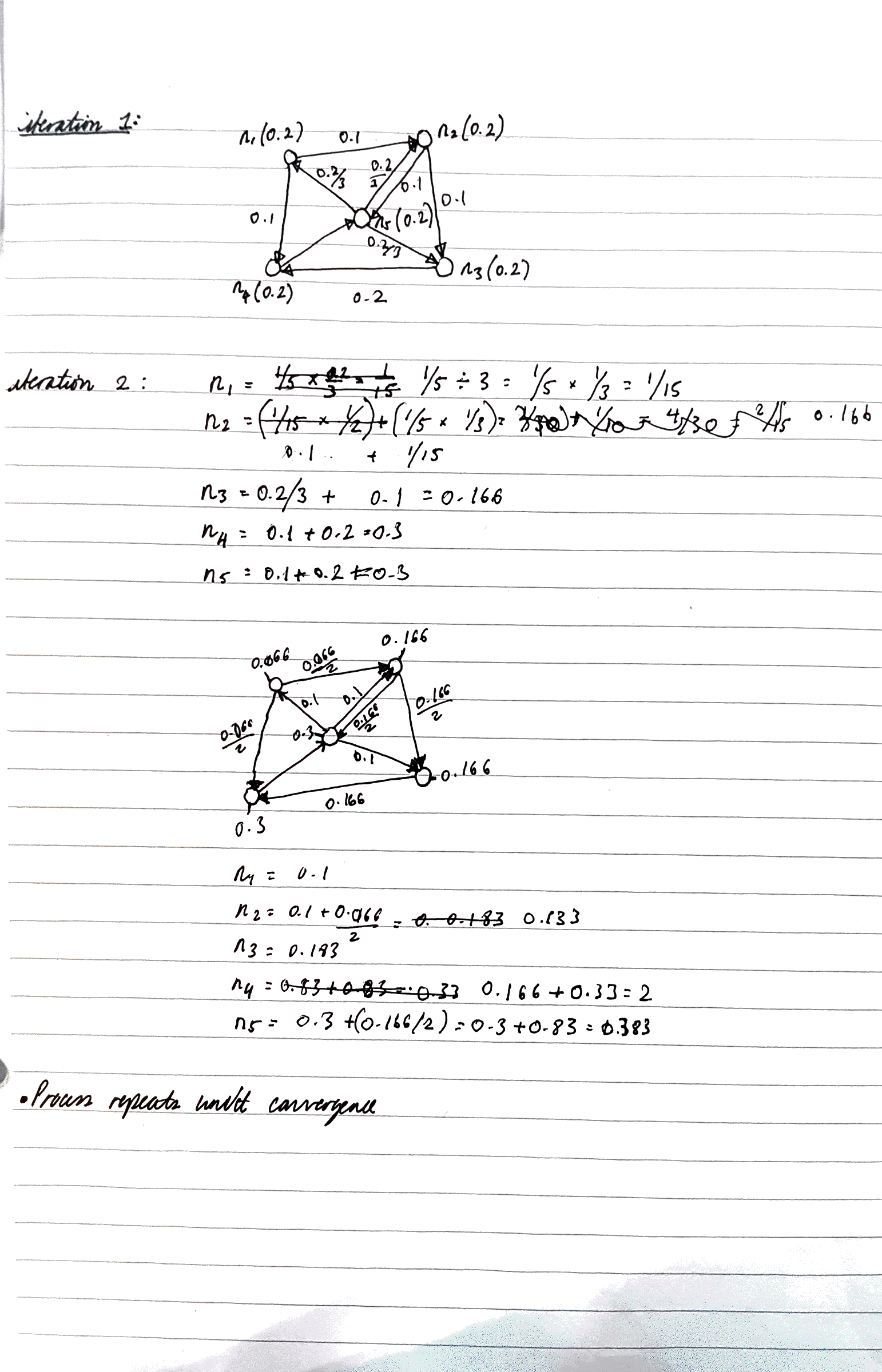
For example page rank of A depends on the page rank of its IN-links and, the page ranks of the IN-links of those IN-links and so on. So, the calculation of a Page Rank is recursive.

85% of the traffic to a page is from a random click on OUT link from a page.

15% of the traffic is due to jumping to the page absolutely.

PageRank: definition



* IN-links t1,…,tn
* C(t) is the out-degree of t
* A is probability of a random jump
* N is the total number of nodes in a graph.
* Explanation on using C(t): If a page that is an in-link to a page that has more out links than a page (in-link) with fewer out-links, then its contribution to the popularity of that page diminishes and contributes less than the in-link page with fewer out-links. If you think of it in a real-world scenario, if your friend (in-link) has many friends (out-links) then they are more likely to have a weaker relationship with you.
* Calculation of PageRank is done until the values do not change much.
* The example below does not take into account the random jumps
* And lost PageRank mass due to dangling nodes (nodes which do not have out-links), the mass of these kind of nodes need to be distributed evenly over all nodes.
*  PageRank gives the global importance of queries independent of the query
* Ranking of results is improved by combining the page rank score with the tf-idf of each document: weight(term, document) = tf-idf(term, doc) x PR(DOC).

Completed task 3 of workshop 5.

Need to complete task 4.

