Group Project Design and Implementation of a Search Engine Report

# Overall

# Gathering Information

We design and write a program named WebSpider to gather the information form the given URL. The program will ask the user to input the URL, the value of X and Y and the program will run on its own. For demonstration purpose, the program will ask if you want to run the demo with the URL set as the main page of HKBU, X = 10, and Y = 100.

Every webspider will have these attributes:

1. **urlString** , a string that stores the url string of the current page;
2. **url**, URL object defined in Java API used to get the html file;
3. **domain**, a string that stores the domain of the current page;
4. **x**, an integer that the spider should know how many links it should be extract from a page;
5. **y**, an integer that the spider should know how many links should be extract from all pages;
6. **inputOk**, a Boolean indicates that the user inputs are fine;
7. **URLPool**, a linked list that stores the urls extracted from current page;
8. **Keywords**, a linked list that stores the keywords of current page;
9. **KeywordNodes**, a linked list that stores the the keywords as well as the number of keywords in current page;
10. **spiderEggs**, candidate spiders to be produced.

A spider will do two things basically, to find the hyperlinks and keywords in its page. The spider will request the HTML file from its stored url and read the HTML file line by line to extract useful information.

## Extract keywords

In this program we mainly focus on special tags that we think are the most possible ones that have meaningful contents to extract keywords in order to save time. So when the program read a new line, it will check if this line contains ant of these tags we defined in string array KeywordTag. The spider will just ignore those lines which do not contain KeywordTag. Then a string array will store the candidate key words by splitting the line by space. Each candidate keyword will be sent to isAWord method of WordChecker class which looks up in an English word dictionary database called Wordnet. If the candidate key words is real English word and is not an element in StopList that is a string array stores meaningless English words, we will deal with it. If it's the first time we meet this word we will add the word into Keywords and create a new keywordNode for this keyword. If this keyword has been added before we will increment the number of its appearance by 1 in keywordNode. It's also possible that the keyword is an English phrase consists of two words that are not English words. For example, Hong Kong is an English word but Hong and Kong are not. So the spider will check the current word if it's not the first one and the one before it combining together to see if the phrase is an English phrase and will be deal the same way as single English word.

## Extract Hyperlinks