Team Member:
Sicheng Jiang - sj62 (group leader)
Zhaokuan Chen - zc56
David Liu - ztliu2

System Chosen: ExpertSearch System
Subtopic: 3.2 Automatically crawling faculty webpages - Identifying faculty directory pages

Briefly describe any datasets, algorithms or techniques you plan to use

a. We plan to use the web page provided in the project instruction as a dataset to scrap. The algorithm or technique will be breadth first search. We will search for all faculty directory first, and store all the links from these directories. Then we will go through each of these links one by one to identify if the linked URL is another directory page.

If you are adding a function, how will you demonstrate that it works as expected? If you are improving a function, how will you show your implementation actually works better?

b. We will implement our solution by Python version 3.7 running under Docker. This will ensure the development environment is consistent throughout our entire team. If we are adding a function, we will demonstrate this by running unit tests under the same docker environment. If we are improving a function, we will show our implementation actually works better by comparing the result with the previous version. The version control is implemented by Git.

How will your code communicate with or utilize the system? It is also fine to build your own systems, just please state your plan clearly

c. The system we are communicating to are web pages of faculty directories. Our code communicates with the system by HTTP protocol. We will mainly use GET requests to retrieve the parseable HTML from the faculty directory URL, then parse the retrieved HTML locally on Docker containers running on our computers.

Which programming language do you plan to use?

d. Python 3.7
e. Docker
f. Git

Please justify that the workload of your topic is at least 20*N hours, N being the total number of students in your team. You may list the main tasks to be completed, and the estimated time cost for each task.

g. research and learn from tutorial (5 hr)
h. explore web page structure (5 hr)
i. learn text pattern to scrap for faculty directory and faculty webpages (5 hr)
j. Setup a consistent environment for each team member (5 hr)
k. Implement the program (30 hr)
l. Verify that the program works as expected (10 hr)