MobiTV Git Sort Go Implementation

Yong Chong

Overview:

I. Parsing the input

A. First I had to figure out how to read in the input using flags, and then find out how to parse the XML files.

B. Once I figured this out I needed a way to actually utilize the fields I extracted from the XML file. I decided the best way to go about organizing the projects into their respective folders/priority queues was to create a map data structure that would keep track of all of the current folder names that had been read, and to initialize a folder if it had not yet been created. This map would have keys that were represented by strings (folderName) and values that would be priority queues. I created a for loop to iterate through the XML field “Filename” for each of the projects, and would sort each project into their respective “key” in the map which was represented by the folder name (i.e. folder1, folder2…).

C. Finally I used another for loop after the first one in order to iterate through each folder in the map. Inside this for loop I had another nested for loop that would iterate through the priority queue for that folder in order of priority. I used waitgroups to ensure that goroutine’s would finish executing before the program returned.

II. Implementing Priority Queue

A. PriorityQueue was implemented given the github package that was provided

III. Cloning the git repositories

A. I did research on how to clone git repositories using SSH keys but found that it required generated credentials. I worked around this by converting the url field from the XML file to HTTPS versions. I then used the os/exec package to execute the local-host git commands of cloning, making sure to capture any failure/error outputs. In the case that there was a specific commit SHA to use I used exec to once again reset to that specific commit.

Sources:

Parsing XML:

1. https://www.youtube.com/watch?v=-PATP8IZq5A
2. https://golang.org/pkg/encoding/json/#Unmarshal

Go Routines:

3. https://golangbot.com/goroutines/

PriorityQueues:

4. https://www.geeksforgeeks.org/binary-heap/
5. https://www.geeksforgeeks.org/priority-queue-set-1-introduction/
6. https://golang.org/pkg/container/heap/
7. https://golang.org/src/container/heap/heap.go?s=1328:1480#L22
8. https://github.com/bgadrian/data-structures/blob/master/priorityqueue/README.md

Misc:

9. https://www.geeksforgeeks.org/what-is-blank-identifierunderscore-in-golang/
10. https://www.linuxjournal.com/content/back-github-and-gitlab-repositories-using-golang
11. https://github.com/src-d/go-git/
12. https://stackoverflow.com/questions/6182369/exec-a-shell-command-in-go
13. https://golang.org/pkg/strconv/
14. https://medium.com/@gianbiondi/interfaces-in-go-59c3dc9c2d98