Pelinsu Saraç - 28820

CS 403 Distributed Systems PA2 Report

Abstract MapReduce Class:

- Initializer of the class sets the number of workers that will be used, along with some barriers that helps with keeping the order of opening and closing socket connections.
- Producer method receives the file to be processed in list format and shares this file equally among the workers. It does an integer division to find the rough size of the packets, but also checks the modulo to see how many left after division. It adds one line from file per worker while sending the packets until there is no remaining left. That's how the file is shared almost equally. Note that for the sake of simplicity, a single line in file is represented as two integers in the list. The producer opens a new connection for each consumer and sends one packet only through these sockets.
- Consumers start the pull socket that corresponds to their order. They know which port to use via the arguments that start method sends while creating the processes. They receive the packet that the producer sent and then process it via the Map method. Then they send the processed packet through a new socket to the collector. For this purpose, each consumer uses a different port from each other and the from the port they used to receive the packets.
- Result Collector opens different sockets with different ports that consumers use to send the partial results. It gathers the partial results and calls the Reduce method on them. Finally, it writes the result to the "results.txt" file.
- Barriers used for these methods are to be sure that the binding sockets are created before connecting sockets or just to be sure that the socket remains open until the message is received.
- Start method reads the file to be processed and creates and Array object out of it. For the sake of simplicity, each line is stored as two integers in the Array object, thus two elements construct one line. Then it starts the necessary processes and waits for them to terminate.
- Protected scope of the methods are assured with "_" in front of the method names.

FindCitations Concreate Class:

- Map implementation basically counts every second element in the packet, as they are the cited papers.
- Reduce implementation basically goes through each partial result and adds the values of same keys in a separate dictionary.

FindCyclicReferences Concreate Class:

- Map implementation process the list two by two. It creates a key out of the nodes in the asked order, from smaller to larger. Then it checks whether this key was created and stored in the dictionary before. If so, it means cycle and the value is updated as 1. If not, it still stores the edge to the partial result dictionary as the other edge may be in different packet.
- Reduce goes over all results of partial result dictionaries. If a cycle is already detected at partial, it reflects it exactly to the final dictionary. Otherwise, it checks whether one edge is on one packet and other is on another, by basically storing the keys and checking one by one. If it catches one, it updates the value of the key to 1. Finally, it removes any other edge that has not appeared in a cycle.

Main: It basically accepts the arguments, checks them, create the instance of appropriate class and starts the whole flow.