1 Objective

This is the continuation of Homework 5. The purpose of this task is to find unique set of extracted sub-strings.

2 Assignment

In homework 5, children (slaves) fetched random data from the master. The slaves then extracted all sub-strings matching the pattern. The task for homework 6 is to send back a unique set of extracted sub-strings to the master. The master then collects those sub-strings and returns as the final result. The final result is one list containing all sub-strings.

This task can be done in several ways. An easy way to explain how to do the task is by using a task-dependency graph. For this reason, we have provided two such possible solutions using a task-dependency graph that you can follow to solve this assignment. You can come up with your own ways to solve this problem. But in doing so, you need to provide the task-dependency graph for your solution during the submission of homework.

For the reference, data generator will be based on the following script [1]
res = ''.join(random.choice(string.ascii_uppercase + string.digits)
             for i in range(20))

Listing 1: Random String Generation, please note the parameter changed now from 180 to 20 in this case!

In what follows, both solutions will be discussed;

3 Possible Solution I

Assumed that master P0 being in communication with 4 children (slaves) like P1, P2, P3, P4. (Homework 5) After the request from slaves, all children fetched arbitrary data from master. For this purpose the lists of sub-strings need to be cross-checked through different processes as well. All sub-strings need to be unique, for example children (slaves) supposed to perform communication from P1 with P2 and P3 with P4. Then, P1 communicate with P3 for further unique extraction of sub-strings. Finally, P1 gets sent unique data from all the children (slaves) and responds to master as shown in figure 1. Hint: For better performance of uniqueness checks you can sort the extracted lists of sub-strings. As a result, master received all the data in the merged form.

4 Possible Solution II

Round-robin technique (ring communication) is another one possible solution shown in figure 2 (discussed at last lecture as well). After the request from slaves, all children fetched arbitrary data from master. For this purpose all the lists of sub-strings need to be cross-checked on slaves P1...P4. The lists are moved around step by step in a ring communication pattern of 3 steps, each containing corresponding uniqueness checks. Now all the children (slaves) have unique set of sub-strings, which they send to master on individual basis.
Figure 1: Task-Dependency Graph

Figure 2: Task-Dependency Graph