YASİN ATLI - 2020400246
ENES BAŞER – 2020400141
Group : 49
CMPE 300

PROJECT 2

REPORT OF THE PROJECT

First we begin with hw.py file. It is the main file.We take input and output file name as argv. We initialize variables total_machine,clock_cycle,,threshold integers and operation cost ,and machines lists. In our program, each machine is a machine object that constructed in machine.py file. Machine object has product,id,parent_id, isLeaf, child_machines, operation, and a_cost attributes and add_child function to add its child machine to its child_machines list.

Then, we create our terminal machine with id=1 and append it the machines list. After that, we read the input file and setting total_machine as first row's int value, clock_cycle as second's. At the third line, we get all the elements in a list for using later in the worker.py. The fourt line stands for threshold value.

The other lines in the input file stand for machines and first products. We split the rest of the lines and if a line's length is 3 we see that it is a machine initialization and we create machine object and add it to the machines list. If line's length is not 3 we conclude that it is a product. Then we set the leaf machines product as the line's sting value.

We open the output file and create MPI.COMM with spawning processes as number of machines.Then we create a nested for loop for clock cycle and every machine. For every process, we send log string, operation cost's list, clock cycle, machine object, threshold value.

The worker.py file is where the communication between machines and machine operations happen. We define the operation functions. In the main body, we create the MPI.COMM.WORLD for the communication between machines. We receive the data send from the hw.py file.

For every machine and clock cycle, we split the machines into 3 categories. 1 is if machine is leaf machine, 2 is if machine is terminal machine and the 3 is for the machines between leaf and terminal machine.

YASİN ATLI - 2020400246
ENES BAŞER – 2020400141
Group : 49

If machine is leaf machine, its product is setted and we directly perform its operation. After, we perform the wear factor and check for exceeding threshold (if so we enter the log string and do maintanence) and change the operation.Then we send its product to its parent machine.

If machine is a machine between leaf and terminal, its product is not setted and we wait to receive all the products from its child machines.After we get the product,we perform its operation. Then, we perform the wear factor and check for exceeding threshold (if so we enter the log string and do maintanence) and change the operation.Last, we send its product to its parent machine.

If machine is the terminal, its product is not setted and we wait to receive all the products from its child machines.After we get the product,we perform add operation because terminal performs only add operation.Last, we send its product to masters process in hw.py.

For every machine in every clock cycle, we send its log string to the master process in hw.py. If there is no log, it will send empty string. However, if there exist a log it will send the lof and it will be written in the output file.

Later that, for every clock cycle we receive a final product from terminal machine with specific tag.Then we write it into output file. Lastly, we receive the logs for every clock cycle and every machine. If there exists any logs, we write them into the output file.

YASİN ATLI - 2020400246
ENES BAŞER – 2020400141
Group : 49

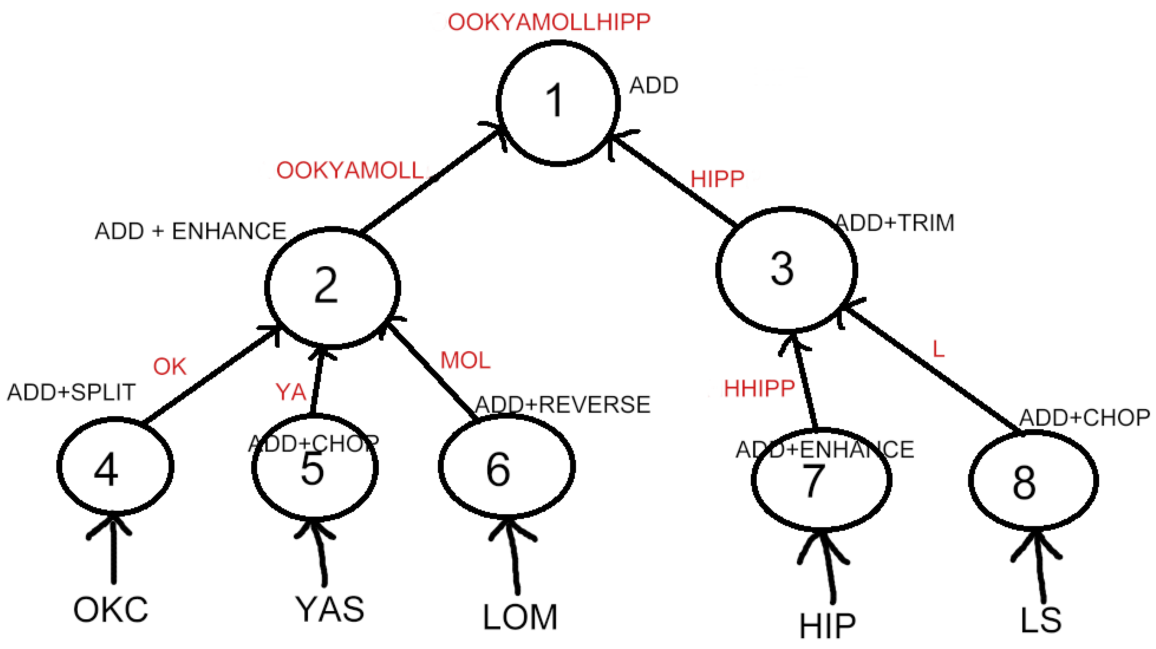
For given input :

```
8
10
7 6 5 4 3
30
2 1 enhance
3 1 trim
4 2 split
5 2 chop
6 2 reverse
7 3 enhance
8 3 chop
OKC
YAS
LOM
HIP
LS
```

The first line represents the number of machines. In this example, there are 8 machines. The second line represents number of production cycles. The third line denotes wear factor for each operation, in the order of enhance (7), reverse (6), chop (5), trim (4), split (3). The fourth line is the threshold value for maintenance. The subsequent 7 lines, the number of machines minus 1, is an adjacency list of machines. In each line, the first number is a child machine's id, and the next number is its parent machine's id. Since this is a tree structure, and each machine has only one parent except the terminal machine, the number of edges (lines) is equal to the number of machines minus 1. The initial operation for a child machine is given next (Machine 2 starts with enhance, machine 3 starts with trim...). Lastly, the subsequent lines are products received by leaf machines. In this example, machines with id's 4,5, 6, 7,8 do not have any other machine preceding them (meaning that they are leaf machines). Therefore, machine 4 processes product "OKC", machine 5 processes product "YAS" ,machine 6 processes product "LOM", machine 7 processes product "HIP", and machine 8 processes product "LS". These products must be assigned to leaf machines by the order of their id's, not in the order of their appearance in the adjacency list.

The figure demonstrates the first production cycle for the example input. For instance, Machine 3 has Machine 7 and Machine 8 as its children, and has Machine 1 as its parent. Machine 3 receives "HHIPP" from Machine 7 and "L" from machine 8. First it adds them

YASİN ATLI - 2020400246
ENES BAŞER – 2020400141
Group : 49
and produces "HHIPPL". Then it performs trim operation and produces "HIPP" as its output. Then it passes "HIPP" to Machine 1. Machine 2 receives "OK" from Machine 4 and "YA" from machine 5 and "MOL" from machine 6. First it adds them and produces "OKYAMOL". Then it performs enhance operation and produces "OOKYAMOLL" as its output. Then it passes "OOKYAMOLL" to Machine 1.Last, Machine 1 receives "OOKYAMOLL" from Machine 2 and "HIPP" from machine 3. It adds them and produces "OOKYAMOLLHIPP".



YASİN ATLI - 2020400246
ENES BAŞER – 2020400141
Group : 49
BONUS:

If this was a real Industry 4.0 project aiming to implement a digital twin, what would be the challenges for the implementation?

In real Industry 4.0, In real-life Industry 4.0 scenarios, due to the fact that machines do not operate flawlessly, the products they produce may have defects. These defects can create a chain reaction, impacting other machines and ultimately leading to defective final products. This chain of defects results in time and raw material wastage. Additionally, as machines operate, they undergo wear and tear, requiring maintenance. Unlike in code, where maintenance can be done instantly, in the real world, allocating time for maintenance temporarily halts production. This interruption affects the operation of other machines, having a negative impact on overall production efficiency.