Assignment 3 UNDO Logging and Recovery

Note: Strictly follow input/output formats. Make sure your code outputs only what is asked, i.e. no extra outputs such as debug prints. Deadline is 11:55pm, 7th **November 2019. No deadline extensions under any circumstances.**

Languages Allowed: Python only

Part 1: UNDO Logging

Write `UNDO` logs into a file for a given set of transactions. Do not worry about concurrency on transactions, there will be inconsistencies.

Input File Format

The first line of the file will be a list of database element names and their initial values, space separated, on a single line.
Each transaction will begin on a new line, with the transaction name and number of actions in the first line, followed by actions of form `READ(), WRITE(), OUTPUT()`, or an operation in successive lines. Successive transactions are separated by a newline character. For example:

```
A 4 B 4 D 5

T1 4
READ(A, t)
t := t+2
WRITE(A, t)
OUTPUT(A)

T2 5
READ(A, t)
t := t+2
t := t-4
WRITE(A, t)
OUTPUT(A)
```

The transactions are assumed to be executed in a Round-Robin(`RR`) fashion. For this another additional command line argument is provided '`x`'. Given `n` transactions, carry out first `x` instructions/actions of the first transaction, then the first `x` instructions/actions of the second transaction and so on ...

The set of operations you'll have to handle are $\{+, -, *, /\}$ and the second operand is always an integer.

Assignment 3 UNDO Logging and Recovery

Output File Format

If your roll number is 2016XXXX then your python file name for the first part of the assignment should be 2016XXXX_1.py and the output file name should be 2016XXXX_1.txt

Therefore, an example of the command line arguments is as follows

```
python 2016XXXX_1.py <input file name> <X>
ex:  python 20160000_1.py input.txt 5
```

Task: UNDO Logs

Write UNDO logs for the set of transactions. In addition, after every log record also print the values of the variables in both the main memory and the disk corresponding to the state after the current log record. The variables should be in Lexicographic order.

Suppose the input file had contents as previously specified in the input file format section and the value of x is 1, the contents of the output file would look like this

```
<START T1>

A 4 B 4 D 5
<START T2>
A 4
A 4 B 4 D 5
<T1, A, 4>
A 8
A 4 B 4 D 5
<COMMIT T1>
A 8
A 8 B 4 D 5
<T2, A, 8>
A 4
A 8 B 4 D 5
<COMMIT T2>
A 4
A 4 B 4 D 5
```

where the first line after a log record is the contents of the main memory and the second line is the contents of the disk. The values after the START log correspond to values of variables right before the first action

Note: Please note that if the variable has already been read from the disk into the main memory, another READ() command will not result in another read operation. The contents

of the main memory will be used. Additionally, if the variable is not in the main memory `INPUT()` will implicitly be called by `READ()`

Part 2: Undo Recovery

Given an input file containing `UNDO` logs till a crash point, and the current set of database element values, perform a recovery - output the set of database elements and their recovered values.

Input File Format

The first line of the file will be a list of database element names and their current disk values, space separated, on a single line.
This is followed by a number of log statements which are either `START`s, update logs, `COMMIT`s, Nonquiescent `START CKPT`s or `END CKPT`s in successive lines. For formats, check the example:

```
A 4 B 4 D 5

<START T1>
<START T3>
<T1, A, 8>
<START CKPT (T1, T3)>
<START T2>
<COMMIT T1>
<T3, D, 10>
<COMMIT T3>
<END CKPT>
```

And so on. The last log entry in the file is the entry just before the crash happened.

Output File Format

If your roll number is `2016XXXX` then your `python` file name for the second part of the assignment should be `2016XXXX_2.py` and the output file name should be `2016XXXX_2.txt`
Therefore, an example of the command line arguments is as follows

```
python 2016XXXX_2.py <input file name>
ex:  python 20160000_2.py input.txt
```

Assignment 3 UNDO Logging and Recovery

Task: UNDO Recovery

The task is to output a single line containing the list of database elements and their values after recovery, space-separated, in lexicographic order. For example:

A 4 B 4 D 5

Submission Format

- Code from part 1 named as `<RollNo>_1.py` and part 2 as `<RollNo>_2.py`
- A bash script named `<RollNo>.sh`, which, if given two command-line arguments runs logging (The two arguments being input file and `x`), and if given only one argument, runs recovery, and produces the output.
- Zip the three files - `<RollNo>_1.py`, `<RollNo>_2.py`, `<RollNo>.sh` into `<RollNo>.zip`. Upload the zip.

Plagiarism of any kind will not be tolerated. It may result in 0 marks for the assignment or even an F in the course.