This will simulate a 32KB write-back cache with 8-way associativity and FIFO replacement policy. The memory trace will be read from `/home/TRACES/MCF.t`.

Note: the trace file will contain addresses that can be for 64-bit system, so you might need data types that are large enough to read them correctly and bookkeep the metadata in your simulator. For example, if the tag is 9 bytes and you allocate your tag array bookkeeping array as an array of integers, you will not be able to store the whole 9 bytes; integer is only 4 bytes. Accordingly, use data types such as long long int and its equivalents in other languages.

4 Output from the Simulator

The following outputs are expected from your simulator:

a. **Total miss ratio for L1 cache**

b. **The # writes to memory**

c. **The # reads from memory**

It is recommended to ensure that your output formatting is clean, structured, and easy for readers to understand.

5 Testing your code

There are two samples of collected addresses along with the output simulation results found in the provided project files. These resources aim to facilitate verification of the correct functionality of your program, potentially streamlining the debugging process.

The files `smallTest.t` and `mediumTest.t` contain a total of 1,000 and 10,000 address traces, respectively. Both files are bundled in a zip archive named `partialResults.zip`. The commands below can be utilized to verify program correctness:

For the simulation using the file `smallTest.t`, containing 1,000 collected addresses:

```
./SIM 32768 8 0 0 smallTest.t
```

Expected output:

```
Miss ratio 0.143000
write 392
read 143
```

For the simulation using the file `mediumTest.t`, containing 10,000 collected addresses:

```
./SIM 32768 8 0 0 mediumTest.t
```

Expected output:

```
Miss ratio 0.112800
write 2537
read 1128
```

Again, with the file `mediumTest.t`:

```
./SIM 32768 8 0 1 mediumTest.t
```

Expected output:

```
Miss ratio 0.112800
write 34
read 1128
```

And once more with the file `mediumTest.t`:

```
./SIM 32768 8 1 0 mediumTest.t
```

Expected output:

```
Miss ratio 0.117800
write 2537
read 1178
```

Final Validation Runs:

```
./SIM 32768 8 0 1 /home/TRACES/XSBENCH.t
```

Expected output:

```
0.112008
44.000000
2371758.000000
```

And `./SIM 32768 8 0 0 /home/TRACES/XSBENCH.t`

Expected output:

```
0.112008
5013495.000000
2371758.000000
```

6 Grading Policy

1. (20 points) A functional code that runs cleanly, i.e., showing serious efforts to implement the simulator.
2. (40 points) Your code simulates correctly the sample runs we provide and all output statistics should be within less than 0.001% error of our results. 2 tests will be provided along with the configurations and the expected output. Two tests will not be provided to you and are used to make sure your simulator isn't tuned for our sample tests. Each test will weigh 10%.