Project 2: Memory Hierarchy Simulator Report

Adham Ali - 900223243
Ebram Thabet - 900214496
Khadeejah Iraky - 900222731
Submitted to Dr. Cherif Salama

CSCE2303, Section 1
Computer Science and Engineering Department
The American University in Cairo

Our Implementation

We created a function called cache simulation that implements the project, while the main asks for user inputs and sends them to the function. The function also displays the output by creating an output.txt file where it displays the output of all the cache lines and the data of hit ratio, miss ratio, AMAT, and total accesses at the end.

Assumptions

- All data/memory address files must be submitted such that each memory address is on a line with no commas.
- The output file has the index number in decimal format and the tag in decimal format.
- The output file gives the hit ratio, miss ratio, AMAT, and total accesses at the end and not at each access.
- We assumed that the tag is calculated by getting the floor of the address divided by the cache-line size, and that the index is the tag modulo the number of cache lines.
- Memory given will be equal to or smaller than 2^30.

Bugs and Issues

- The handling for if the memory address is greater than 2^30 is not great as it only breaks.

Sequence 1

0
28
384
1107
2523
3010
1721
3294
4082
1234
94857632
65284096

```
48957696
42177280
84356096
6870976
73742848
8957696
47483648
47483650
```

Inputs:

- Cache size = 4096
- Cache line size = 32
- Time = 5

Output 1

```
Valid bits and tags:
Index 0 - Valid: 1, Tag: 0
Index 1 - Valid: 0, Tag: 0
Index 2 - Valid: 0, Tag: 0
Index 3 - Valid: 0, Tag: 0
Index 4 - Valid: 0, Tag: 0
Index 5 - Valid: 0, Tag: 0
Index 6 - Valid: 0, Tag: 0
Index 7 - Valid: 0, Tag: 0
Index 8 - Valid: 0, Tag: 0
Index 9 - Valid: 0, Tag: 0
Index 10 - Valid: 0, Tag: 0
Index 11 - Valid: 0, Tag: 0
Index 12 - Valid: 1, Tag: 0
Index 13 - Valid: 0, Tag: 0
Index 14 - Valid: 0, Tag: 0
Index 15 - Valid: 0, Tag: 0
Index 16 - Valid: 0, Tag: 0
Index 17 - Valid: 0, Tag: 0
Index 18 - Valid: 0, Tag: 0
Index 19 - Valid: 0, Tag: 0
Index 20 - Valid: 0, Tag: 0
Index 21 - Valid: 0, Tag: 0
```

```
Index 22 - Valid: 0, Tag: 0
Index 23 - Valid: 0, Tag: 0
Index 24 - Valid: 1, Tag: 10297
Index 25 - Valid: 0, Tag: 0
Index 26 - Valid: 0, Tag: 0
Index 27 - Valid: 0, Tag: 0
Index 28 - Valid: 0, Tag: 0
Index 29 - Valid: 0, Tag: 0
Index 30 - Valid: 0, Tag: 0
Index 31 - Valid: 0, Tag: 0
Index 32 - Valid: 0, Tag: 0
Index 33 - Valid: 0, Tag: 0
Index 34 - Valid: 1, Tag: 0
Index 35 - Valid: 0, Tag: 0
Index 36 - Valid: 0, Tag: 0
Index 37 - Valid: 0, Tag: 0
Index 38 - Valid: 1, Tag: 0
Index 39 - Valid: 0, Tag: 0
Index 40 - Valid: 0, Tag: 0
Index 41 - Valid: 0, Tag: 0
Index 42 - Valid: 0, Tag: 0
Index 43 - Valid: 0, Tag: 0
Index 44 - Valid: 0, Tag: 0
Index 45 - Valid: 0, Tag: 0
Index 46 - Valid: 0, Tag: 0
Index 47 - Valid: 0, Tag: 0
Index 48 - Valid: 0, Tag: 0
Index 49 - Valid: 0, Tag: 0
Index 50 - Valid: 0, Tag: 0
Index 51 - Valid: 0, Tag: 0
Index 52 - Valid: 0, Tag: 0
Index 53 - Valid: 1, Tag: 0
Index 54 - Valid: 0, Tag: 0
Index 55 - Valid: 0, Tag: 0
Index 56 - Valid: 0, Tag: 0
Index 57 - Valid: 0, Tag: 0
Index 58 - Valid: 0, Tag: 0
Index 59 - Valid: 0, Tag: 0
Index 60 - Valid: 0, Tag: 0
```

```
Index 61 - Valid: 0, Tag: 0
Index 62 - Valid: 1, Tag: 1677
Index 63 - Valid: 0, Tag: 0
Index 64 - Valid: 1, Tag: 15938
Index 65 - Valid: 0, Tag: 0
Index 66 - Valid: 0, Tag: 0
Index 67 - Valid: 0, Tag: 0
Index 68 - Valid: 0, Tag: 0
Index 69 - Valid: 0, Tag: 0
Index 70 - Valid: 0, Tag: 0
Index 71 - Valid: 0, Tag: 0
Index 72 - Valid: 1, Tag: 11952
Index 73 - Valid: 0, Tag: 0
Index 74 - Valid: 0, Tag: 0
Index 75 - Valid: 0, Tag: 0
Index 76 - Valid: 0, Tag: 0
Index 77 - Valid: 1, Tag: 23158
Index 78 - Valid: 1, Tag: 0
Index 79 - Valid: 0, Tag: 0
Index 80 - Valid: 1, Tag: 18003
Index 81 - Valid: 0, Tag: 0
Index 82 - Valid: 0, Tag: 0
Index 83 - Valid: 0, Tag: 0
Index 84 - Valid: 0, Tag: 0
Index 85 - Valid: 0, Tag: 0
Index 86 - Valid: 0, Tag: 0
Index 87 - Valid: 0, Tag: 0
Index 88 - Valid: 1, Tag: 11592
Index 89 - Valid: 0, Tag: 0
Index 90 - Valid: 0, Tag: 0
Index 91 - Valid: 0, Tag: 0
Index 92 - Valid: 0, Tag: 0
Index 93 - Valid: 0, Tag: 0
Index 94 - Valid: 1, Tag: 0
Index 95 - Valid: 0, Tag: 0
Index 96 - Valid: 1, Tag: 20594
Index 97 - Valid: 0, Tag: 0
Index 98 - Valid: 0, Tag: 0
Index 99 - Valid: 0, Tag: 0
```

```
Index 100 - Valid: 0, Tag: 0
Index 101 - Valid: 0, Tag: 0
Index 102 - Valid: 1, Tag: 0
Index 103 - Valid: 0, Tag: 0
Index 104 - Valid: 0, Tag: 0
Index 105 - Valid: 0, Tag: 0
Index 106 - Valid: 0, Tag: 0
Index 107 - Valid: 0, Tag: 0
Index 108 - Valid: 0, Tag: 0
Index 109 - Valid: 0, Tag: 0
Index 110 - Valid: 0, Tag: 0
Index 111 - Valid: 0, Tag: 0
Index 112 - Valid: 0, Tag: 0
Index 113 - Valid: 0, Tag: 0
Index 114 - Valid: 0, Tag: 0
Index 115 - Valid: 0, Tag: 0
Index 116 - Valid: 0, Tag: 0
Index 117 - Valid: 0, Tag: 0
Index 118 - Valid: 0, Tag: 0
Index 119 - Valid: 0, Tag: 0
Index 120 - Valid: 1, Tag: 2186
Index 121 - Valid: 0, Tag: 0
Index 122 - Valid: 0, Tag: 0
Index 123 - Valid: 0, Tag: 0
Index 124 - Valid: 0, Tag: 0
Index 125 - Valid: 0, Tag: 0
Index 126 - Valid: 0, Tag: 0
Index 127 - Valid: 1, Tag: 0

Total number of accesses: 20
Hit ratio: 0.1
Miss ratio: 0.9
AMAT: 140 cycles
```

Sequence 2

```
4032
986
2598
3478
```

```
2045
1756
1423
65
66
2939
44033
4987
52599
773479
2046
1757
1424
66
67
2938
```

Inputs:

- Cache size = 4096
- Cache line size = 32
- Time = 5

Output 2

```
Valid bits and tags:
Index 0 - Valid: 0, Tag: 0
Index 1 - Valid: 0, Tag: 0
Index 2 - Valid: 1, Tag: 0
Index 3 - Valid: 0, Tag: 0
Index 4 - Valid: 0, Tag: 0
Index 5 - Valid: 0, Tag: 0
Index 6 - Valid: 0, Tag: 0
Index 7 - Valid: 0, Tag: 0
Index 8 - Valid: 0, Tag: 0
Index 9 - Valid: 0, Tag: 0
Index 10 - Valid: 0, Tag: 0
Index 11 - Valid: 0, Tag: 0
Index 12 - Valid: 0, Tag: 0
Index 13 - Valid: 0, Tag: 0
```

```
Index 14 - Valid: 0, Tag: 0
Index 15 - Valid: 0, Tag: 0
Index 16 - Valid: 0, Tag: 0
Index 17 - Valid: 0, Tag: 0
Index 18 - Valid: 0, Tag: 0
Index 19 - Valid: 0, Tag: 0
Index 20 - Valid: 0, Tag: 0
Index 21 - Valid: 0, Tag: 0
Index 22 - Valid: 0, Tag: 0
Index 23 - Valid: 0, Tag: 0
Index 24 - Valid: 0, Tag: 0
Index 25 - Valid: 0, Tag: 0
Index 26 - Valid: 0, Tag: 0
Index 27 - Valid: 1, Tag: 1
Index 28 - Valid: 0, Tag: 0
Index 29 - Valid: 0, Tag: 0
Index 30 - Valid: 1, Tag: 0
Index 31 - Valid: 0, Tag: 0
Index 32 - Valid: 0, Tag: 0
Index 33 - Valid: 0, Tag: 0
Index 34 - Valid: 0, Tag: 0
Index 35 - Valid: 0, Tag: 0
Index 36 - Valid: 0, Tag: 0
Index 37 - Valid: 0, Tag: 0
Index 38 - Valid: 0, Tag: 0
Index 39 - Valid: 0, Tag: 0
Index 40 - Valid: 0, Tag: 0
Index 41 - Valid: 0, Tag: 0
Index 42 - Valid: 0, Tag: 0
Index 43 - Valid: 0, Tag: 0
Index 44 - Valid: 1, Tag: 0
Index 45 - Valid: 0, Tag: 0
Index 46 - Valid: 0, Tag: 0
Index 47 - Valid: 0, Tag: 0
Index 48 - Valid: 0, Tag: 0
Index 49 - Valid: 0, Tag: 0
Index 50 - Valid: 0, Tag: 0
Index 51 - Valid: 0, Tag: 0
Index 52 - Valid: 0, Tag: 0
```

```
Index 53 - Valid: 0, Tag: 0
Index 54 - Valid: 1, Tag: 0
Index 55 - Valid: 0, Tag: 0
Index 56 - Valid: 0, Tag: 0
Index 57 - Valid: 0, Tag: 0
Index 58 - Valid: 0, Tag: 0
Index 59 - Valid: 0, Tag: 0
Index 60 - Valid: 0, Tag: 0
Index 61 - Valid: 0, Tag: 0
Index 62 - Valid: 0, Tag: 0
Index 63 - Valid: 1, Tag: 0
Index 64 - Valid: 0, Tag: 0
Index 65 - Valid: 0, Tag: 0
Index 66 - Valid: 0, Tag: 0
Index 67 - Valid: 0, Tag: 0
Index 68 - Valid: 0, Tag: 0
Index 69 - Valid: 0, Tag: 0
Index 70 - Valid: 0, Tag: 0
Index 71 - Valid: 0, Tag: 0
Index 72 - Valid: 0, Tag: 0
Index 73 - Valid: 0, Tag: 0
Index 74 - Valid: 0, Tag: 0
Index 75 - Valid: 0, Tag: 0
Index 76 - Valid: 0, Tag: 0
Index 77 - Valid: 0, Tag: 0
Index 78 - Valid: 0, Tag: 0
Index 79 - Valid: 0, Tag: 0
Index 80 - Valid: 0, Tag: 0
Index 81 - Valid: 1, Tag: 0
Index 82 - Valid: 0, Tag: 0
Index 83 - Valid: 0, Tag: 0
Index 84 - Valid: 0, Tag: 0
Index 85 - Valid: 0, Tag: 0
Index 86 - Valid: 0, Tag: 0
Index 87 - Valid: 0, Tag: 0
Index 88 - Valid: 0, Tag: 0
Index 89 - Valid: 0, Tag: 0
Index 90 - Valid: 0, Tag: 0
Index 91 - Valid: 1, Tag: 0
```

```
Index 92 - Valid: 0, Tag: 0
Index 93 - Valid: 0, Tag: 0
Index 94 - Valid: 0, Tag: 0
Index 95 - Valid: 0, Tag: 0
Index 96 - Valid: 1, Tag: 10
Index 97 - Valid: 0, Tag: 0
Index 98 - Valid: 0, Tag: 0
Index 99 - Valid: 0, Tag: 0
Index 100 - Valid: 0, Tag: 0
Index 101 - Valid: 0, Tag: 0
Index 102 - Valid: 0, Tag: 0
Index 103 - Valid: 0, Tag: 0
Index 104 - Valid: 0, Tag: 0
Index 105 - Valid: 0, Tag: 0
Index 106 - Valid: 0, Tag: 0
Index 107 - Valid: 1, Tag: 188
Index 108 - Valid: 1, Tag: 0
Index 109 - Valid: 0, Tag: 0
Index 110 - Valid: 0, Tag: 0
Index 111 - Valid: 0, Tag: 0
Index 112 - Valid: 0, Tag: 0
Index 113 - Valid: 0, Tag: 0
Index 114 - Valid: 0, Tag: 0
Index 115 - Valid: 0, Tag: 0
Index 116 - Valid: 0, Tag: 0
Index 117 - Valid: 0, Tag: 0
Index 118 - Valid: 0, Tag: 0
Index 119 - Valid: 0, Tag: 0
Index 120 - Valid: 0, Tag: 0
Index 121 - Valid: 0, Tag: 0
Index 122 - Valid: 0, Tag: 0
Index 123 - Valid: 0, Tag: 0
Index 124 - Valid: 0, Tag: 0
Index 125 - Valid: 0, Tag: 0
Index 126 - Valid: 1, Tag: 0
Index 127 - Valid: 0, Tag: 0

Total number of accesses: 20
Hit ratio: 0.35
```

Miss ratio: 0.65
AMAT: 102.5 cycles

Bonus Features

- Supporting separate caches for data and instructions; it was implemented by requiring the user to give two separate files one for data and one for instructions, and then the program outputs the results of the data cache-ing and then the instruction cache-ing.
- Supporting multi-level cache, as specified by the user; the user needs to input the cache size (S), the cache line size (L), and the number of clock cycles needed to access the cache (Time) for each cache, and is then given at the end the hit and miss ratio for each cache alongside the AMAT.

Step-by-Step Guide

1. You will need to fill a file called “AccesssSeq.txt” with the memory address sequence. (in case you will choose data and instructions you need to create two files one named “DataAccesssSeq.txt” and “InstAccesssSeq.txt”.
2. You will be requested to enter how many caches you wish to have.
3. If you answer “1”, you will have the option to have an instruction cache and data cache.
4. If you choose to forgo the option, you will only have a data cache.
5. In all of the above situations you will need to enter the following data for each cache.
    a. Cache size (S)
    b. Cache line size (L)
    c. Number of clock cycles to access the cache (Time)

```
Enter the number of Cache levels you want:1

For Level 1:
Enter cache size (S):4096

Enter cache line size (L):32

Enter access time (Time):5
 Do you want instruction and data cache? (1 yes, 0 no)0

Process finished with exit code 0
```

6. After which the code will run and the output will appear in the output.txt