CS 312

Lab 3 – Cache Calculator

Description

Develop a Cache calculator in the **Java** programming language, which specifies the three different cache mapping of direct, associative, and set-associative.

The input of the program is the following:

- ***Main memory*** in the format $A \times 2^E$, where *A* and *E* are the user inputs
- ***Cache blocks*** as a base-2 value
- ***Bytes*** in a Cache block as a base-2 value
- ***k***-way set associative value where *k* is a base-2 value (e.g., 2, 4)
- ***Memory address*** in base-16

The program should do the following:

- Calculate and display the ***Main memory*** in bytes.
- Convert and display the ***Memory address*** in base-2.
- Direct Mapping
    - Calculate the ***Tag***, ***Line*** and ***Word*** bit sizes
    - Display the binary ***Memory address*** in the ***Tag***, ***Line*** and ***Word*** bit fields
- Associative Mapping
    - Calculate the ***Tag*** and ***Word*** bit sizes
    - Display the binary ***Memory address*** in the ***Tag*** and ***Word*** bit fields
- Set-Associative Mapping
    - Calculate the ***Tag***, ***Set*** and ***Word*** bit sizes
    - Display the binary ***Memory address*** in the ***Tag***, ***Set*** and ***Word*** bit fields

The program should **verify** the following:

- ***Cache blocks***, ***Bytes*** and ***k*** are a base-2 value. If not, the program should display appropriate message and terminate.
- ***Memory address*** is a base-16 value. If not, the program should display appropriate message and terminate.
- ***Memory address*** size is not exceeding ***Main memory*** size. If yes, the program should display appropriate message and terminate.

Example

Suppose a byte-addressable main memory contains 1MB and cache consists of 32 blocks, where each block contains 16 bytes, and it is 4-way set associative. Specify how the main memory address 0x326A0 maps to in the cache.

Main memory

The program should display line 1 and accept user input for line 2 (Value A) and 3 (Exponent E) as given below.

```
Main memory representation is A x 2^E
Value A: 1
Exponent E: 20
```

Calculate and display the *main memory* in bytes as given below.

```
Main memory: 1 x 2^20 = 1048576 bytes
```

Cache Memory

The program should accept the three inputs for *cache blocks*, *bytes* and *k* as given below.

```
Cache blocks: 32
Bytes in Cache blocks: 16
k-set associative value: 4
```

Memory address

The program should accept the memory address in base-16 as given below.

```
Memory address in Base 16: 0x326A0
```

Calculate and display the address in base-2 **equal** to the size of the main memory as given below.

```
Address in binary: 00110010011010100000
```

Direct Cache

Calculate and display the following: Line 2 give the Tag, Line and Word bit sizes and line 3 gives the bit representation from the binary memory address for the three components as given below.

```
Direct Cache mapping of 0x326A0 address
[TAG] 11 : [LINE] 5 : [WORD] 4
[TAG] 00110010011 : [LINE] 01010 : [WORD] 0000
```

Associative Cache

Calculate and display the following: Line 2 give the Tag and Word bit sizes and line 3 gives the bit representation from the binary memory address for the two components as given below.

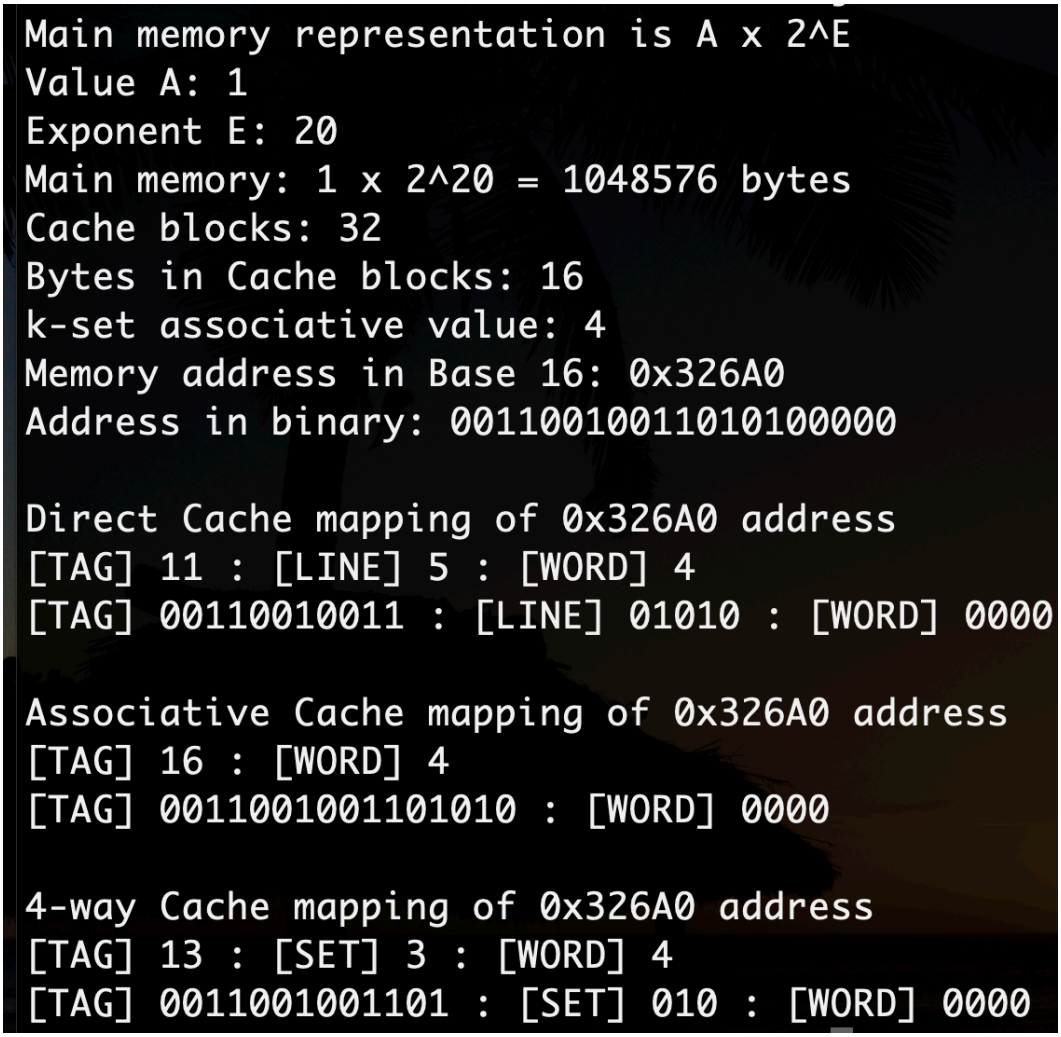
```
Associative Cache mapping of 0x326A0 address
[TAG] 16 : [WORD] 4
[TAG] 0011001001101010 : [WORD] 0000
```

Set-Associative Cache

Calculate and display the following: Line 2 give the Tag, Set and Word bit sizes and line 3 gives the bit representation from the binary memory address for the three components as given below.

```
4-way Cache mapping of 0x326A0 address
[TAG] 13 : [SET] 3 : [WORD] 4
[TAG] 0011001001101 : [SET] 010 : [WORD] 0000
```

The full program can be seen below.



Error checking

Cache blocks, ***Bytes*** and ***k*** should be checked to see if it is a base-2 value. If not, the program should display appropriate message and terminate as shown below.

```
Cache blocks: 3
Cache Blocks is not in Base 2. Program terminated
```

```
Bytes in Cache blocks: 15
Bytes in Cache blocks is not in Base 2. Program terminated
```

```
k-set associative value: 3
k-set associative values is not in Base 2. Program terminated
```

Memory address should be checked to see if it is a base-16 value. If not, the program should display appropriate message and terminate as shown below.

```
Memory address in Base 16: 0x123FG
Memory address is not in Base 16. Program terminated
```

Memory address size should not exceed ***Main memory*** size. If yes, the program should display appropriate message and terminate as shown below where memory address is 20 bits and memory address is 24 bits.

```
Main memory representation is A x 2^E
Value A: 1
Exponent E: 20
Main memory: 1 x 2^20 = 1048576 bytes
Cache blocks: 32
Bytes in Cache blocks: 16
k-set associative value: 4
Memory address in Base 16: 0x326A05
Size of address is larger than main memory size. Program terminated
```

User Input and Output

You are required to use the **exact** input and output wording and formatting as shown in the previous example.

Test Cases

Please test your software on the following test cases:

1. ***Main memory*** : 1 x 2^{20}
   Cache blocks: 32
   Bytes: 16
   K: 4
   Memory address: 0x326A0

2. ***Main memory*** : 2 x 2^{20}
   Cache blocks: 32
   Bytes: 16
   K: 4
   Memory address: 0x326A0

Submission

The **student's name**, **ID** and **honor code** is required all submitted files.

Submit only the **Java** code file to Canvas before **5pm, February 10, 2023**.

The grading rubric is given as the following:

Description	**Points**
Direct Cache calculation	15
Associative Cache calculation	15
Set-Associative Cache calculation	15
Input and output display	10
Main memory byte conversion	10
Memory address binary conversion	10
Error checking and termination	20
Code documentation	5