

1. Berechnen Sie unter Annahme von 32-Bit Adressen die folgenden Werte für einen 2-fach assoziativen 32KB großen Cache mit 32 Bytes cache lines:

- Anzahl der Bits für den Offset
 - 5Bits
- Anzahl der Sets im Cache
 - 512
- Anzahl der lines im Cache
 - 1024
- Anzahl der index bits
 - 7Bits
- Anzahl der tag bits
 - 20Bits (32Bit-5BitOffset-7BitIndex)

2. Gegeben sei ein 128 Bytes grosser 2-fach assoziativer Cache mit cache lines von 16 Bytes; der Adressraum der CPU umfasst $2^{16} = 65536$ Bytes. Als Ersetzungs-Strategie wird LRU verwendet.

Auf folgende Adressen wird zugegriffen (binär) (hier gleich die Hits und Misses mit drin:

Adresse	Hit/Miss	Eintrag in Set/Cacheline
0..0001000000	Miss	Set1/Cacheline1
0..0000010100	Miss	Set2/Cacheline3
0..0000000000	Miss	Set1/Cacheline2
0..0001000100	Hit	Set1/Cacheline1
0..0011000001	Miss	Set1/Cacheline2
0..0001001000	Hit	Set1/Cacheline1
0..0011001001	Hit	Set1/Cacheline2
TAG,INDEX,OFFSET		

a) wie ist der Cache organisiert (Skizze)? Wie viele Bits werden für tag, index, offset verwendet?

Tag = 10Bit

Offset = 4Bit

Index = 2Bit

Aufteilungsreihenfolge: Tag/Index/Offset

Set1(00)	Cacheline1
	Cacheline2
Set2(01)	Cacheline3
	Cacheline4
Set3(10)	Cacheline5
	Cacheline6
Set4(11)	Cacheline7
	Cacheline8

b) wie viele hits und misses treten auf?

Siehe Tabelle (4 Misses , 3 Hits)

c) Welche tags stehen am Ende im Cache?

Tag 2, 6, 7

3. Stellen Sie den Cache-Simulator fertig (Methode `process(tag,index,write)`); siehe die Kommentare was zu tun ist. Implementieren Sie die Ersetzungsstrategien random, LRU, FIFO und vergleichen Sie die Hits/Miss rate. Erfinden Sie eine (sinnvolle) Ersetzungsstrategie, die zwischen Lese- und Schreiboperationen unterschieden und erklären Sie Ihre Überlegungen.

4. Implementieren Sie zwei Programme, die Daten über ein Mittel der Interprozesskommunikation austauschen (wählen sie: Shared Memory oder Pipes oder Unix domain Sockets). Entweder unter MacOS, Linux oder Windows. Die zwei Programme sollen eine einfache Aufgabe kooperativ lösen; blödes Beispiel: ein Prozess wählt eine Zufallszahl, der zweite berechnet die Wurzel und gibt das Ergebnis zurück.

5. Wo wird Interprozesskommunikation praktisch eingesetzt? Und Warum? Was sind die Vorteile? AM besten Sie beschreiben eine Architektur eines Open-Source Projekt, das Interprozesskommunikation verwendet...