Homework 2

Problem 1

The following table gives the parameters for a number of different caches. Fill in the missing fields in the table for each cache. Recall that **m** is the number of memory address bits, **C** is the cache size (number of data bytes), **B** is the block size in bytes, **E** is the associativity, **S** is the number of cache sets, **t** is the number of tag bits, **s** is the number of set index bits, and **b** is the number of block offset bits.

Cache	m	C	B	E	S	t	s	b
1	64		8	2		53	8	3
2	64	2048			32		5	4
3		8192		8	128	55		
4	64	2048	4		16		4	2

Problem 2

Bob has another old machine with 4-way cache. The cache line size is 64 bytes. There are 4 sets in the cache. Alice executes the code below on the old machine. Suppose sum, i, j, k are stored in registers. sizeof(int) will return 4.

```
1.  # define M 16
2.  # define N 16
3.
4.  int a[M][N];
5.
6.  int sum() {
7.      int i, j;
8.      int sum = 0;
9.
10.     for(i = 0; i < M; i++)
11.         for(j = 0;j < N; j++)
12.             sum += a[i][j];
13.
14.    return sum ;
15. }
```

1) How many memory accesses in total are there in the loop between line 10 and line 12?

2) How many cache misses in the loop in total?

3) Suppose the latency of cache hit is 4 cycles. An access to main memory requires 200 cycles for 64 bytes. What is the average latency of accessing array elements in cycles when executing the loop between line 9 and line 11?

4) Alice wants to execute this program on a new machine, whose cache doubles in size, to get better performance. The larger cache comes in different styles. Which one(s) of the followings will help?

A. double the associativity of a set
B. double the number of sets
C. double the size of cache line

Problem 3

Suppose we have a 16-bit machine whose cache is **two-way set associative** (E=2), with a **4-byte block size** (B =4) and **four sets** (S =4). The contents of the cache are shown as follows, with all addresses, tags, and values given in hexadecimal notation:

Set Index	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3
0	281	0	40	41	42	43
	361	1	FE	97	CC	D0
1	98E	0	44	45	46	47
	361	1	15	20	05	15
2	0B5	1	48	49	4A	4B
	412	0	06	68	E6	25
3	5FF	1	9A	C0	03	FF
	084	1	5B	47	62	9F

For each of the following memory accesses, indicate if it will be a cache hit or miss when carried out **in sequence** as listed. Also give the value of a read if it can be inferred from the information in the cache.

Operation	Address	Hit (Y or N)	Read value (or unknown)
Read	0x3615		
Read	0x2816		
Read	0x084F		
Read	0x412b		
Read	0x5637		