8.32

block offset bits = log2(block size) log2(64B) = 6 bits

blocks in the cache = 32K / 64B = 32*2^{10} / 64 = 512

index bits = log2(#blocks/#sets) = log2(512/4) = 7 bits

tag bits = 32 - 6 - 7 = 19 bits

8.33

a)

#sets = #blocks / associativity

#blocks = #sets * associativity

#index bits = log2(#sets) = 8

#sets = 2^8 = 256

#blocks = 256 * 2 = 512

b)

#address bits = #tag bits + #index bits + #block offset bits
= 14 + 8 + 2 = 24

24 bit addresses implies 2^{24} bytes of memory

therefore

#memory blocks = 2^{24} bytes / 4 bytes per block = 2^{22} memory blocks

c)

d)

8.34
8B per object
128 blocks in the cache
8 objects per block

A
Misses once every 8 j cycles
1024^2 j cycles / 8 misses per j cycle = 131072 misses

B
Misses every j cycle once every 8 i cycles
1024 j cycles * 1024 i cycles * ⅛ misses per i cycle = 131072 misses

8.36
24 bit virtual addresses
16 bit physical addresses
1K page size

virtual pages = 2^{24} / 1K = 2^{24} / 2^{10} = 2^{14} = 16K pages
physical pages = 2^{16} / 1K = 2^{16} / 2^{10} = 2^{6} = 64 pages
bits in virtual page number = log2(16K) = log2(2^{14}) = 14 bits
bits in physical page number = log2(64) = log2(2^{6}) = 6 bits
bits in page offset = log2(1K) = log2(2^{10}) = 10 bits

8.37
a)
access violation
b)
access violation
c)
access violation