

- [CPSC 275: Introduction to Computer Systems](#)

[CPSC 275: Introduction to Computer Systems](#)

Fall 2025

- [Syllabus](#)
- [Schedule](#)
- [Resources](#)
- [Upload](#)
- [Solution](#)

Solution to Homework 29

1.

- A. With high-order bit indexing, each contiguous array chunk consists of 2^t blocks, where t is the number of tag bits. Thus, the first 2^t contiguous blocks of the array would map to set 0, the next 2^t blocks would map to set 1, and so on.
- B. For a direct-mapped cache where $(S, E, B, m) = (512, 1, 32, 32)$, the cache capacity is 512 32-byte blocks, and there are $t = 18$ tag bits in each cache line. Thus, the first 2^{18} blocks in the array would map to set 0, the next 2^{18} blocks to set 1. Since our array consists of only $(4096 * 4)/32 = 512$ blocks, all of the blocks in the array map to set 0. Thus, the cache will hold at most one array block at any point in time, even though the array is small enough to fit entirely in the cache. Clearly, using high-order bit indexing makes poor use of the cache.

2.

No. virtual address bits (n)	No. virtual addresses (N)	Largest possible virtual address
8	$2^8 = 256$	$2^8 - 1 = 255$
16	$2^{16} = 64K$	$2^{16} - 1 = 64K - 1$
32	$2^{32} = 4G$	$2^{32} - 1 = 4G - 1$
48	$2^{48} = 256T$	$2^{48} - 1 = 256T - 1$
64	$2^{64} = 16,384P$	$2^{64} - 1 = 16,384P - 1$

3.

n	$P = 2^p$	# PTEs
16	4K	16
16	8K	8
32	4K	1M
32	8K	512K

- **Welcome: Sean**

- [LogOut](#)

