Memory

Memory Technology
- SRAM: static random access memory
- DRAM: dynamic RAM
- SDRAM: synchronous DRAM
- ROM: read only memory
- PROM: programmable ROM
- EPROM: erasable PROM

EEPROM: electrically erasable PROM
Flash is EEPROM
Disk: magnetic media

Principle of locality
- Spatial locality
- Temporal locality

Hierarchical:
- A: register file
- B: L1 cache
- C: L2 cache
- D: L3 cache
- E: main memory
- F: disk

```c
int sumvec (l) {
    int sum, i;
    sum = 0;
    for (i = 0; i < N; i++) {
        sum = sum + vec[i];
    }
    return (sum);
}
```
Movements of data between levels in fixed size units called blocks
A → B word 32-bit or 64-bits
B ← C 4 k & F words
B → E 4 Kbyte or more
on miss, more block from level D (when it is) to level E. This might displace something from level E.

Read replacement policy
(e.g., random, least recently used (LRU))
consider two-level cache system
B → E

CACHE is org. as collection of sets
if S = 2^s is # of sets, s is # bits to ID set
each set contains 1 or more cache lines
each line holds 1 block
C is # of lines per set
if D = 2^d is block size (size per set)
in bytes, b is # bits to 32 byte within block

Each line stores:
- block (data)
- valid bit
- tag bits (where in memory?)
- dirty bit (is memory up to date?)

- if M is size of main memory, M = 2^m
m is # bits of address
(logical addr)
characterize cache org. by $(L, S, B, m)$

divide m address bits into 3 components

- tag
- set
- index block offset

1) direct-mapped cache $(E = 1)$

- set size is $1/n$ lines
- line size is $5 \times B$ in bytes

- on write, can either write through
  - or write back

- on write miss, write allocate or
  - write no allocate

Classic pairs:
1) write through - no allocate
2) write back - allocate