This works because of locality
2 types of locality

**Spatial locality**
- if I'm accessed it's likely my neighbor will be

**Temporal locality**
- I'm likely to be reused

> instruction stream
  > stack accesses

> reading an array

> access data repeatedly
  > loops of instructions
  > stack accesses
Sum = 0
for (int i = 0; i < n; i++) {
    Sum += a[i];
} -> spatial

return Sum;

spatial locality in instruction stream + temporal locality in the loop

Caches take advantage of locality
- Temporal locality -> keep recently used things close
- Spatial locality -> load more than was asked for

> 4-bytes
> 64-bytes
> much bigger to take advantage of spatial locality
CPU checks cache. If it's there: **HIT**
If not there: **MISS**

3 different kinds of cache misses (3 C's)

- **Cold/compulsory** misses
  - **Hit** miss because cache is empty
  - "Warm up" caches

- **Conflict** misses
  - Cache could hold the data, but because of limited "slots" it's a miss

- **Capacity** miss
  - Cache is full
  - Working set is too big for cache

Caches are managed by hardware.
No software/OS/etc involved.
Logically transparent to the program.
In contrast to registers which are managed by the program.
Average memory access time or miss ratio

\[ (T_{hit}) + (T_{miss}) \cdot (\text{miss rate}) \]

Cache hit time \( \text{L1} \) in ns

miss time \( \text{L1} \) in ns

Hit rate \( \rightarrow 99\% \) \( \rightarrow \) Miss rate = 1%

\[ \frac{\# \text{hits}}{\text{total \# accesses}} \cdot \frac{\# \text{misses}}{\text{total \# of accesses}} \]

\( \text{AMAT} = \text{L1 ns} + 100 \text{ns} \cdot (.01) \)
\[ = 2 \text{ns} \]

\( T_{miss} = AMAT_{L2} = T_{hit L2} + T_{miss L2} \cdot (\text{L2 miss rate}) \)

L2 miss rate = 25%

L2 hit time = 100ns

L2 miss time = 100ns

\( \text{AMAT} = \text{L1 ns} + (.01) \cdot [100 \text{ns} + .25 \cdot 100] \)
\[ = 1 + .01 \cdot (35) \]
\[ = 1.35 \text{ns} \]
\[ AMAT = T_{hit} + T_{miss} \cdot (\text{rate}) \]

\[ T_{miss} = AMAT_{L2} = T_{hitL2} + T_{missL2} \cdot (\text{L2 rate}) \]

\[ AMAT = T_{hitL1} + (L1 miss) \cdot [T_{hitL2} + T_{missL2} \cdot (\text{L2 rate})] \]