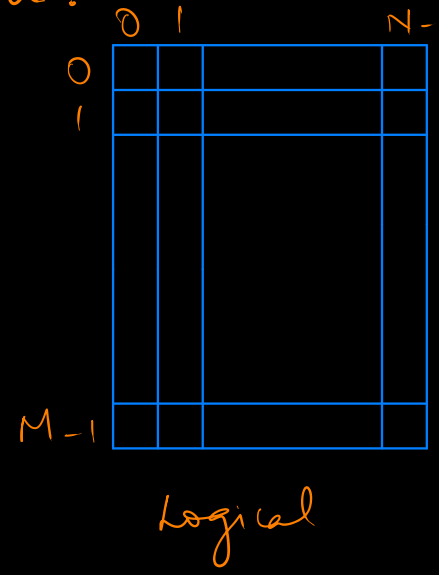
Sai

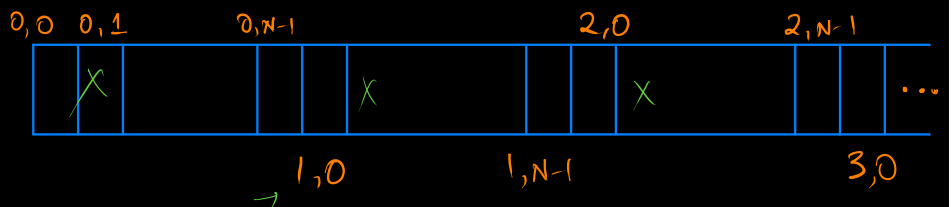
EFFECT OF LOCALITY ON CACHE

```
#define M 1000
#define N 100
float a[M][N] = { };
float b[N] = { };
int i = 0, j = 0;
// a is set here
for(j = 0; j < N; j++)
    for(i = 0; i < M; i++)
        b[j] += a[i][j];
```

a:

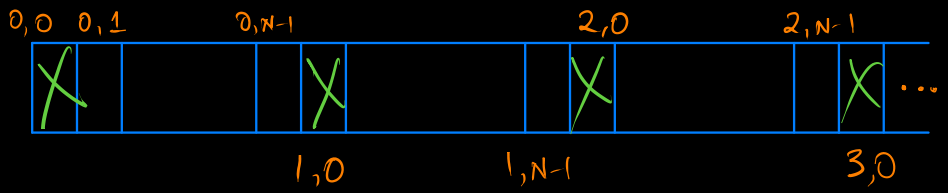


Logical



Physical
(Row major order)

How are elements accessed from a ?



Physical

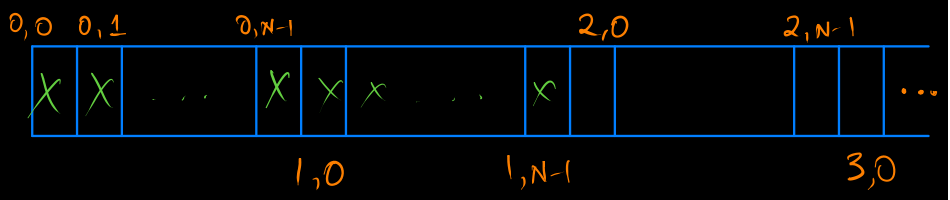
⇒ Strided Access (lack of spatial locality)

⇒ Leads to poor memory system performance

Can the program be altered for better performance?

```
for (i = 0; i < M; i++)
    for (j = 0; j < N; j++)
        b[j] += a[i][j];
```

How are elements accessed from a?



Physical

Ratio of Peak FLOPS rate to peak Memory Bandwidth ranges between 1 - 100 MFLOPS/MB/s

↙ SuperComputers

↘ Microprocessor based systems

⇒ A word must be reused 100 times after being fetched into L1, on an average.

⇒ Exploit spatial and temporal locality

⇒ Memory layout and organizing computations can significantly impact locality.