

6. Measuring Cache Effects

[Homework P19](#)

Consider a 2D array with each row occupying one page of memory:

```
#include <unistd.h>
#include <stdlib.h>
...
int main( void)
{
    int nrows = 300, PAGE_SIZE = sysconf(_SC_PAGE_SIZE), ncols = PAGE_SIZE/sizeof(int);

    int *a = calloc( nrows*ncols, sizeof(int));
```

Note that a dynamically allocated 2D array is stored as a 1D array, so `a[row][col]` is really `a[row*ncols+col]`

Accessing the array by rows should be much faster than access by columns, due to TLB and data caching:

```
// by rows
//
for( int row = 0; row < nrows; ++row)
    for( int col = 0; col < ncols; ++col)
        a[row*ncols+col] += 1;
```

or:

```
// by cols
//
for( int col = 0; col < ncols; ++col)
    for( int row = 0; row < nrows; ++row)
        a[row*ncols+col] += 1;
```