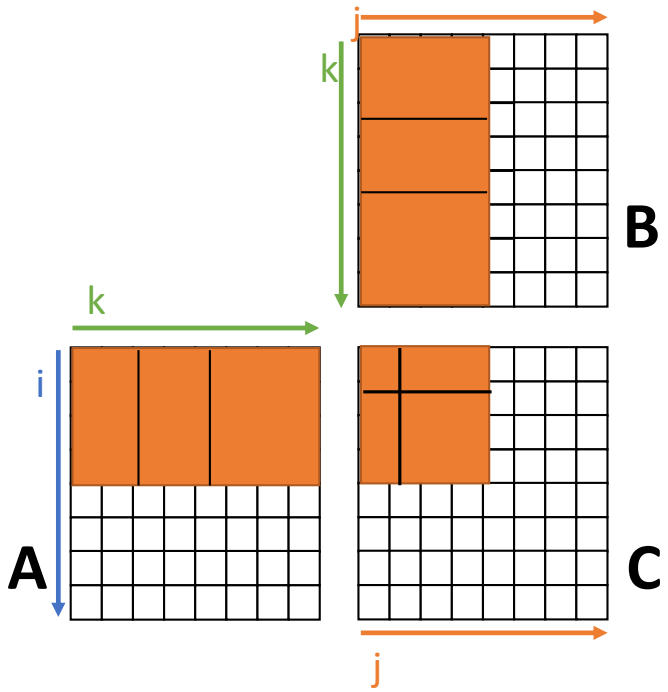
Homework

- Finish the code for matrix multiplication

```
__shared__ float smem_c[64][64];
__shared__ float smem_a[64][8];
__shared__ float smem_b[8][64];
int c = blockIdx.x * 64;
int r = blockIdx.y * 64;
for (int kk=0; kk<N; kk+=T) {
  for (int i=threadIdx.x+blockDim.x*threadIdx.y;
i<64*8; i+=blockDim.x*blockDim.y) {
    int k = kk + i / 64;
    int rt = r + i % 64;
    int ct = c + i % 64;
    smem_a[i%64][i/64] = A[rt*N+k];
    smem_b[i/64][i%64] = B[k*N+ct];
}
__syncthreads();
…..
```



Experiment with the use of coalesced memory accesses and avoiding bank conflicts in your implementation. Also experiment with different size of matrices and report execution times.