```
__global__ void matrix_mul(float* d_a, float* d_b,
                           float* d_c, int width)
{

  int row = blockIdx.x * blockDim.x + threadIdx.x;
  int col = blockIdx.y * blockDim.y + threadIdx.y;

  if ((row < width) && (col < width))
    {
      float single_entry = 0;
      // each thread computes one
      // element of the block sub-matrix
      for (int i = 0; i < width; ++i)
        {
          single_entry += d_a[row*width+i]*d_b[i*width+col];
        }
      d_c[row*width+col] = single_entry;
    }
}
```