

CS425: HW5 CUDA
Cameron Hall, Arthur Wuterich
11/22/2015
Problem 1

A.

```
__global__ void add( int *a, int *b, int *c ) {  
    int tid = blockIdx.x;  
    int start = tid * 2;  
    int end = (tid + 1) * 2;  
  
    for (int i = start; i < end; ++i) {  
        if (i < N)  
            c[i] = a[i] + b[i];  
    }  
}
```

To assign elements to each thread, I used the range $(\text{blockIdx.x} * 2, (\text{blockIdx} + 1) * 2)$. I chose this approach because it's simple and introduces no additional control divergence.

B.

I will create 25 blocks with 1024 threads each.

C.

For the kernel in Part A, I expect one warp to take a divergent path through the code. The warp that encompasses the addition for the vector element 50,000 will have some threads that pass the if condition each time, and others that fail it. I don't expect this divergence to affect the performance of the kernel much at all.