CUDA - hello world!

The following program take the string "Hello ", send that plus the array 15, 10, 6, 0, -11, 1 to a kernel. The kernel adds the array elements to the string, which produces the array "World!". This string is passed back to the host and printed out (collected from http://computer-graphics.se).

```c
#include <stdio.h>

const int N = 16;
const int blocksize = 16;

__global__
void hello(char *a, int *b)
{
        a[threadIdx.x] += b[threadIdx.x];
}

int main()
{
        char a[N] = "Hello \0\0\0\0\0\0";
        int b[N] = {15, 10, 6, 0, -11, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};

        char *ad;
        int *bd;
        const int csize = N*sizeof(char);
        const int isize = N*sizeof(int);

        printf("%s", a);

        cudaMalloc( (void**)&ad, csize );
        cudaMalloc( (void**)&bd, isize );
        cudaMemcpy( ad, a, csize, cudaMemcpyHostToDevice );
        cudaMemcpy( bd, b, isize, cudaMemcpyHostToDevice );

        dim3 dimBlock( blocksize, 1 );
        dim3 dimGrid( 1, 1 );
        hello<<<dimGrid, dimBlock>>>(ad, bd);
        cudaMemcpy( a, ad, csize, cudaMemcpyDeviceToHost );
        cudaFree( ad );
        cudaFree( bd );

        printf("%s\n", a);
        return EXIT_SUCCESS;
}
```

You can now compile the code `nvcc hello.cu -o hello.out` and then run:

```
./hello.out
Hello world!
```