Excercise 1

- Get accustomed to the CUDA programming environment!

Compile and run the first CUDA example (vector addition)
Excercise 2

Increase the vector length in Ex. 1 by factor 1024. Use 2-dim grid!
Try different blocksizes: 2, 16, 64, 256!
What is the difference in runtime?
Excercise 3

Modify program from Ex. 1:

- array hA (and dA) shall be N*N matrix, with
  \[
  \text{for(int } j=0; j<N; j++) \text{ hA[j*N+i] = 0.5*(float)(i-j)}; 
  \]

- Compute Matrix-Vector product in vecadd, i.e. compute
  \[
  c[i]=0; \text{ for(j=0; j<N; j++) } c[i]+=a[i*N+j]*b[j]; 
  \]
Excercise 4

Given is list of numbers. Compute (approximately) the log3 of all numbers by repeated division. Then parallelize red line!

```c
#define N 65536
int i, cnt=0, len=N;
float list[N];
for(i=0;i<N;i++) list[i]=5.0+(float)(i-N/2)*(float)(i-N/2);
do{
    cnt++;
    for(i=0;i<len;i++) list[i]=list[i]/3.0;
    for(i=0;i<len;) if(list[i]<3.0){
        printf("log3(%f)=%d\n",list[i],cnt); list[i]=list[len-1]; len--;
    } else i++;
}while(len>0);
```
Thank you very much
for concentrated working