Assignment-3

2. Write a CUDA program to print TEXT using one thread and one block.

```
#include<iostream>
using namespace std;
__global__ void printHello(){
}
int main(){
        printHello<<<1,1>>>();
cout<<”Hello World”;
        return 0;
```

3. Write a CUDA program for vector addition using one thread and multiple blocks.

```
using namespace std;
__global__ void add(int *a, int *b, int *c){
        int i = blockIdx.x;
        c[i] = a[i]+b[i];
}
int main(){
        int c[6];
        int a[6] = {1,2,3,4,5,6};
        int b[6] = {11,12,13,14,15,16};
        int *da, *db, *dc;
        cudaMalloc(&da, 6*sizeof(int));
        cudaMalloc(&db, 6*sizeof(int));
        cudaMalloc(&dc, 6*sizeof(int));
        cudaMemcpy(da, &a, 6*sizeof(int), cudaMemcpyHostToDevice);
        cudaMemcpy(db, &b, 6*sizeof(int), cudaMemcpyHostToDevice);
        add<<<6,1>>>(da,db,dc);
        cudaMemcpy(&c, dc, 6*sizeof(int), cudaMemcpyDeviceToHost);
        for (int j=0; j<6; j++){
                cout<<a[j]<<" + "<<b[j]<<" = "<<c[j]<<endl;
        }
        cudaFree(da);
        cudaFree(db);
        cudaFree(dc);
        return 0;
}
```

4. Write a CUDA program for vector subtraction using one block and multiple threads.

```
#include<iostream>
using namespace std;
__global__ void add(int *a, int *b, int *c){
       int i = blockIdx.x;
       c[i] = a[i]+b[i];
}
int main(){
       int c[6];
       int a[6] = {1,2,3,4,5,6};
       int b[6] = {11,12,13,14,15,16};
       int *da, *db, *dc;
       cudaMalloc(&da, 6*sizeof(int));
       cudaMalloc(&db, 6*sizeof(int));
       cudaMalloc(&dc, 6*sizeof(int));
       cudaMemcpy(da, &a, 6*sizeof(int), cudaMemcpyHostToDevice);
       cudaMemcpy(db, &b, 6*sizeof(int), cudaMemcpyHostToDevice);
       add<<<6,1>>>(da,db,dc);
       cudaMemcpy(&c, dc, 6*sizeof(int), cudaMemcpyDeviceToHost);
       for (int j=0; j<6; j++){
              cout<<a[j]<<" + "<<b[j]<<" = "<<c[j]<<endl;
       }
       cudaFree(da);
       cudaFree(db);
       cudaFree(dc);
       return 0;
}
```