Write a CUDA Program for : 1. Addition of two large vectors

2. Matrix Multiplication using CUDA C

```
#include <cuda_runtime.h>
#include <iostream>
__global__ void matmul(int* A, int* B, int* C, int N) {
    int Row = blockIdx.y*blockDim.y+threadIdx.y;
    int Col = blockIdx.x*blockDim.x+threadIdx.x;
    if (Row < N && Col < N) {
        int Pvalue = 0;
        for (int k = 0; k < N; k++) {
            Pvalue += A[Row*N+k] * B[k*N+Col];
        }
        C[Row*N+Col] = Pvalue;
    }
}

int main() {
    int N = 512;
    int size = N * N * sizeof(int);
    int* A, * B, * C;
    int* dev_A, * dev_B, * dev_C;
    cudaMallocHost(&A, size);
    cudaMallocHost(&B, size);
    cudaMallocHost(&C, size);
    cudaMalloc(&dev_A, size);
cudaMalloc(&dev_B, size);
cudaMalloc(&dev_C, size);

// Initialize matrices A and B
for (int i = 0; i < N; i++) {
    for (int j = 0; j < N; j++) {
        A[i*N+j] = i*N+j;
        B[i*N+j] = j*N+i;
    }
}

cudaMemcpy(dev_A, A, size, cudaMemcpyHostToDevice);
cudaMemcpy(dev_B, B, size, cudaMemcpyHostToDevice);

dim3 dimBlock(16, 16);
dim3 dimGrid(N/dimBlock.x, N/dimBlock.y);

matmul<<<dimGrid, dimBlock>>>(dev_A, dev_B, dev_C, N);

cudaMemcpy(C, dev_C, size, cudaMemcpyDeviceToHost);

// Print the result
for (int i = 0; i < 10; i++) {
    for (int j = 0; j < 10; j++) {
        std::cout << C[i*N+j] << " ";
    }
```

```
        std::cout << std::endl;
    }

    // Free memory
    cudaFree(dev_A);
    cudaFree(dev_B);
    cudaFree(dev_C);
    cudaFreeHost(A);
    cudaFreeHost(B);
    cudaFreeHost(C);

    return 0;
}
```

OUTPUT:-

```
0 512 1024 1536 2048 2560 3072 3584 4096 4608
512 2048 3584 5120 6656 8192 9728 11264 12800 14336
1024 3584 6144 8704 11264 13824 16384 18944 21504 24064
1536 5120 8704 12288 15872 19456 23040 26624 30208 33792
2048 6656 11264 15872 20480 25088 29696 34304 38912 43520
2560 8192 13824 19456 25088 30720 36352 41984 47616 53248
3072 9728 16384 23040 29696 36352 43008 49664 56320 62976
3584 11264 18944 26624 34304 41984 49664 57344 65024 72704
4096 12800 21504 30208 38912 47616 56320 65024 73728 82432
4608 14336 24064 33792 43520 53248 62976 72704 82432 92160
```