

```

#include<stdio.h>

__global__ void vecadd(float *a, float *b, float *c, int n)
{
    int i= threadIdx.x + blockDim.x*blockIdx.x;
    if(i<n)
        c[i] = a[i]+b[i];
}

int main(){
    int n;
    scanf("%d",&n);
    int a[n],b[n];
    for(int i=0; i<n; i++)
        scanf("%d",&a[i]);
    for(int i=0;i<n; i++)
        scanf("%d",&b[i]);
    int c[n];
    float *da,*db,*dc;
    int size = n*sizeof(float);
    cudaMalloc((void **) &da,size);
    cudaMalloc((void **) &db,size);
    cudaMalloc((void **) &dc,size);

    cudaMemcpy(da,a,sizeof(int),cudaMemcpyHostToDevice);
    cudaMemcpy(db,b,sizeof(int),cudaMemcpyHostToDevice);
    // cudaMemcpy(dc,c,sizeof(int),cudaMemcpyHostToDevice);
    // cudaMemcpy(n,n,sizeof(int),cudaMemcpyHostToDevice);

    vecadd<<<ceil(n/32.0),15>>>(da,db,dc,n);

    cudaMemcpy(c,dc,sizeof(int),cudaMemcpyDeviceToHost);

    for(int i=0; i<n; i++)
        printf("%d ",c[i]);
    cudaFree(da);
    cudaFree(db);
    cudaFree(dc);

    return 0;
}

```