```
#include<stdio.h>

__global__ void sum(int *a_d, int n,int* maxsum){
      int strid=n/2;
      //printf("fas");
      int t=threadIdx.x;
      while(strid>=1){
      //printf("af");
            __syncthreads();
            if(t<strid){
                  a_d[t]=a_d[t]+a_d[strid+t];
                   // printf("threadid=%d val=%d\n",t,a_d[t]);
            }
            strid/=2;
      }
      maxsum[0]=a_d[0];
}

int main(){
      int n=512;
      int a[n];
      for(int i=0; i<n; i++){
            a[i]=i;
      }
      int *a_d,*maxsum;
      cudaMalloc((void**)&a_d,n*sizeof(int));
      cudaMalloc((void**)&maxsum,sizeof(int));
      //cudaMalloc((void**)&n_d,sizeof(int));
      cudaMemcpy(a_d,a,n*sizeof(int),cudaMemcpyHostToDevice);
      //for(int i=0; i<n; i++)
      //printf("%d ",a[i]);
      //printf("\n");
      sum<<<1,n>>>(a_d,n,maxsum);
      int maxi[n];
      int max_val[1];
      cudaMemcpy(maxi,a_d,n*sizeof(int),cudaMemcpyDeviceToHost);
      cudaMemcpy(max_val,maxsum,sizeof(int),cudaMemcpyDeviceToHost);
//    for(int i=0; i<n; i++)
      printf("%d ",max_val);
      printf("%d ",maxi[0]);
return 0;
}
```