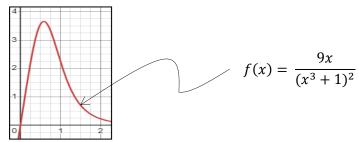
Homework Assignment 10 – due on Saturday, December 7 (Midnight) Description of Assignment:

Complete the CUDA program(area.cu) which computes the area (using the trapezoidal rule) under the curve of a graph f(x) shown in the following figure. Your program should measure execution time using cudaEvent and calculate GFLOPS. In the program, use **double** and **long** for variables. Use #segments ≤ 100000 for tests.



```
#include <stdio.h>
                                                                    if (nt > MAX_THREADS) {
#include <unistd.h>
                                                                        fprintf(stderr, "%d threads are used.\n", MAX THREADS);
                                                                        nt = MAX THREADS;
#include <pwd.h>
#define MAX THREADS 256
                                                                    dim3 dimBlock(nt);
#define gflops(n,ms) (((n*18.0)/(ms/1.0e+3))/1.0e+9)
                                                                    dim3 dimGrid((N+dimBlock.x-1)/dimBlock.x);
                                                                    smsize = sizeof(double)*nt;
  _device__ double f(double x)
                                                                    dev = (getpwuid(getuid())->pw_name[3]-'0')%2? 1: 0;
                                                                    cudaSetDevice(dev);
   // (1) COMPLETE
                                                                    cudaGetDeviceProperties(&deviceProp, dev);
                                                                    printf("Device(%d) used: \"%s\"\n", dev, deviceProp.name);
  _global__ void area_kernel(double *local_area, long N, double
a, double b)
                                                                    local area = (double*)malloc(sizeof(double)*dimGrid.x);
                                                                    cudaMalloc((void **)&local area d, sizeof(double)
                                                                 *dimGrid.x);
   double dx. x:
   long i = blockDim.x*blockIdx.x+threadIdx.x;
   int half;
                                                                    // (3) timing: start
   extern __shared__ double sdata[];
                                                                    // (4) call GPU kernel function
   // (2) COMPLETE
                                                                    // (5) copy values from GPU memory to CPU memory
                                                                    // (6) timing: stop
int main()
   double *local area, *local area d, area, a, b;
                                                                    // (7) sum local area's computed on GPU
   long N:
   int nt, i, smsize, dev;
                                                                    printf("area: %5.5lf\n", area);
                                                                    printf("elapsed time: %5.2f milliseconds", elapsed);
   cudaEvent t start, stop;
                                                                    printf(" (GFLOPS: %5.2f)\n", (N*18.0/(elapsed/1.0e+3))
   float elapsed:
   cudaDeviceProp deviceProp;
                                                                 /1.0e+9);
                                                                     free(local_area);
       fprintf(stderr, "usage: %s #segments #threads\n", argv[0]);
                                                                    cudaFree(local_area_d);
       exit(1);
                                                                    exit(0);
   N = atol(argv[1]);
                                                                 }
   nt = atoi(argv[2]);
```

Turnin the assignment:

After done your assignment, type **turnin** in your current working directory. You can retype the command at any time before the due date.