CS 382: Homework Assignment 5
Due: December 9, 11:55pm

Collaboration Policy. Homeworks will be done **individually**: each student must submit their own answers. It is acceptable for students to collaborate in understanding the material but not in solving the problems or programming. Use of the Internet is allowed, but should not include searching for existing solutions.

Under absolutely no circumstances code can be exchanged between students. If some code was shown in class, it can be used, but it must be obtained from Canvas, the instructor or the TA.

Assignments from previous offerings of the course must not be re-used. Violations will be penalized.

Late Policy. No late submissions will be allowed without consent from the instructor. If urgent or unusual circumstances prohibit you from submitting a homework assignment in time, please e-mail me.

Deliverable. A single **pdf** file on Canvas. **HANDWRITTEN SUBMISSIONS ARE PROHIBITED**.

Problem 1 (25 points) We define the Arithmetic Intensity of code as the number of operations in the program divided by the number of bytes accessed in the main memory. Operations considered are those that lead to output generation, not those needed for loops and conditionals. What is the arithmetic intensity of the following snippets of code? Assume that all data are double precision, and that the system has no cache, but has a small number of registers, like the LEGv8 processor. Explain your answers briefly, emphasizing how each example differs from the previous ones.

(a)

```
for (j=0; j<100; j++)
    c[j] = a[j] * b[j];
```

(b)

```
for (j=0; j<100; j++)
    a[j] = a[j] * b[j];
```

(c)

```
for (j=0; j<100; j++)
    c[j] = 4.2 * b[j];
```

(d)

```
for (j=0; j<100; j++)
    d[j] = a[j] * b[j] + c[j];
```

(e)

```
for (j=0; j<100; j++)
    a[j] = a[j] * b[j] + b[j];
```

Problem 2 (25 points) Assume a GPU that runs at 2.5 GHz with 8 SIMD processors, each having 32 single –precision FP units. This GPU is supported by a 112 GB/s off-chip memory. Assume all memory latencies can be hidden.

(a) (10 points) Ignoring memory bandwidth, what is the peak SP FP operation in GFLOPs?

(b) (15 points) Is this throughput sustainable given the bandwidth for performing SAXPY ($Z[i] = a * X[i] + Y[i]$) on large amounts of data? Sustainable throughput here means that the memory can provide enough data to the processing units to operate at full capacity. Justify your answer. (Problem 1 may be helpful here.)

Problem 3 (20 points) Parallelizable programs are typically accelerated by a factor of 100 on a GPU with 2,000 cores. What percentage of the total execution time (on a single core) of such programs is allocated to sequential parts of the program and what percentage of time is allocated to parallelizable parts of the program? Explain your steps clearly.

Problem 4 (30 points) Consider a network with 16 nodes that can be configured in three different ways:

1. as a ring,
2. as a 4×4 2D square grid (mesh), where each node is connected up to 4 nearest neighbors,
3. as a fully connected network.

Assume that all links provide the same bandwidth, equal to 1 Gbit/s. For each topology, answer the following questions:

(a) (12 points) What is the total bandwidth?

(b) (12 points) What is the minimum bisection bandwidth?

(c) (6 points) How many links can fail but still allow us to guarantee that an unbroken link will exist to connect any node to any other node?