9 GPUs and SIMD [35 points]

We define the *SIMD utilization* of a program that runs on a GPU as the fraction of SIMD lanes that are kept busy with *active threads* during the run of a program. As we saw in lecture and practice exercises, the SIMD utilization of a program is computed across the *complete run* of the program.

The following code segment is run on a GPU. Each thread executes **a single iteration** of the shown loop. Assume that the data values of the arrays `A` and `B` are already in vector registers so there are no loads and stores in this program. (Hint: Notice that there are 4 instructions in each iteration.) A warp in the GPU consists of 32 threads, and there are 32 SIMD lanes in the GPU.

```
for (i = 0; i < 1026; i++) {
    if (A[i] < 33) {         // Instruction 1
        B[i] = A[i] << 1;    // Instruction 2
    }
    if (A[i] > 33) {         // Instruction 3
        B[i] = A[i] >> 1;    // Instruction 4
    }
}
```

Please answer the following five questions.

(a) [2 points] How many warps does it take to execute this program?

33 warps.

Explanation:
The number of warps is calculated as:
$\#Warps = \lceil \frac{\#Total\_threads}{\#Warp\_size} \rceil$,

where
$\#Total\_threads = 1026 = 2^{10} + 2$ (i.e., one thread per loop iteration),

and
$\#Warp\_size = 32 = 2^5$ (given).

Thus, the number of warps needed to run this program is:
$\#Warps = \lceil \frac{2^{10}+2}{2^5} \rceil = 2^5 + 1 = 33$.

(b) [10 points] What is the maximum possible SIMD utilization of this program? Show your work. (Hint: The warp scheduler does not issue instructions where no threads are active).

$\frac{3076}{3136} = \frac{769}{784}$.

Explanation:
The maximum SIMD utilization is achieved when all threads of the complete warps follow the same execution path and execute `Instruction 2` or `Instruction 4` (`A[i] > 33` or `A[i] < 33`), and the two active threads of the last warp do not execute `Instruction 2` or `Instruction 4` (`A[i] = 33`).
The maximum SIMD utilization sums to $\frac{1026+32\times32+1026}{1056+1024+1056} = \frac{3076}{3136}$.

(c) [5 points] Please describe what needs to be true about array `A` to reach the maximum possible SIMD utilization asked in part (b). (Please cover all cases in your answer.)

For every 32 consecutive elements of `A` out of the first 1024 elements, every element should be lower than 33 (`if(A[i] < 33)`), or greater than 33 (`if(A[i] > 33)`). The last two elements should be equal to 33. (NOTE: The solution is correct if the three cases are given.)

(d) [13 points] What is the minimum possible SIMD utilization of this program? Show your work.

$\frac{353}{704}$.

Explanation:
`Instruction 1` is executed by every active thread ($\frac{1026}{1056}$ utilization).
The minimum SIMD utilization of `Instruction 2` occurs if only one thread per warp executes it.
`Instruction 3` is again executed by every active thread ($\frac{1026}{1056}$ utilization).
Finally, the minimum SIMD utilization of `Instruction 4` occurs if only one thread per warp executes it.

The minimum SIMD utilization sums to $\frac{1026+1\times33+1026+1\times33}{1056+1056+1056+1056} = \frac{353}{704}$.

(e) [5 points] Please describe what needs to be true about array `A` to reach the minimum possible SIMD utilization asked in part (d). (Please cover all cases in your answer.)

For every 32 consecutive elements among the first 1024 elements of `A`, one element should be lower than 33 (`if(A[i] < 33)`), one element should be greater than 33 (`if(A[i] > 33)`), and the remaining 30 elements should be equal to 33.
For the last 2 elements of `A`, one element should be lower than 33 (`if(A[i] < 33)`), and the other element should be greater than 33 (`if(A[i] > 33)`).