QUESTION #2

Most of the application programs consist of loop structures (e.g., **for** loops and **while** loops). The term “the size of a loop structure” refers to the number of instructions in a loop structure (i.e., the instruction count from the beginning to the end of a loop structure – see the figure below). Suppose that programmers do not apply any compile-time code optimization techniques, such as loop-unrolling, what is the impact of “the size of a loop structure” to the performance (execution speed) improvement for a tightly-coupled multi-processor system? Technically justify your response.

![Diagram of loop structure](image)

**Solution:**

The larger the size of loop structures gets, the more opportunities for parallel execution will be there. For example, if a loop structure consists of only two instructions, only up to two instructions can be executed in parallel, while more instructions potentially executed in parallel in a loop structure that consists of a large number of instructions.

Thus, the larger the size of loop structure gets, more opportunities for parallel executions, making more performance gain by a tightly-coupled multi-processor system.