

Worksheet #2: Reverse Engineering a Parallel Program from a Computation Graph (CG)

Write a parallel program that generates exactly the same ordering constraints as the computation graph shown. The program should be written in pseudocode using `finish` and `async` annotations. The CG nodes should be clearly identified as statements in the program e.g., as method calls `A()`, `B()`, etc. Since the CG edges are not labeled as `spawn`, `continue`, or `join`, you can make whatever assumptions you choose about the edges when writing your program. The only requirement is that the ordering constraints in your program coincide with those in the graph. Submit solution in Canvas.

