Parallelism Overview Solutions

Concurrency and Parallelism

- Briefly explain the difference between concurrency and parallelism
  - Concurrency is when threads perform different tasks
  - Concurrency is mainly used when programs are structured for "separation of concerns" (to keep conceptually tasks distinct and improve responsiveness)
  - Parallelism is when threads perform the same task
  - Parallelism is mainly used for computationally intensive work, which can be split into units, and these units can be processed independently of each other

Explicit and Implicit Parallelism

- Briefly explain the difference between explicit and implicit parallelism
  - In explicit parallelism, the programmer has to write code which performs the tasks in parallel
  - In particular, the programmer has to choose how many threads to use and start them
  - This can be useful if the problem requires a specific number of tasks, or the program will run on fixed hardware
  - In implicit parallelism, the system will automatically start the optimal number of threads
  - This is usually the best solution