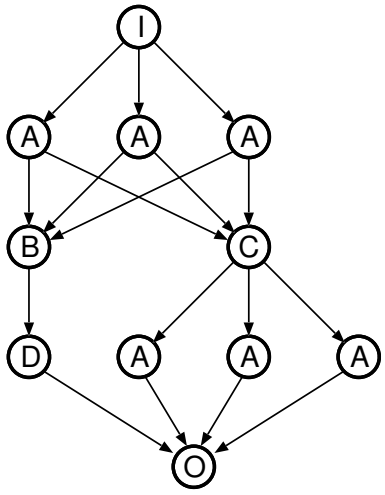
Parallel and Distributed Computing

2020/2021

$2^{\underline{\text{nd}}}$ Exercise Class

1. Consider the following data dependency graph. Identify situations of data parallelism and functional parallelism.



2. An application has been re-written for a parallel machine with 10 processors, with a measured speedup of 6. Assuming that the original application is composed of sections either completely parallelizable or purely sequential, determine the fraction that is completely parallelizable.

3. The level of parallelization of an application can be approximated by a linear model expressed by the following equation:

$$f(q,\theta) = \beta\left[(1-\theta)\cdot(p+1) + (2\theta-1)\cdot q\right]$$

where $f(q,\theta)$ is the probability of, at any given time, q processors are active ($1 \leq q \leq p$). Parameter θ ($0 \leq \theta \leq 1$) allows the modeling of the higher (θ closer to 1) or lesser (θ closer to 0) degree of parallelism of applications.

   a) Compute β as a function of p and θ.

   b) Re-write Amdahl's law to obtain speedup as a function of $f(q,\theta)$.

c) Determine the speedup for $\theta = 0$, $\theta = 0.5$ e $\theta = 1$. Comment the result obtained. (if needed, you can assume that, for large p, $\sum_{q=1}^{p} \frac{1}{q} \approx \ln(p) + 1$)

4. Consider a shared-memory computer with 8 processors. Assume that for such a machine it was determined that on average each instruction takes 2 clock cycles to execute and makes 1.25 memory accesses, already taking into account in these values a cache hit rate of 99%. If an access to main memory takes 10 clock cycles, what is the rate of occupation of the bus to main memory?

   How would you compute the bus rate of occupation for a machine with 16 processors?

5. Consider a shared-memory system (UMA), as the one depicted in the figure below, with one level of cache private to each of the p processors. To improve system performance, the addition of a second level of cache is under study. Discuss the relative merits of the following two options:

   A. place a single cache of size C next to main memory, hence shared by all processors

   B. use p caches of size C/p, private to each processor, placed after the cache of level 1