$\mathcal{NC}$ and $\mathcal{P}$-Completeness

Many of the problems that you are familiar with are in the class $\mathcal{NC}$. For example, the 0/1 version of the shortest path problem is in $\mathcal{NC}$, and every context-free language is in the class $\mathcal{NC}$.

Whether $\mathcal{NC} = \mathcal{P}$ is an open question of enormous importance.

The Circuit Value Problem, or the Boolean Circuit Problem

We now give a $\mathcal{P}$-complete problem, namely the circuit value problem.

An instance of the Circuit Value Problem is a sequence of n Boolean assignments.

1. The left side of the i^{th} assignment is the Boolean variable x_i.
2. The right side of the i^{th} assignment is one of the following.
   (a) 0 (false)
   (b) 1 (true)
   (c) x_j for $j < i$
   (d) $!\,x_j$ for $j < i$ (! means 'not')
   (e) $x_j + x_k$ for $j < i$ and $k < i$ (+ means 'or')
   (f) $x_j * x_k$ for $j < i$ and $k < i$ ($*$ means 'and')
3. The answer to an instance of the CVP is the value of x_n.

Trivially, CVP is in $\mathcal{P}$. Simply execute the n statements in order.

It is known that CVP is $\mathcal{P}$-complete, which implies that if it is in Nick's Class, then $\mathcal{NC} = \mathcal{P}$.

Problems

1. Given an alphabet Σ, an instance of the *symbol find problem over* Σ consists of a string w over Σ together with a single symbol $x \in \Sigma$. The instance is *true* if and only if some symbol of the string w is equal to x. Prove that the symbol find problem over Σ is in Nick's Class.
2. Prove that every regular language is in Nick's Class.
3. Using the fact that CVP is $\mathcal{P}$-complete, prove that the programming language C++ is $\mathcal{P}$-complete.