# Communicating Sequential Processes
# Exercises 4: Abstraction and Communication
# Leila M. A. Silva

1. Process $P$, with alphabet $\{a, b\}$, is defined
   $P = (a \rightarrow b \rightarrow Q)[](b \rightarrow R)$
   $Q = (b \rightarrow P)[](a \rightarrow R)$
   $R = a \rightarrow Stop$ .
   Find $P \setminus a$ and shows using the CSP laws that it satisfies the guarded recursion
   $\mu X \bullet b \rightarrow (Stop \mid\sim\mid b \rightarrow X)$ .
2. If $P$ can never reach a deadlock state, does it follows that $P\backslash A$ is deterministic?
3. If $P = a?x \text{ -> } b!x \text{ -> } b!x \text{ -> } P$ then is it possible to find a process $Q$ such that
   $(P[\{a,b\}||\{b,c\}]Q)\backslash\{b\} =_T COPY$,
4. If $x$ is a positive approximation to the positive square root of t then the next approximation according to Newton's Rule is $(x + t/x )/2$. Suppose that after $n$ iterations the approximation is close enough. Write a pipe *Sqrt* that repeatedly inputs numbers on the left and outputs their square roots on the right. To achieve high speed, use at least $n$ processes, one for each.
5. The value of a punched card is a sequence of eight characters, which may be read as a single value along the left channel. Write a CSP process which reads cards and outputs their characters one at a time.
6. Construct a process *Fib* that outputs on its right channel the successive elements of the Fibonacci sequence; its alphabet is $\{right.n \mid n : \mathbf{N}\}$.
7. Write a process which behaves like a stack of messages. When empty, it responds to the signal *empty*. At all times it is ready to input a new message from the left and put it on top of the stack; and whenever nonempty, it is prepared to output and remove the top element of the stack.