Outline for January 11, 2001

1. Greetings and felicitations!
   a. First part of project due Friday
   b. Web page up and running!
2. Process models
   a. Theorem: If a system is mutually noninterfering, it is determinate.
   b. Theorem: Let f_p be an interpretation of process p. Let $\prod$ be a system of processes, with $p \in \prod$. If for all such p, $domain(p) \neq \emptyset$ and $range(p) \neq \emptyset$, but f_p unspecified, is determinate for all f_p, then all processes in $\prod$ are mutually noninterfering
   c. Maximally parallel system: determinate system for which the removal of any pair from the relation $\rightarrow$ makes the two processes in the pair interfering processes.
3. Critical section problem
   a. Mutual exclusion
   b. Progress
   c. Bounded wait
4. Classical problems
   a. Producer/consumer
   b. Readers/writers (first: readers priority; second: writers priority)
   c. Dining philosophers
5. Basic language constructs
   a. Semaphores
   b. Send/receive
6. Evaluating higher-level language constructs
   a. Modularity
   b. Constraints
   c. Expressive power
   d. Ease of use
   e. Portability
   f. Relationship with proram structure
   g. Process failures, unanticipated faults (exception handling)
   h. Real-time systems
7. Higher-level language constructs
   a. Monitors
   b. Crowd monitors
   c. Invariant expressions
   d. CSP
   e. RPC
   f. ADA™

Mutual Non-Interference and Determinism

Introduction

A determinate system of processes is a set of process that always produces the same output given the same input. A mutually non-interfering set of processes is a set of processes that do not interfere with the input or output of one another. The question is, to what degree are these the same concepts?

Formal Definitions and Notations

- A system of processes $S = (\prod, \rightarrow)$ is a set of processes $\prod = \{ p_1, \ldots, p_n \}$ and a precedence relation $\rightarrow: \prod \times \prod$. The $\rightarrow$ relation is a partial ordering (we define $p \rightarrow p$ as true). When $p \rightarrow q$, process p must complete before process q may begin.
- Each process $p \in \prod$ has an associated set of input memory locations called *domain*(p) and an associated set of output memory locations *range*(p) $\neq \varnothing$. An interpretation f_p of p associates values with each set of memory locations. The set of all inputs to S is abbreviated *domain*(S), and the set of all outputs from S is abbreviated *range*(S).
- Two systems of processes $S = (\prod, \rightarrow)$ and $S' = (\prod', \rightarrow')$ are equivalent if
  a. $\prod = \prod'$;
  b. $\rightarrow \neq \rightarrow'$; and
  c. if S and S' are given the same element of *domain*(S), then they output the same element of *range*(S).
- An execution sequence α is any string of process initiation and termination events satisfying the precedence constraints of the system.
- $V(M_i, \alpha)$ is the sequence of values written into memory location M_i at the termination of processes in α. The final value stored in M_i after execution sequence α completes is represented by $F(M_i, \alpha)$.
- A determinate system of processes is a system of processes S for which each element of *domain*(S) produces the same set *range*(S) regardless of the order or overlapping of the elements of S. More formally, a system S is determinate if, for any initial state and for all execution sequences α and α' of S, $V(M_i, \alpha) = V(M_i, \alpha')$
- A mutually noninterfering system of processes is a system of processes S in which all pairs of processes meet the Bernstein conditions. Processes p and q are noninterfering if either process is a predecessor of the other, or the processes satisfy the Bernstein conditions.
- The initiation of a process p is writtten $\overline{p}$, and the termination of p is written $\underline{p}$.

Relationship of Determinate Systems and Mutually Noninterfering Systems

Theorem 1: If a system is mutually non-interfering, it is determinate.
Theorem 2: Let S be a system with *domain*(p) and *range*(p) specified, *range*(p) $\neq \varnothing$, for all $p \in \prod$, and f_p unspecified. Then if S is determinate for all f_p, it is mutually non-interfering.

Proofs

The following lemma is helpful:
Lemma: Let S be a mutually noninterfering system. Let p be a terminal process of S. If $\alpha = \beta\overline{p}\gamma\underline{p}\delta$ is an execution sequence of S, then $\alpha' = \beta\gamma\delta\overline{p}\underline{p}$ is an execution sequence of S for which $V(M_i, \alpha) = V(M_i, \alpha')$ for all i.
Proof: As p is a terminal process in S, it has no successors in S. Hence α' satisfies the precedence constraints of S. So α' is an execution sequence. We now consider two cases.

1. $M_i \notin$ *range*(p). Note p does not write memory locations not in *range*(p). Consider any process p' with $\overline{p'}$ in δ. As p and p' are mutually noninterfering, *range*(p) $\cap$ *domain*(p') $= \varnothing$. So all such p' find the same values in *domain*(p') whether the execution sequence is α or α'. Thus, $V(M_i, \alpha) = V(M_i, \alpha')$.
2. $M_i \in$ *range*(p). Let $\overline{p'}$ in $\gamma\delta$. As p and p' are mutually noninterfering, *domain*(p) $\cap$ *range*(p') $= \varnothing$. So no p' in $\gamma\delta$ writes into an element of *domain*(p). Hence for all $M_j \in$ *domain*(p), $V(M_j, \beta) = V(M_j, \beta\gamma\delta)$. By definition, for all $M_j \in$ *domain*(p), $F(M_j, \beta) = F(M_j, \beta\gamma\delta)$. As p has the same input for both α and α', it writes the same value into

each $M_i \in$ *range*(p) in α and α'. Let v denote the value that p writes into M_i in α. Then

$V(M_i, \alpha) = V(M_i, \beta\overline{p}\gamma\underline{p}\delta)$ as no process p' in δ writes into an element of *range*(p)
$= (V(M_i, \beta\overline{p}\gamma), v)$ as p writes v into M_i
$= (V(M_i, \beta), v)$ as no process p' in γ writes into an element of *range*(p)
$= (V(M_i, b\gamma\delta), v)$ as no process p' in γ writes into an element of *range*(p)
$= V(M_i, b\gamma\delta\overline{p}\underline{p})$ as p writes v into M_i
$= V(M_i, \alpha')$

This proves the lemma. ■

Proof of Theorem 1: We proceed by induction on the number k of processes in a system.

Basis: $k = 1$. The claim is trivially true.

Hypothesis: For $k = 1, \ldots, n-1$, if a system of k processes is mutually noninterfering, it is determinate.

Step: Let S be an n process system of mutually noninterfering processes.

If S has exactly one execution sequence, it is determinate. So, assume that S has two distinct execution sequences α and β.

Let p be a terminal process of S, and form α' and β' according to the lemma. Then

$\alpha' = \alpha''\overline{p}\underline{p}$ $V(M_i, \alpha) = V(M_i, \alpha')$ for all i such that $1 \le i \le m$
$\beta' = \beta''\overline{p}\underline{p}$ $V(M_i, \beta) = V(M_i, \beta')$ for all i such that $1 \le i \le m$

Now form the n–1 process system $S' = (\prod - \{ p \}, \rightarrow')$, where →' is formed by deleting from → all pairs with p in them. Clearly, α'' and β'' are execution sequences of S'. Further, by the induction hypothesis, $V(M_i, \alpha'') = V(M_i, \beta'')$ for all i such that $1 \le i \le m$. This means that the values in the elements of *domain*(p) are the same in both α'' and β''; in other words, $F(M_j, \alpha'') = F(M_j, \beta'')$ for all $M_j \in$ *domain*(p). As the inputs for p are the same in both execution sequences, the outputs will also be the same. It follows that p writes the same value v into $M_i \in$ *range*(p) in both α' and β'.

Hence for $M_i \notin$ *range*(p):

$V(M_i, \alpha) = V(M_i, \alpha')$ by the lemma
$= V(M_i, \alpha'')$ as $M_i \notin$ *range*(p)
$= V(M_i, \beta'')$ by the induction hypothesis
$= V(M_i, \beta')$ as $M_i \notin$ *range*(p)
$= V(M_i, \beta)$ by the lemma

and for $M_i \in$ *range*(p):

$V(M_i, \alpha) = V(M_i, \alpha')$ by the lemma
$= (V(M_i, \alpha''), v)$ p writes v into M_i
$= (V(M_i, \beta''), v)$ by the induction hypothesis
$= (V(M_i, \beta'), v)$ p writes v into M_i
$= V(M_i, \beta)$ by the lemma

Either way, $V(M_i, \alpha) = V(M_i, \beta)$. Hence S is determinate, completing the induction step and the proof. ■

Proof of Theorem 2: We prove this theorem by contradiction. Let S be a determinate system. Let $p, p' \in \prod$ be interfering processes. Then there exist execution sequences

$\alpha = \beta\overline{p}\underline{p}\overline{p'}\underline{p'}\gamma$
$\alpha' = \beta\overline{p'}\underline{p'}\overline{p}\underline{p}\gamma$

Consider the Bernstein conditions. As p and p' are interfering, at least one of those conditions does not hold. We examine them separately.

1. Let $M_i \in$ *range*(p) ∩ *range*(p'). We choose the interpretation f_p so that p writes the value u into M_i, and we choose the interpretation $f_{p'}$ so that p' writes the value v into M_i, and $u \ne v$. But then
   $V(M_i, \beta\overline{p}\underline{p}\overline{p'}\underline{p'}) = (V(M_i, \beta), u, v)$
   and
   $V(M_i, \beta\overline{p'}\underline{p'}\overline{p}\underline{p}) = (V(M_i, \beta), v, u)$.
   This means S is not determinate, contradicting hypothesis. So *range*(p) ∩ *range*(p') = ∅.
2. Let $M_i \in$ *domain*(p) ∩ *range*(p'). As *range*(p) ≠ ∅, take $M_i \in$ *range*(p). Choose the interpretation $f_{p'}$ so that p

reads different values in α and α'; that is, $F(M_j, \beta) \neq F(M_j, \beta\overline{p'}p')$ for some j such that $1 \leq j \leq m$. Also, choose f_p so that p writes u in α and v in α', where $u \neq v$. But then

$$
\begin{aligned}
V(M_i, \beta\overline{p}\underline{p}\overline{p'}\underline{p'}) &= V(M_i, \beta\overline{p}\underline{p}) && \text{as } range(p) \cap range(p') = \varnothing \\
&= (V(M_i, \beta), u) \\
V(M_i, \beta\overline{p'}\underline{p'}\overline{p}\underline{p}) &= (V(M_i, \beta\overline{p'}\underline{p'}), v) \\
&= (V(M_i, \beta), v) && \text{as } range(p) \cap range(p') = \varnothing
\end{aligned}
$$

As $u \neq v$, this means that S is not determinate, contradicting hypothesis. So $domain(p) \cap range(p') = \emptyset$. [As an aside, if $range(p) = \emptyset$, then $M_i \notin range(p)$ and p and p' are noninterfering. Hence there is no contradiction.]

3. By symmetry, the argument for case 2 also shows that $range(p) \cap domain(p') = \emptyset$.

In all three cases, the Bernstein conditions must hold. This completes the proof. ■