Sentencias Concurrentes

- Ahora es necesario, dado un programa concurrente, saber que secciones del código son concurrentes y cuáles no, además es indispensable especificarlo en un lenguaje de programación.

- No todas las sentencias son concurrentes, consideremos el siguiente fragmento del programa:
  - S1: x:=x+1
  - S2: y:=x+2

  - En este caso las instrucciones no pueden ejecutarse de forma independiente.

Sentencias Concurrentes

- Consideremos ahora:
  - x:=1
  - y:=2
  - z:=3

- Cada una de las sentencias se pueden ejecutar concurrentemente puesto que el orden en que se ejecuten no afecta el resultado final.

- Si se tuvieran 3 procesadores en cada uno se colocaría cada instrucción.
- Aunque la intuición nos indique cuando ejecutar concurrentemente Bernstein definió condiciones para garantizar la concurrencia.

Condiciones de Bernstein

- Para determinar si dos conjuntos de instrucciones se pueden ejecutar de forma concurrente se definen:

  - $L(S_k)=\{a_1,a_2,\ldots,a_n\}$ **conjunto de lectura** del conjunto de instrucciones S_k, formado por todas las variables cuyos valores son leídos (referenciados) durante la ejecución de las instrucciones en S_k.

  - $E(S_k) =\{b_1,b_2,\ldots,b_n\}$ **conjunto de escritura** del conjunto de instrucciones S_k, formado por todas las variables cuyos valores son actualizados (se escriben) durante la ejecución de las instrucciones en S_k.

Condiciones de Bernstein

- Para que dos conjuntos de instrucciones S_i y S_j, $i \neq j$, $i < j$ se puedan ejecutar concurrentemente se tiene que cumplir que:

1. $L(S_i) \cap E(S_j) = \emptyset$
2. $E(S_i) \cap L(S_j) = \emptyset$
3. $E(S_i) \cap E(S_j) = \emptyset$

Ejemplo de Condiciones de Bernstein

- Sean:

$S_1 \rightarrow a:=x+y;$

$S_2 \rightarrow b:=z-1;$

$S_3 \rightarrow c:=a-b;$

$S_4 \rightarrow w:=c+1;$

Se calculan los conjuntos de lectura y escritura

- $L(S_1)=\{x,y\}$
  $E(S_1)=\{a\}$
- $L(S_2)=\{z\}$
  $E(S_2)=\{b\}$
- $L(S_3)=\{a,b\}$
  $E(S_3)=\{c\}$
- $L(S_4)=\{c\}$
  $E(S_4)=\{w\}$

Ejemplo de Condiciones de Bernstein

- Sean:

$L(S_1)=\{x,y\}$
$E(S_1)=\{a\}$

$L(S_2)=\{z\}$
$E(S_2)=\{b\}$

$L(S_3)=\{a,b\}$
$E(S_3)=\{c\}$

$L(S_4)=\{c\}$
$E(S_4)=\{w\}$

2. Aplicando las condiciones de Bernstein

Entre S_1 y S_2

1. $L(S_1) \cap E(S_2)= \varnothing$
2. $E(S_1) \cap L(S_2)= \varnothing$
3. $E(S_1) \cap E(S_2)= \varnothing$

Entre S_1 y S_3

1. $L(S_1) \cap E(S_3)= \varnothing$
2. $E(S_1) \cap L(S_3)=\{ a \} \neq\varnothing$
3. $E(S_1) \cap E(S_3)= \varnothing$

Entre S_1 y S_4

1. $L(S_1) \cap E(S_4)= \varnothing$
2. $E(S_1) \cap L(S_4)= \varnothing$
3. $E(S_1) \cap E(S_4)= \varnothing$

Ejemplo de Condiciones de Bernstein

- Sean:

$L(S_1)=\{x,y\}$

$E(S_1)=\{a\}$

$L(S_2)=\{z\}$

$E(S_2)=\{b\}$

$L(S_3)=\{a,b\}$

$E(S_3)=\{c\}$

$L(S_4)=\{c\}$

$E(S_4)=\{w\}$

2. Aplicando las condiciones de Bernstein

Entre S_2 y S_3

1. $L(S_2) \cap E(S_3)= \varnothing$
2. $E(S_2) \cap L(S_3)=\{ b \} \neq \varnothing$
3. $E(S_2) \cap E(S_3)= \varnothing$

Entre S_2 y S_4

1. $L(S_2) \cap E(S_4)= \varnothing$
2. $E(S_2) \cap L(S_4)=\varnothing$
3. $E(S_2) \cap E(S_4)= \varnothing$

Entre S_3 y S_4

1. $L(S_3) \cap E(S_4)= \varnothing$
2. $E(S_3) \cap L(S_4)=\{ c \} \neq \varnothing$
3. $E(S_3) \cap E(S_4)= \varnothing$

Tabla resultante al aplicar las condiciones de Bernstein

- En este caso se indica que sentencias se pueden ejecutar concurrentemente y cuales no

	S_1	**S_2**	**S_3**	**S_4**
S_1	----	**Si**	**No**	**Si**
S_2	----	----	**No**	**Si**
S_3	----	----	----	**No**
S_4	----	----	----	----