

VERY BUSI EXPRESSIONS

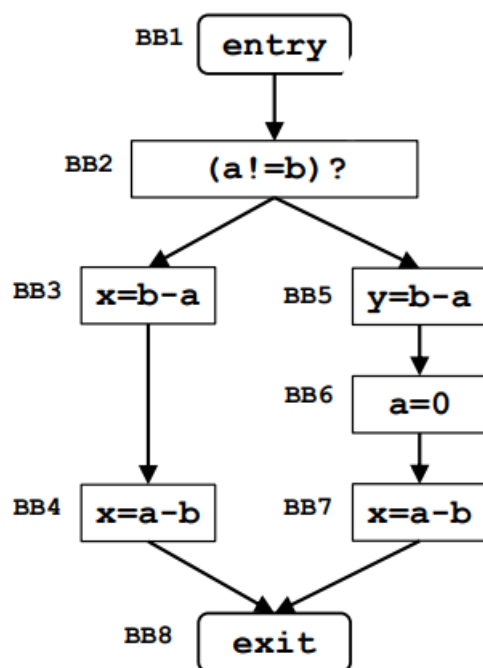
SCHEMA DATAFLOW ANALYSIS

Dominio	Espressioni
Direzione	Indietro
Funzione di trasferimento	$UseB \cup (x - DefB)$
Operatore d'incontro	Intersezione
Condizioni ai bordi	$in[exit] = 0$
Equazioni	$in[B] = f(out[B])$ $out[B] = \wedge in[S] \text{ } (\forall \text{ successore } S \text{ di } B)$
Condizione iniziale	$in[B] = U \text{ (insieme Universo = \#espressioni)}$

RAPPRESENTAZIONE MEDIANTE VETTORI DI BIT

- **DOMINIO**: bit vector contenente tante cifre quante sono le espressioni valutate. In particolare, se l'elemento i-esimo è pari a 1, allora l'espressione associata ad i è very busy, altrimenti il viceversa.
- **UseB**: bit vector di dimensioni analoghe a quello di dominio. In particolare, se l'elemento i-esimo è pari a 1, allora l'espressione associata ad i è utilizzata nel blocco B, altrimenti il viceversa.
- **DefB**: bit vector di dimensioni analoghe a quello di dominio. In particolare, se l'elemento i-esimo è pari a 1, allora il valore di un operando dell'espressione associata ad i è cambiata nel blocco B, altrimenti il viceversa.

ESEMPIO



BIT VECTOR: $\langle a \neq b, b - a, a - b \rangle$

	Iterazione 1: inizializzazione		Iterazione 2	
	IN[B]	OUT[B]	IN[B]	OUT[B]
BB8	0	0	000	
BB7	0	0	001	000
BB6	0	0	000	001

BB5	0	0	010	000
BB4	0	0	001	000
BB3	0	0	011	001
BB2	0	0	110	010
BB1	0	0		

INSIEMI USE E DEF

	USE	DEF
BB8	000	000
BB7	001	000
BB6	000	111
BB5	010	000
BB4	001	000
BB3	010	000
BB2	100	000
BB1	000	000

DOMINATOR ANALYSIS

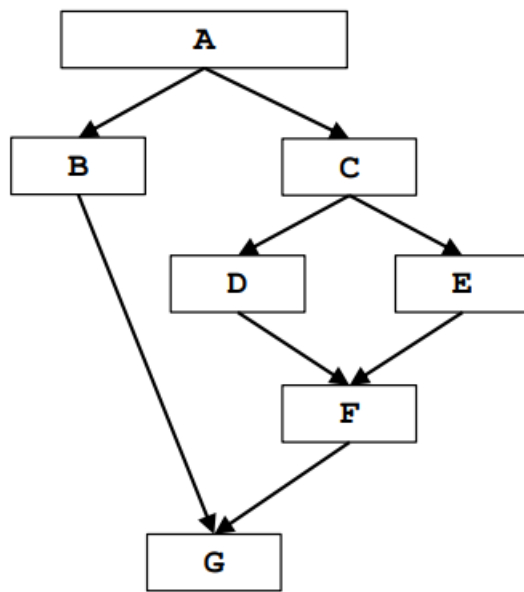
SCHEMA DATAFLOW ANALYSIS

Dominio	Basic Blocks
Direzione	Avanti
Funzione di trasferimento	Aggiunta Basic Block B all'insieme dei nodi d'ingresso. $x \cup \{ B \}$
Operatore d'incontro	Intersezione
Condizioni ai bordi	$out[entry] = \{ entry \}$
Equazioni	$out[B] = f(in[B])$ $in[B] = \bigwedge out[P] \ (\forall \text{ predecessore } P \text{ di } B)$
Condizione iniziale	$out[B] = U \ (\text{insieme Universo} = \# \text{nodi})$

RAPPRESENTAZIONE MEDIANTE VETTORI DI BIT

- DOMINIO: bit vector contenente tante cifre quanti sono i Basic Block del flow graph. Se l'elemento i-esimo è pari a 1, allora indica che il Basic Block i-esimo domina il Basic Block corrente, altrimenti viceversa
- Il valore x nella funzione di trasferimento indica il Basic Block corrente. In particolare, rappresenta un bit vector contenente tante cifre quanti sono i Basic Block ed è usato per indicare l'auto dominanza di un blocco.

ESEMPIO



$DOM[F] = \{A, C, F\}$

BIT VECTOR: <A, B, C, D, E, F, G>

	Iterazione 1: inizializzazione		Iterazione 2	
	IN[B]	OUT[B]	IN[B]	OUT[B]
A	0	1000000	0	1000000

B	0	1111111	1000000	1100000
C	0	1111111	1000000	1010000
D	0	1111111	1010000	1011000
E	0	1111111	1010000	1010100
F	0	1111111	1010000	1010010
G	0	1111111	1000000	1000001

INSIEMI DI DOMINANZA

	Dominatori
A	{A} = 1000000
B	{A, B} = 1100000
C	{A, C} = 1010000
D	{A, C, D} = 1011000
E	{A, C, E} = 1010100
F	{A, C, F} = 1010010
G	{A, G} = 1000001

CONSTAN PROPAGATION

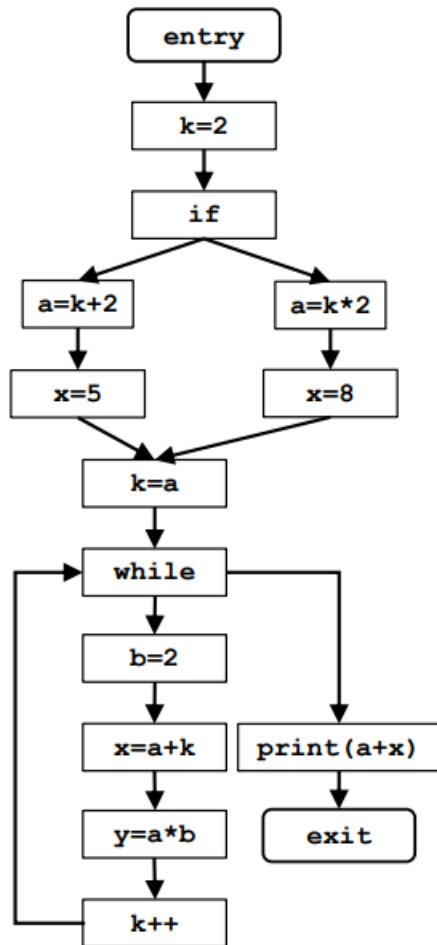
SCHEMA DATA FLOW ANALYSIS

Dominio	Variabili
Direzione	Avanti
Funzione di trasferimento	$(GenB \cup in[B]) - KillB$
Operatore d'incontro	Intersezione
Condizioni ai bordi	$out[entry] = 0$
Equazioni	$out[B] = f(in[B])$ $in[B] = \bigwedge out[P] \text{ } (\forall \text{ predecessore } P \text{ di } B)$
Condizione iniziale	$out[B] = U \text{ (insieme Universo = \#variabili)}$

ULTERIORI CONSIDERAZIONI

- **DOMINIO**: insieme contenente tante coppie <nome variabile, valore> quante sono le diverse variabili definite nel flow graph.
- *GenB*: insieme contenente le coppie <nome variabile, *> (dove * indica un qualsiasi valore dell'insieme Universo).
- *KillB*: insieme delle variabili definite come non costanti nel blocco.
- *Insieme U* = $\{(k,*), (a,*), (x,*), (b,*), (y,*)\}$

ESEMPIO



	Iterazione 1: inizializzazione		Iterazione 2	
	IN[B]	OUT[B]	IN[B]	OUT[B]
Entry	-	{}	-	{}
BB1	-	U	{}	{{(k, 2)}}
BB2	-	U	{{(k, 2)}}	{{(k, 2)}}
BB3	-	U	{{(k, 2)}}	{{(k, 2), (a, 4)}}
BB4	-	U	{{(k, 2), (a, 4)}}	{{(k, 2), (a, 4), (x, 5)}}
BB5	-	U	{{(k, 2)}}	{{(k, 2), (a, 4)}}
BB6	-	U	{{(k,2), (a, 4)}}	{{(k, 2), (a, 4), (x, 8)}}
BB7	-	U	{{(k, 2), (a, 4)}}	{{(k, 4), (a, 4)}}
BB8	-	U	{{(k, 4), (a, 4)}}	{{(k, 4), (a, 4)}}
BB9	-	U	{{(k, 4), (a, 4)}}	{{(k, 4), (a, 4), (b, 2)}}
BB10	-	U	{{(k, 4), (a, 4), (b, 2)}}	{{(k, 4), (a, 4), (b, 2), (x, 8)}}
BB11	-	U	{{(k, 4), (a, 4), (b, 2), (x, 8)}}	{{(k, 4), (a, 4), (b, 2), (x, 8), (y, 8)}}
BB12	-	U	{{(k, 4), (a, 4), (b, 2), (x, 8), (y, 8)}}	{{(k, 5), (a, 4), (b, 2), (x, 8), (y, 8)}}
BB13	-	U	{{(k, 4), (a, 4)}}	{{(k, 4), (a, 4)}}
Exit	-	U	{{(k, 4), (a, 4)}}	{{(k, 4), (a, 4)}}

	Iterazione 3	Iterazione 4
--	--------------	--------------

	IN[B]	OUT[B]	IN[B]	OUT[B]
Entry	-	{}		{}
BB1	{}	{(k, 2)}	{}	{(k, 2)}
BB2	{(k, 2)}	{(k, 2)}	{(k, 2)}	{(k, 2)}
BB3	{(k, 2)}	{(k, 2), (a, 4)}	{(k, 2)}	{(k, 2), (a, 4)}
BB4	{(k, 2), (a, 4)}	{(k, 2), (a, 4), (x, 5)}	{(k, 2), (a, 4)}	{(k, 2), (a, 4), (x, 5)}
BB5	{(k, 2)}	{(k, 2), (a, 4)}	{(k, 2)}	{(k, 2), (a, 4)}
BB6	{(k, 2), (a, 4)}	{(k, 2), (a, 4), (x, 8)}	{(k, 2), (a, 4)}	{(k, 2), (a, 4), (x, 8)}
BB7	{(k, 2), (a, 4)}	{(k, 4), (a, 4)}	{(k, 2), (a, 4)}	{(k, 4), (a, 4)}
BB8	{(k, 4), (a, 4)}	{(a, 4)} **	{(k, 4), (a, 4)}	{(a, 4)}
BB9	{(a, 4)}	{(a, 4), (b, 2)}	{(a, 4)}	{(a, 4), (b, 2)}
BB10	{(a, 4), (b, 2)}	{(a, 4), (b, 2)} ***	{(a, 4), (b, 2)}	{(a, 4), (b, 2)}
BB11	{(a, 4), (b, 2)}	{(a, 4), (b, 2), (y, 8)}	{(a, 4), (b, 2)}	{(a, 4), (b, 2), (y, 8)}
BB12	{(a, 4), (b, 2), (y, 8)}	{(a, 4), (b, 2), (y, 8)}	{(a, 4), (b, 2), (y, 8)}	{(a, 4), (b, 2), (y, 8)}
BB13	{(a, 4)}	{(a, 4)}	{(a, 4)}	{(a, 4)}
exit	{(a, 4)}	{(a, 4)}	{(a, 4)}	{(a, 4)}

** $\{(k, 4), (a, 4)\} \cap \{(k, 5), (a, 4), (b, 2), (x, 8), (y, 8)\}$ (dal blocco 12 della iterazione 2)

*** (x, 8) non è una coppia in output, perché seppur x sia stata definita nel Basic Block 8 il suo assegnamento coinvolge la variabile k che non è più costante in questa iterazione.

L'iterazione 4 porta alla convergenza dell'algoritmo: gli insiemi IN[B] e OUT[B] non cambiano.

INSIEMI GEN E KILL

	GEN	KILL
Entry	{}	{}
BB1	{(k, 2)}	{}
BB2	{}	{}
BB3	{(a, k+2)}	{}
BB4	{(x, 5)}	{}
BB5	{(a, k*2)}	{}
BB6	{(x, 8)}	{}
BB7	{(k, a)}	{(k, 2)}
BB8	{}	{}
BB9	{(b, 2)}	{}
BB10	{(x, a+k)}	{(x, 5), (x, 8)}
BB11	{(y, a*b)}	{}
BB12	{(k, a+1)}	{(k, a)}
BB13	{}	{}
exit	{}	{}