

Assignment 2 – Iannone Raimondo

1) Very Busy Expressions

Teoria: La VBE è un'espressione **che sarà sicuramente usata prima che uno dei suoi operandi venga ridefinito**, quindi in un punto p, se un'istruzione a+b è valutata in tutti percorsi, da p a exit e non c'è una ridefinizione di a o b lungo il percorso, allora vuol dire che quell'espressione è **very busy**.

Nella sezione **Direction** applichiamo una logica in **backward**, perché dipende dai **futuri usi dell'espressione**. Dobbiamo partire dai blocchi di **uscita** e risalire il CFG per sapere se in tutti i cammini l'espressione verrà usata prima che venga invalidata.

	Dataflow Problem
Domain	Sets of expression
Direction	Backward $in[b] = f_b(out[b])$ $out[b] = \cap in[succ(b)]$
Transfer function	$f_b(x) = Use_b \cup (x - Def_b)$
Meet Operation	\cap
Boundary Condition	$in[exit] = \emptyset$
Initial interior points	$in[b] = u$

	Iteration 1	
	$in[b]$	$out[b]$
BB1	\emptyset	b-a
BB2	b-a	b-a
BB3	b-a, a-b	a-b
BB4	a-b	\emptyset
BB5	b-a	\emptyset
BB6	\emptyset	a-b
BB7	a-b	\emptyset
BB8	\emptyset	\emptyset

2) Dominator Analysis

Teoria: La DA, in un CFG diciamo che nodo X **domina** Y se il nodo X appare in ogni percorso del grafo che porta dal blocco ENTRY al blocco Y.

Usiamo in questo caso invece **forward**, perché per sapere se un nodo D domina un nodo N, devi guardare tutti i **cammini che partono da entry fino a N**.

	Dataflow Problem
Domain	Sets of basic blocks
Direction	Forward $out[b] = f_b(in[b])$ $in[b] = \cap out[prec(b)]$
Transfer function	$f_b(x) = B \cup x$
Meet Operation	\cap
Boundary Condition	$out[entry] = \emptyset$
Initial interior points	$out[b] = u$

	Iteration 1	
	$in[b]$	$out[b]$
A	\emptyset	A
B	A	A, B
C	A	A, C
D	A, D	A, C, D
E	A, E	A, C, E
F	A, F	A, C, F
G	A	A, G

3) Constant Propagation

Teoria: L'obiettivo della constant propagation è quello di determinare in quali punti del programma le variabili hanno un valore costante. In questo caso, l'informazione da calcolare per ogni nodo n del CFG è un insieme di coppie del tipo "variabile, valore costante", che metteremo nel dataflow

Anche in questo caso usiamo come direction Forward, per lo stesso motivo del punto precedente (Dominator Analysis).

	Dataflow Problem
Domain	Sets of variables and constant values
Direction	Forward $out[b] = f_b(in[b])$ $in[b] = \cap out[prec(b)]$
Transfer function	$f_b(x) = Def_b \cup (x - Kill_b)$
Meet Operation	\cap
Boundary Condition	$out[entry] = \emptyset$
Initial interior points	$out[b] = u$

	Iteration 1		Iteration 2		Iteration 3	
	$in[b]$	$out[b]$	$in[b]$	$out[b]$	$in[b]$	$out[b]$
Entry	\emptyset	\emptyset	X	X	X	X
K=2	\emptyset	K=2	X	X	X	X
IF	K=2	K=2	X	X	X	X
A=K+2	K=2	A=4,K=2	X	X	X	X
X=5	A=4,K=2	X=5,A=4,K=2	X	X	X	X
A=K*2	K=2	K=2,A=4	X	X	X	X
X=8	K=2,A=4	K=2,A=4,X=8	X	X	X	X
K=A	K=2,A=4	K=4,A=4	X	X	X	X
While	K=4,A=4	K=4,A=4	A=4	A=4	A=4	A=4
B=2	K=4,A=4	K=4,A=4,B=2	A=4	A=4,B=2	A=4	A=4,B=2
X=A+K	K=4,A=4, B=2	K=4,A=4,B=2, X=8	A=4, B=2	A=4, B=2	A=4, B=2	A=4, B=2
Y=A*B	K=4,A=4,B=2, X=8	K=4,A=4,B=2, X=8, Y=8	A=4, B=2	A=4, B=2, Y=8	A=4, B=2	A=4, B=2, Y=8
K++	K=4,A=4,B=2, X=8	K=5,A=4,B=2, X=8	A=4, B=2, Y=8	A=4, B=2, Y=8	A=4, B=2, Y=8	A=4, B=2, Y=8
Print(a+x)	K=4, A=4	K=4,A=4	A=4	A=4	A=4	A=4
Exit	X	X	X	X	X	X

Ci fermiamo alla terza iterazione, visto che il loop while, si ripete con le stesse costanti.