Secondo Assignment di Compilatori

D'Alterio Dario
Matricola 176689

Patrini Andrea
Matricola 176907

Shaukat Arslan
Matricola 176687

April 14, 2025

Contents

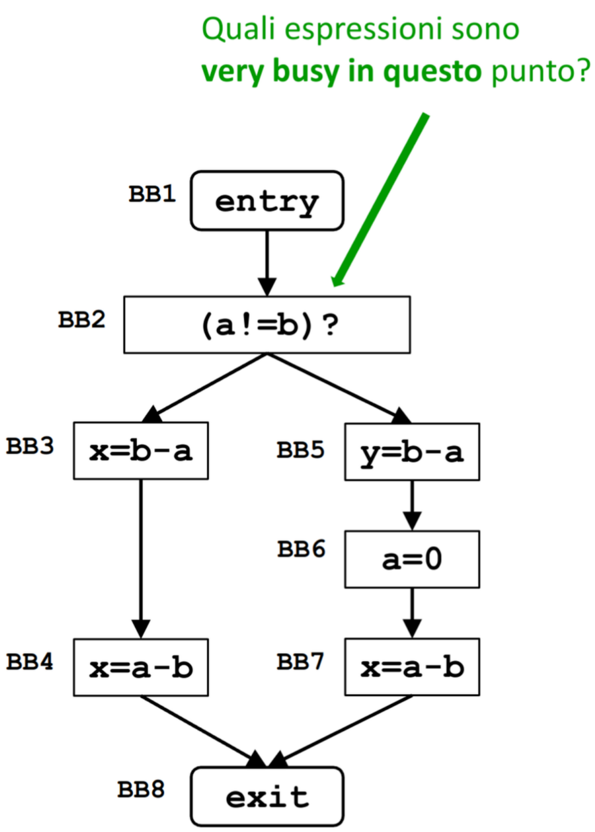
1 Very Busy Expression

1.1 DFA Framework

- Un'espressione è **very busy** in un punto p se, indipendentemente dal percorso preso da p, l'espressione viene usata prima che uno dei suoi operandi venga definito.
- Un'espressione $a + b$ è **very busy** in un punto p se $a + b$ è valutata in tutti i percorsi da p a $EXIT$ e non c'è una definizione di a o b lungo tali percorsi

Domain	Sets of Expressions
Direction	Backward • in[b] $= f_b(out[b])$ • out[b] $= \wedge$ in[succ(b)]
Transfer function	$f_b = Gen_b \cup (x \setminus Kill_b)$
Meet operation ($\wedge$)	($\cap$)
Boundary condition	in[exit] $= \varnothing$
Initial interior	$in[b] = \mathbb{U}$ per tutti i blocchi tranne EXIT

1.2 Iterazioni dell'Algoritmo Iterativo



	Iterazione 1		Iterazione 2	
Blocco	**IN[B]**	**OUT[B]**	**IN[B]**	**OUT[B]**
BB1	$\{b - a\}$	$\{b - a\}$	$\{b - a\}$	$\{b - a\}$
BB2	$\{b - a\}$	$\{b - a\}$	$\{b - a\}$	$\{b - a\}$
BB3	$\{a - b,\ b - a\}$	$\{a - b\}$	$\{a - b,\ b - a\}$	$\{a - b\}$
BB4	$\{a - b\}$	$\varnothing$	$\{a - b\}$	$\varnothing$
BB5	$\{b - a\}$	$\varnothing$	$\{b - a\}$	$\varnothing$
BB6	$\varnothing$	$\{a - b\}$	$\varnothing$	$\{a - b\}$
BB7	$\{a - b\}$	$\varnothing$	$\{a - b\}$	$\varnothing$
BB8	$\varnothing$	$\varnothing$	$\varnothing$	$\varnothing$

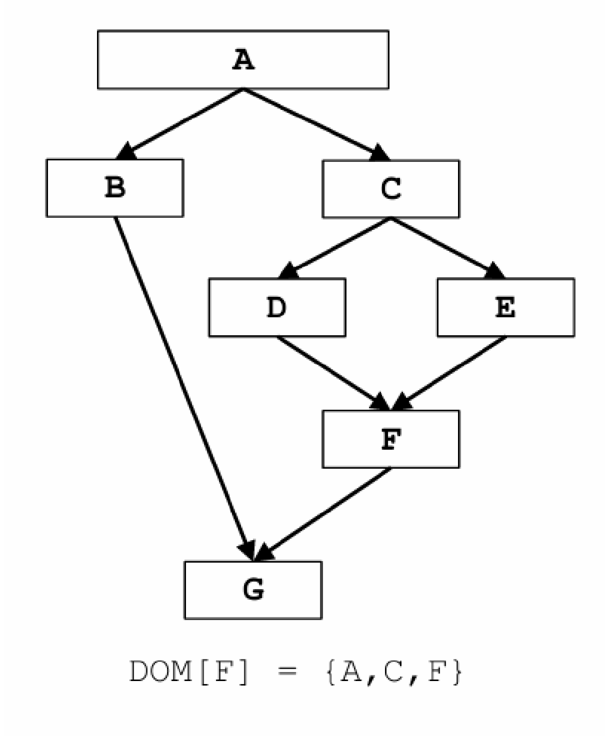
L'unica espressione very busy è $b - a$ che possiamo precalcolare

2 Dominator Analysis

In un **CFG** diciamo che un nodo X domina un altro nodo Y se il nodo X appare in ogni percorso del grafo che porta dal blocco $ENTRY$ al blocco Y.

Annotiamo ogni **basic block** B_i con un insieme $DOM[B_i]$:

- $B_i \in DOM[B_j]$ se e solo se B_i domina B_j
- Per definizione, un nodo domina se stesso: $B_i \in DOM[B_i]$



2.1 DFA Framework

Domain	Insieme dei Basick Blocks
Direction	Forward: $out[b] = f_b(in[b])$ $in[b] = \wedge\ out[pred(b)]$
Transfer function	$f_b(x) = \{b\} \cup x$
Meet operation ($\wedge$)	$\cap$
Boundary condition	$out[entry] = \{entry\}$
Initial interior	$out[b] = \mathbb{U}$

2.2 Iterazioni dell'Algoritmo Iterativo

	Iterazione 1		Iterazione 2	
	IN[B]	**OUT[B]**	**IN[B]**	**OUT[B]**
A	$\emptyset$	{A}	$\emptyset$	{A}
B	{A}	{A, B}	{A}	{A, B}
C	{A}	{A, C}	{A}	{A, C}
D	{A, C}	{A, C, D}	{A, C}	{A, C, D}
E	{A, C}	{A, C, E}	{A, C}	{A, C, E}
F	{A, C}	{A, C, F}	{A, C}	{A, C, F}
G	{A}	{A, G}	{A}	{A, G}

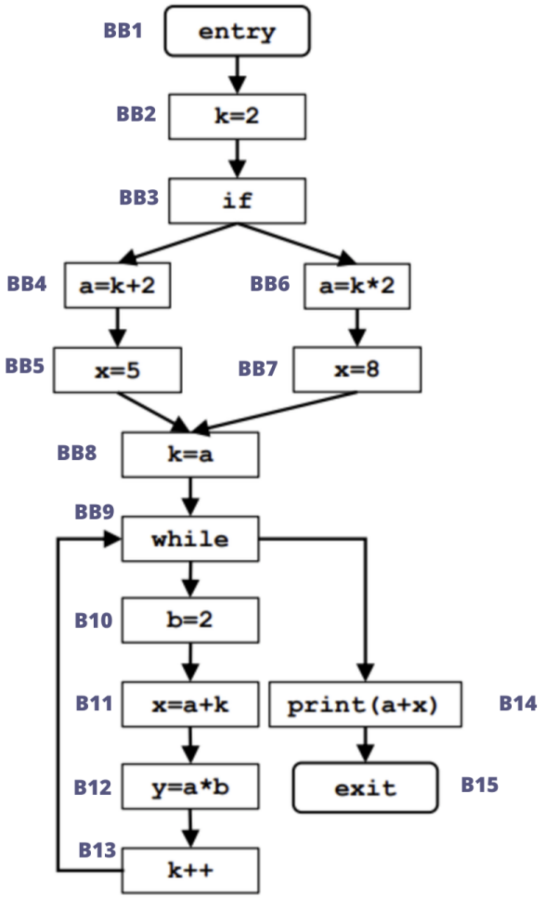
Notiamo che la soluzione converge alla seconda iterazione poichè gli insiemi out non hanno subito variazioni.

3 Constant Propagation

L'obiettivo della **constant propagation** è quello di determinare in quali punti del programma le variabili hanno un valore costante.

L'informazione da calcolare per ogni nodo ndel **CFG** è un insieme di coppie del tipo <**variabile, valore costante**>.

Se abbiamo la coppia <**x, c**> al nodo n, significa che x è garantito avere il valore c ogni volta che n viene raggiunto durante l'esecuzione del programma.



3.1 DFA Framework

Domain	Insieme di coppie $(var, const)$
Direction	Forward: $out[b] = f_b(in[b])$ $in[b] = \bigwedge out[pred(b)]$
Transfer function	$f_b(x) = Gen(b) \cup (\ \mathtt{x} - Kill_b)\ )$
Meet operation	Intersezione $(\cap)$
Boundary condition	$out[entry] = \emptyset$
Initial interior	$out[b] = \mathbb{U}$

3.2 Iterazioni dell'Algoritmo Iterativo

	Iterazione 1		Iterazione 2	
	IN[B]	**OUT[B]**	**IN[B]**	**OUT[B]**
BB1	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
BB2	$\emptyset$	{(k,2)}	$\emptyset$	{(k,2)}
BB3	{(k,2)}	{(k,2)}	{(k,2)}	{(k,2)}
BB4	{(k,2)}	{(k,2),(a,4)}	{(k,2)}	{(k,2),(a,4)}
BB5	{(k,2),(a,4)}	{(k,2),(a,4),(x,5)}	{(k,2),(a,4)}	{(k,2),(a,4),(x,5)}
BB6	{(k,2)}	{(k,2),(a,4)}	{(k,2)}	{(k,2),(a,4)}
BB7	{(k,2),(a,4)}	{(k,2),(a,4),(x,8)}	{(k,2),(a,4)}	{(k,2),(a,4),(x,8)}
BB8	{(k,2),(a,4)}	{(k,4),(a,4)}	{(k,2),(a,4)}	{(k,4),(a,4)}
BB9	{(k,4),(a,4)}	{(k,4),(a,4)}	{(a,4)}	{(a,4)}
BB10	{(k,4),(a,4)}	{(k,4),(a,4),(b,2)}	{(a,4)}	{(a,4),(b,2)}
BB11	{(k,4),(a,4),(b,2)}	{(k,4),(a,4),(b,2),(x,8)}	{(a,4),(b,2)}	{(a,4),(b,2)}
BB12	{(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)}
BB13	{(k,4),(a,4),(b,2),(x,8),(y,8)}	{(k,5),(a,4),(b,2),(x,8),(y,8)}	{(a,4),(b,2),(y,8)}	{(a,4),(b,2),(y,8)}
BB14	{(k,4),(a,4)}	{(k,4),(a,4)}	{(a,4)}	{(a,4)}
BB15	{(k,4),(a,4)}	{(k,4),(a,4)}	{(a,4)}	{(a,4)}

	Iterazione 3	
	IN[B]	**OUT[B]**
BB1	$\emptyset$	$\emptyset$
BB2	$\emptyset$	{(k,2)}
BB3	{(k,2)}	{(k,2)}
BB4	{(k,2)}	{(k,2),(a,4)}
BB5	{(k,2),(a,4)}	{(k,2),(a,4),(x,5)}
BB6	{(k,2)}	{(k,2),(a,4)}
BB7	{(k,2),(a,4)}	{(k,2),(a,4),(x,8)}
BB8	{(k,2),(a,4)}	{(k,4),(a,4)}
BB9	{(a,4)}	{(a,4)}
BB10	{(a,4)}	{(a,4),(b,2)}
BB11	{(a,4),(b,2)}	{(a,4),(b,2)}
BB12	{(a,4),(b,2)}	{(a,4),(b,2),(y,8)}
BB13	{(a,4),(b,2),(y,8)}	{(a,4),(b,2),(y,8)}
BB14	{(a,4)}	{(a,4)}
BB15	{(a,4)}	{(a,4)}