Linguaggi e compilatori: Assignment 2

Davide De Soricellis
Roberto Wang
Jun Wu Wang

Maggio 2024

Contents

1 Very busy expressions

1.1 Dataflow analysis framework

	Very busy expressions
Domain	Sets of expressions
Direction	backward: $in[b] = f_b(out[b])$ $out[b] = \wedge in[succ(b)]$
Transfer function	$f_b(x) = Use_b \cup (x - Def_b)$
Meet operation ($\wedge$)	$\cap$
Boundary condition	$in[Exit] = \emptyset$
Initial interior points	$in[b] = U$

Transfer function:

$$in[b] = f_b(out[b]) = Use[b] \cup (out[b] - Def[b])$$

- $Use[b]$: very busy expressions generate (usi localmente esposti in un BB → usi di espressioni che non sono preceduti nel BB da definizioni dei loro operandi)

- $out[b] - Def[b]$: very busy expressions propagate

    - $out[b]$: very busy expressions ereditate
    - $Def[b]$: very busy expressions che hanno operandi definiti nel BB

1.2 Iterative table

	Iterazione 1		Iterazione 2	
	$IN[B]$	$OUT[B]$	$IN[B]$	$OUT[B]$
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	$\{a - b\}$	$\emptyset$	$\{a - b\}$	$\emptyset$
BB5	$\{b - a\}$	$\emptyset$	$\{b - a\}$	$\emptyset$
BB6	$\emptyset$	$\{a - b\}$	$\emptyset$	$\{a - b\}$
BB7	$\{a - b\}$	$\emptyset$	$\{a - b\}$	$\emptyset$
BB8	$\emptyset$	undefined	$\emptyset$	undefined

La tabella è generata eseguendo un algoritmo iterativo naive. Con un algoritmo worklist è possibile eseguire in 1 iterazione.

2 Dominator analysis

2.1 Dataflow analysis framework

	Dominator analysis
Domain	Sets of basic blocks
Direction	forward: $out[b] = f_b(in[b])$ $in[b] = \wedge out[pred(b)]$
Transfer function	$f_b(x) = Gen_b \cup x$
Meet operation ($\wedge$)	$\cap$
Boundary condition	$out[Entry] = Entry$
Initial interior points	$out[b] = U$

Transfer function:
$$out[b] = f_b(in[b]) = Gen[b] \cup in[b]$$

- $Gen[b]$: dominators generati $\rightarrow$ sè stesso (b)

- $in[b]$: dominators ereditati/propagati

2.2 Iterative table

	Iterazione 1		Iterazione 2	
	$IN[B]$	$OUT[B]$	$IN[B]$	$OUT[B]$
A	undefined	$\{A\}$	undefined	$\{A\}$
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	$\{A\}$	$\{A, G\}$	$\{A\}$	$\{A, G\}$

La tabella è generata eseguendo un algoritmo iterativo naive. Con un algoritmo worklist è possibile eseguire in 1 iterazione.

3 Constant propagation

3.1 Dataflow analysis framework

	Constant propagation
Domain	Sets of <variable, constant value> pairs
Direction	forward: $out[b] = f_b(in[b])$ $in[b] = \wedge out[pred(b)]$
Transfer function	$f_b(x) = Gen_b \cup (x - Kill_b)$
Meet operation ($\wedge$)	$\cap$
Boundary condition	$out[Entry] = \emptyset$
Initial interior points	$out[b] = U$

Transfer function:

$$out[b] = f_b(in[b]) = Gen[b] \cup (in[b] - Kill[b])$$

- $Gen[b]$: coppie <variable, constant value> generate (definizioni localmente disponibili di variabili con un valore costante $\rightarrow$ l'ultima definizione di una variabile in b e con rhs un valore costante)

- $in[b] - Kill[b]$: coppie <variable, constant value> propagate

    - $in[b]$: coppie <variable, constant value> ereditate
    - $Kill[b]$: coppie <variable, constant value> uccise (ridefinizioni di variabili)

3.2 Iterative table

		Iterazione 1	
	$IN[B]$		$OUT[B]$
BB1	undefined		$\emptyset$
BB2	$\emptyset$		$\{< k, 2 >\}$
BB3 (if)	$\{< k, 2 >\}$		$\{< k, 2 >\}$
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 (while)	$\{< k, 4 >, < a, 4 >\}$		$\{< k, 4 >, < a, 4 >\}$
BB10	$\{< k, 4 >, < a, 4 >\}$		$\{< k, 4 >, < a, 4 >, < b, 2 >\}$
BB11	$\{< k, 4 >, < a, 4 >, < b, 2 >\}$		$\{< k, 4 >, < a, 4 >, < b, 2 >, < x, 8 >\}$
BB12	$\{< k, 4 >, < a, 4 >, < b, 2 >, < x, 8 >\}$		$\{< k, 4 >, < a, 4 >, < b, 2 >, < x, 8 >, < y, 8 >\}$
BB13	$\{< k, 4 >, < a, 4 >, < b, 2 >, < x, 8 >, < y, 8 >\}$		$\{< k, 5 >, < a, 4 >, < b, 2 >, < x, 8 >, < y, 8 >\}$
BB14	$\{< k, 4 >, < a, 4 >\}$		$\{< k, 4 >, < a, 4 >\}$
BB15	$\{< k, 4 >, < a, 4 >\}$		$\{< k, 4 >, < a, 4 >\}$

	Iterazione 2	
	$IN[B]$	$OUT[B]$
BB1	undefined	$\emptyset$
BB2	$\emptyset$	$\{<k,2>\}$
BB3 (if)	$\{<k,2>\}$	$\{<k,2>\}$
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 (while)	$\{<a,4>\}$	$\{<a,4>\}$
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	$\{<a,4>\}$	$\{<a,4>\}$

	Iterazione 3	
	$IN[B]$	$OUT[B]$
BB1	undefined	$\emptyset$
BB2	$\emptyset$	$\{<k,2>\}$
BB3 (if)	$\{<k,2>\}$	$\{<k,2>\}$
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 (while)	$\{<a,4>\}$	$\{<a,4>\}$
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	$\{<a,4>\}$	$\{<a,4>\}$

La tabella è generata eseguendo un algoritmo iterativo naive.