

# 1 Alcançabilidade de Definições

*Gen* ...

*Kill* ...

*IN* ...

*OUT* ...

	Gen	Kill	IN	OUT
$B_1$	10000000	10100000	00000000	10000000
$B_2$	01000000	01000110	10000000	11000000
$B_3$	00110000	10110001	10000000	00110000
$B_4$	00001000	00001000	11110000	11111000
$B_5$	00000100	01000110	11111100	10111100
$B_6$	00000000	00000000	10111100	10111100
$B_7$	00000010	01000110	10111100	10111010
$B_8$	00000000	00000000	10111100	10111100
$B_9$	00000001	00010001	10111110	10101111
$B_{10}$	00000000	00000000	11000000	11000000
$B_{11}$	00000000	00000000	00110000	00110000
$B_{12}$	00000000	00000000	11111000	11111000
$B_{13}$	00000000	00000000	10111100	10111100
$B_{14}$	00000000	00000000	10111010	10111010
$B_{15}$	00000000	00000000	10111100	10111100

Table 1: Alcançabilidade de Definições — ( $b := 1, a := b + c, b := 7, d := b + c, e := b + c, a := b + c, a := b + c, d := b + c$ )

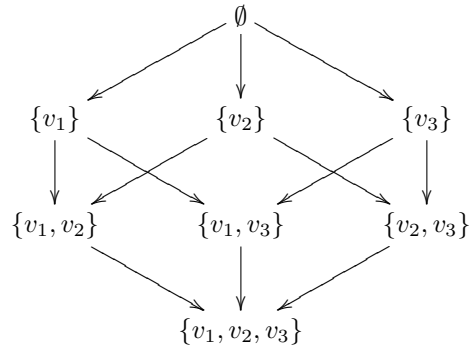
## 2 Vivacidade de Variáveis

*Gen* ...

*Kill* ...

*IN* ...

*OUT* ...



	Gen	Kill	IN	OUT
$B_1$	00000	01000	00100	01100
$B_2$	01100	10000	01100	01100
$B_3$	00100	01010	00100	01100
$B_4$	01100	00001	01100	01100
$B_5$	01100	10000	01100	01100
$B_6$	00000	00000	01100	01100
$B_7$	01100	10000	01100	01100
$B_8$	00000	00000	01100	01100
$B_9$	01100	00010	01100	00000
$B_{10}$	00000	00000	01100	01100
$B_{11}$	00000	00000	01100	01100
$B_{12}$	00000	00000	01100	01100
$B_{13}$	00000	00000	01100	01100
$B_{14}$	00000	00000	01100	01100
$B_{15}$	00000	00000	01100	01100

Table 2: Vivacidade de Variáveis —  $(a, b, c, d, e)$

### 3 Disponibilidade de Expressões

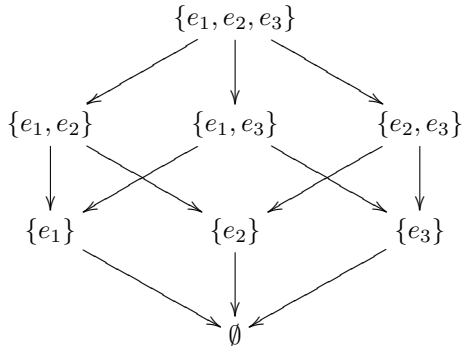
A análise é para frente (*forward*) e sua intenção é determinar em cada ponto do código, quais expressões estão disponíveis, isto é, foram seguramente executadas e, caso fossem executadas novamente (naquele ponto) produziriam o mesmo resultado.

*Gen* Indica quais expressões foram geradas dentro do bloco e que não foram “mortas” por redefinições de seus operandos dentro do mesmo bloco. **É igual à entrada das expressões antecipáveis.**

*Kill* Indica quais expressões (considerando o universo inteiro) foram mortas por redefinições (posteriores <sup>1</sup>) de seus operandos que ocorrem dentro do bloco.

*IN* Indica quais expressões estão disponíveis na entrada do bloco. É uma interseção das saídas dos blocos predecessores.

*OUT* Indica quais expressões estão disponíveis na saída do bloco. **É igual ao último *Gen*.**



	Gen	Kill	IN	OUT
$B_1$	0	1	0	0
$B_2$	1	0	0	1
$B_3$	1	1	0	1
$B_4$	1	0	1	1
$B_5$	1	0	1	1
$B_6$	0	0	1	1
$B_7$	1	0	1	1
$B_8$	0	0	1	1
$B_9$	1	0	1	1
$B_{10}$	0	0	1	1
$B_{11}$	0	0	1	1
$B_{12}$	0	0	1	1
$B_{13}$	0	0	1	1
$B_{14}$	0	0	1	1
$B_{15}$	0	0	1	1

Table 3: Disponibilidade de Expressões —  $((+, b, c))$

<sup>1</sup>Só faz sentido em análises internas ao bloco.

## 4 Disponibilidade de Expressões Antecipáveis

A análise é para trás (**backward**) e sua intenção é determinar em cada ponto do código, quais expressões podem ser movidas para o início do bloco (ou para blocos antecedentes).

*Gen* Indica quais expressões podem ser movidas para o início do bloco (ou para blocos antecedentes).

*Kill* Indica quais expressões (considerando o universo inteiro) foram mortas por redefinições (anteriores <sup>2</sup>) de seus operandos que ocorrem dentro do bloco.

*IN* Indica quais expressões podem ser movidas para blocos antecedentes.

*OUT* Indica quais expressões de blocos subseqüentes podem ser movidas para o final do bloco atual – estas expressões podem ou não serem antecipadas pelo bloco atual.

	Gen	Kill	IN	OUT
$B_1$	0	1	0	0
$B_2$	1	0	1	1
$B_3$	0	1	0	1
$B_4$	1	0	1	1
$B_5$	1	0	1	1
$B_6$	0	0	1	1
$B_7$	1	0	1	1
$B_8$	0	0	1	1
$B_9$	1	0	1	0
$B_{10}$	0	0	1	1
$B_{11}$	0	0	1	1
$B_{12}$	0	0	1	1
$B_{13}$	0	0	1	1
$B_{14}$	0	0	1	1
$B_{15}$	0	0	1	1

Table 4: Disponibilidade de Expressões Antecipáveis —  $((+, b, c))$

---

<sup>2</sup>Só faz sentido em análises internas ao bloco.

## 5 Disponibilidade Parcial de Expressões

*Gen* ...

*Kill* ...

*IN* ...

*OUT* ...

	Gen	Kill	IN	OUT
$B_1$	0	1	0	0
$B_2$	1	0	0	1
$B_3$	1	1	0	1
$B_4$	1	0	1	1
$B_5$	1	0	1	1
$B_6$	0	0	1	1
$B_7$	1	0	1	1
$B_8$	0	0	1	1
$B_9$	1	0	1	1
$B_{10}$	0	0	1	1
$B_{11}$	0	0	1	1
$B_{12}$	0	0	1	1
$B_{13}$	0	0	1	1
$B_{14}$	0	0	1	1
$B_{15}$	0	0	1	1

Table 5: Disponibilidade Parcial de Expressões —  $((+, b, c))$

## 6 Mortalidade de Variáveis

*Gen ...*

*Kill ...*

*In ...*

*In ...*

	Gen	Kill	IN	OUT
$B_1$	01000	00000	11011	10011
$B_2$	10000	01100	10011	10011
$B_3$	01010	00100	11011	10011
$B_4$	00001	01100	10011	10011
$B_5$	10000	01100	10011	10011
$B_6$	00000	00000	10011	10011
$B_7$	10000	01100	10011	10011
$B_8$	00000	00000	10011	10011
$B_9$	00010	01100	10011	11111
$B_{10}$	00000	00000	10011	10011
$B_{11}$	00000	00000	10011	10011
$B_{12}$	00000	00000	10011	10011
$B_{13}$	00000	00000	10011	10011
$B_{14}$	00000	00000	10011	10011
$B_{15}$	00000	00000	10011	10011

Table 6: Mortalidade de Variáveis —  $(a, b, c, d, e)$

## 7 Alcançabilidade de Definições para Propagação de Cópias

*Gen* ...

*Kill* ...

*In* ...

*In* ...

	Gen	Kill	IN	OUT
$B_1$	10000000	10000000	00000000	10000000
$B_2$	00000000	00000000	10000000	10000000
$B_3$	00000000	10000000	10000000	00000000
$B_4$	00000000	00000000	00000000	00000000
$B_5$	00000000	00000000	00000000	00000000
$B_6$	00000000	00000000	00000000	00000000
$B_7$	00000000	00000000	00000000	00000000
$B_8$	00000000	00000000	00000000	00000000
$B_9$	00000000	00000000	00000000	00000000
$B_{10}$	00000000	00000000	10000000	10000000
$B_{11}$	00000000	00000000	00000000	00000000
$B_{12}$	00000000	00000000	00000000	00000000
$B_{13}$	00000000	00000000	00000000	00000000
$B_{14}$	00000000	00000000	00000000	00000000
$B_{15}$	00000000	00000000	00000000	00000000

Table 7: Alcançabilidade de Definições para Propagação de Cópias — ( $b := 1, a := b+c, b := 7, d := b+c, e := b+c, a := b+c, a := b+c, d := b+c$ )

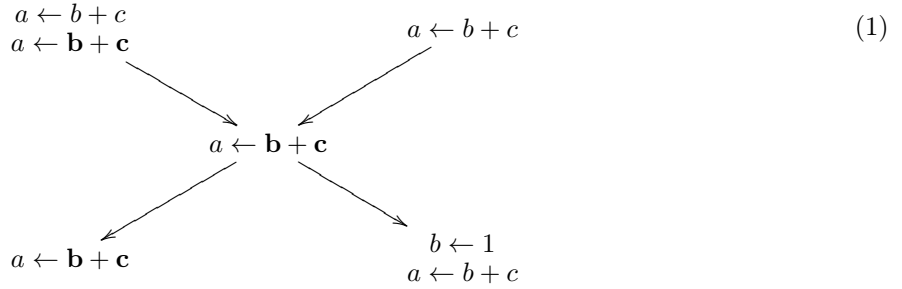
## 8 Eliminação de Redundâncias Parciais

### 8.1 Expressão Redundante

Uma expressão é *redundante* no ponto  $p$  se em cada caminho até  $p$ :

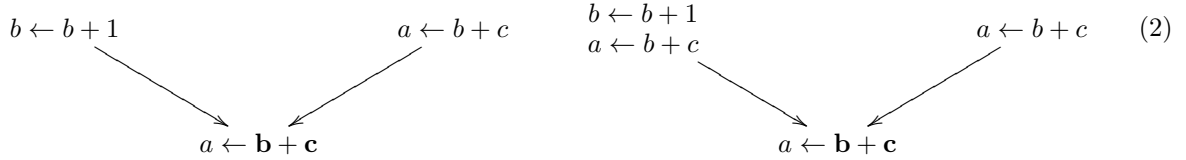
1. Ela é avaliada antes de alcançar  $p$ , e
2. Nenhum de seus operandos constituintes é redefinido antes de  $p$ .

Por exemplo, na Equação 1, as ocorrências de expressões em negrito são redundantes.



Uma expressão é *parcialmente redundante* no ponto  $p$  se ela é redundante ao longo de alguns caminhos, mas não todos, até  $p$ .

Por exemplo, na Equação 2, a expressão  $b + c$  em negrito no diagrama da esquerda é parcialmente redundante. A inserção de uma cópia de  $b + c$  depois da definição de  $b$  pode tornar uma expressão parcialmente redundante em uma totalmente redundante como mostra o diagrama da direita.



	ENTRY	$B_1$	$B_2$	$B_3$	$B_4$	$B_5$	$B_6$	$B_7$	$B_8$	$B_9$	$B_{10}$	$B_{11}$	$B_{12}$	$B_{13}$	$B_{14}$	$B_{15}$
e_gen	{0}	{0}	{1}	{0}	{1}	{1}	{1}	{0}	{1}	{0}	{1}	{0}	{0}	{0}	{0}	{0}
e_kill	{0}	{1}	{0}	{1}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}
anticipated_out	{0}	{0}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{0}	{1}	{1}	{1}	{1}	{1}
anticipated_in	{0}	{0}	{1}	{0}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}
available_in	{0}	{0}	{0}	{0}	{0}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}
available_out	{0}	{0}	{1}	{0}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}
earliest	{0}	{0}	{1}	{0}	{1}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}
postponable_in	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}
postponable_out	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}
latest	{0}	{0}	{1}	{0}	{1}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}
used_out	{0}	{0}	{1}	{0}	{1}	{1}	{1}	{1}	{1}	{1}	{0}	{1}	{1}	{1}	{1}	{1}
used_in	{0}	{0}	{0}	{0}	{0}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}	{1}
cond_1	{0}	{0}	{1}	{0}	{1}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}	{0}
cond_2	{0}	{0}	{1}	{0}	{1}	{1}	{1}	{0}	{1}	{0}	{1}	{0}	{0}	{0}	{0}	{0}

Table 8: Eliminação de Redundâncias Parciais —  $((+, b, c))$



## 9 Propagação de Constantes

*Gen ...*

*Kill ...*

*In ...*

*In ...*

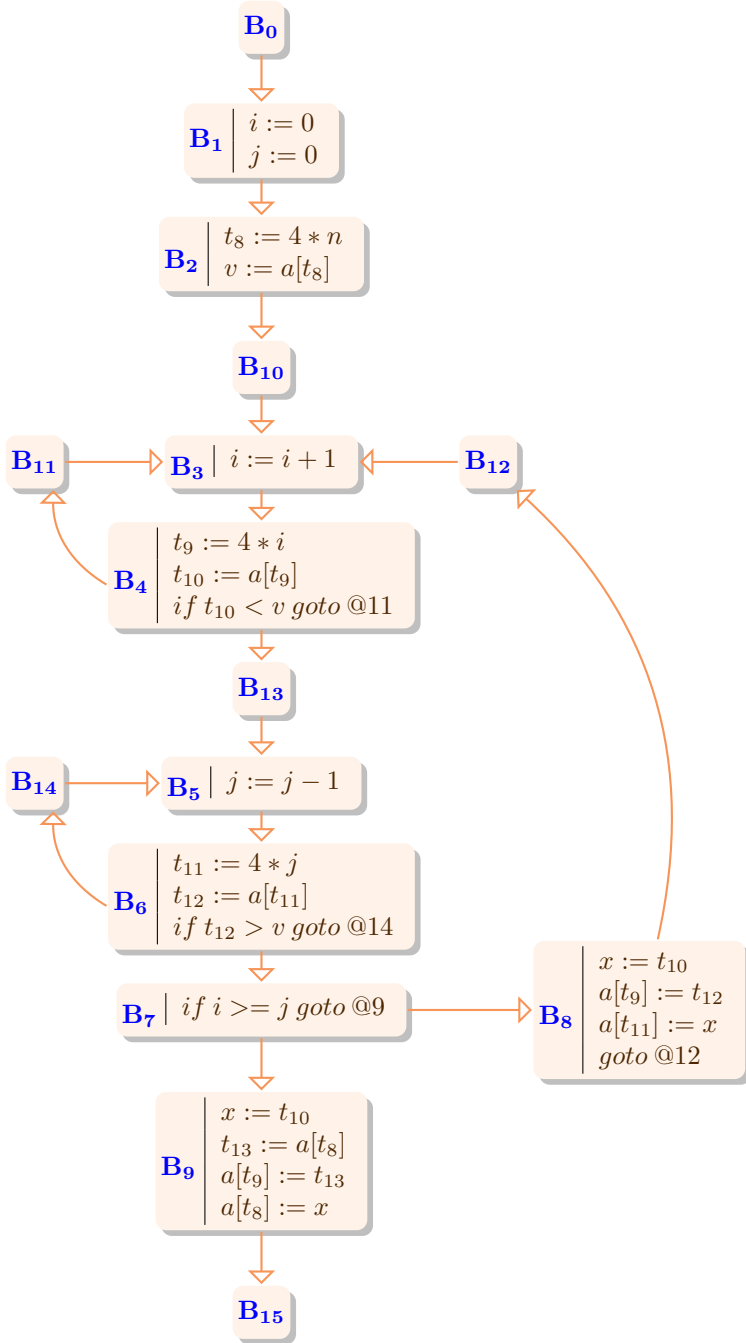
	IN	OUT
$B_1$	$(\top, \top, \top, \top, \top, \top)$	$(\top, 1, \top, \top, \top, \top)$
$B_2$	$(\top, 1, \top, \top, \top, \top)$	$(\top, 1, \top, \top, \top, \top)$
$B_3$	$(\top, 1, \top, \top, \top, \top)$	$(\top, 7, \top, \top, \top, \top)$
$B_4$	$(\top, 7, \top, \top, \top, \top)$	$(\top, 7, \top, \top, \top, \top)$
$B_5$	$(\top, \perp, \top, \top, \top, \top)$	$(\top, \perp, \top, \top, \top, \top)$
$B_6$	$(\top, \perp, \top, \top, \top, \top)$	$(\top, \perp, \top, \top, \top, \top)$
$B_7$	$(\top, \perp, \top, \top, \top, \top)$	$(\top, \perp, \top, \top, \top, \top)$
$B_8$	$(\top, \perp, \top, \top, \top, \top)$	$(\top, \perp, \top, \top, \top, \top)$
$B_9$	$(\top, \perp, \top, \top, \top, \top)$	$(\top, \perp, \top, \top, \top, \top)$
$B_{10}$	$(\top, \perp, \top, \top, \top, \top)$	$(\top, \perp, \top, \top, \top, \top)$
$B_{11}$	$(\top, 1, \top, \top, \top, \top)$	$(\top, 1, \top, \top, \top, \top)$
$B_{12}$	$(\top, 7, \top, \top, \top, \top)$	$(\top, 7, \top, \top, \top, \top)$
$B_{13}$	$(\top, \perp, \top, \top, \top, \top)$	$(\top, \perp, \top, \top, \top, \top)$
$B_{14}$	$(\top, \perp, \top, \top, \top, \top)$	$(\top, \perp, \top, \top, \top, \top)$
$B_{15}$	$(\top, \perp, \top, \top, \top, \top)$	$(\top, \perp, \top, \top, \top, \top)$
$B_{16}$	$(\top, \perp, \top, \top, \top, \top)$	$(\top, \perp, \top, \top, \top, \top)$

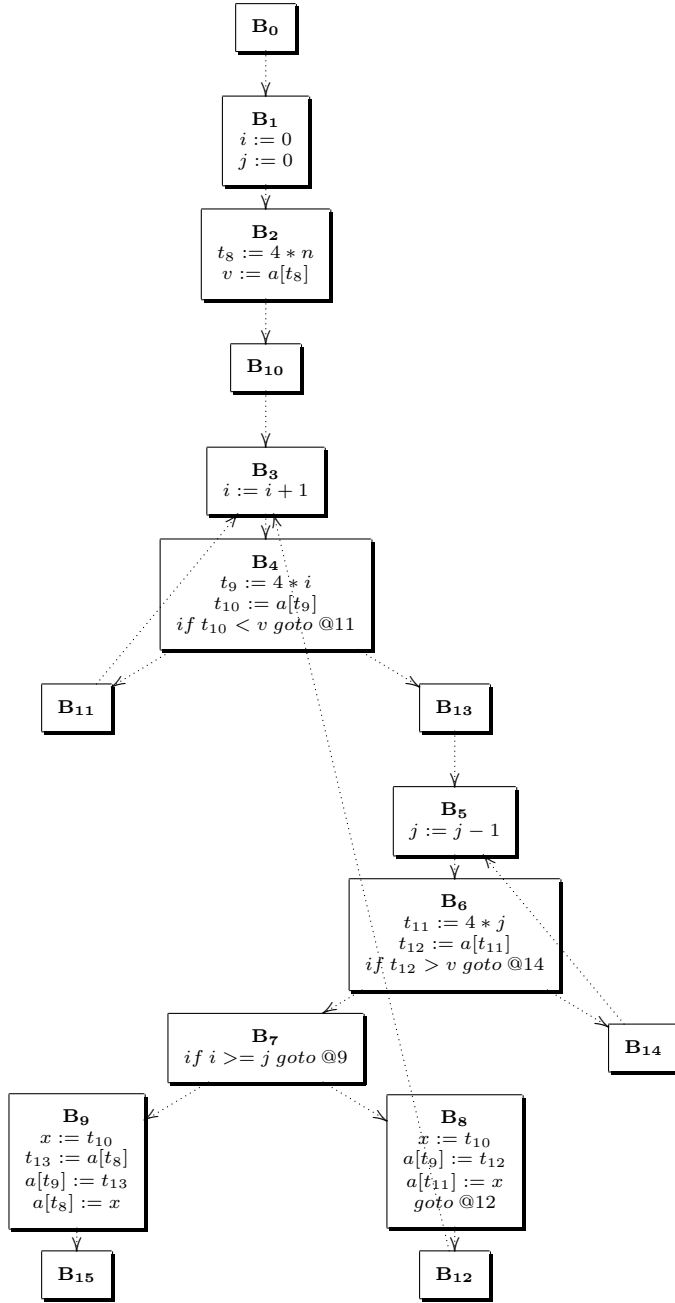
Table 9: Propagação de Constantes —  $(a, b, c, d, e, t_6)$

## 9.1 Exemplos

### 9.1.1 Exemplo 1

```
@1:  i := 0
      j := 0
@2:  t8 := 4 * n
      v := a[t8]
@10:
@11:
@12:
@3:  i := i + 1
@4:  t9 := 4 * i
      t10 := a[t9]
      if t10 < v goto @11
@13:
@14:
@5:  j := j - 1
@6:  t11 := 4 * j
      t12 := a[t11]
      if t12 > v goto @14
@7:  if i >= j goto @9
@8:  x := t10
      a[t9] := t12
      a[t11] := x
      goto @12
@9:  x := t10
      t13 := a[t8]
      a[t9] := t13
      a[t8] := x
@15:
```



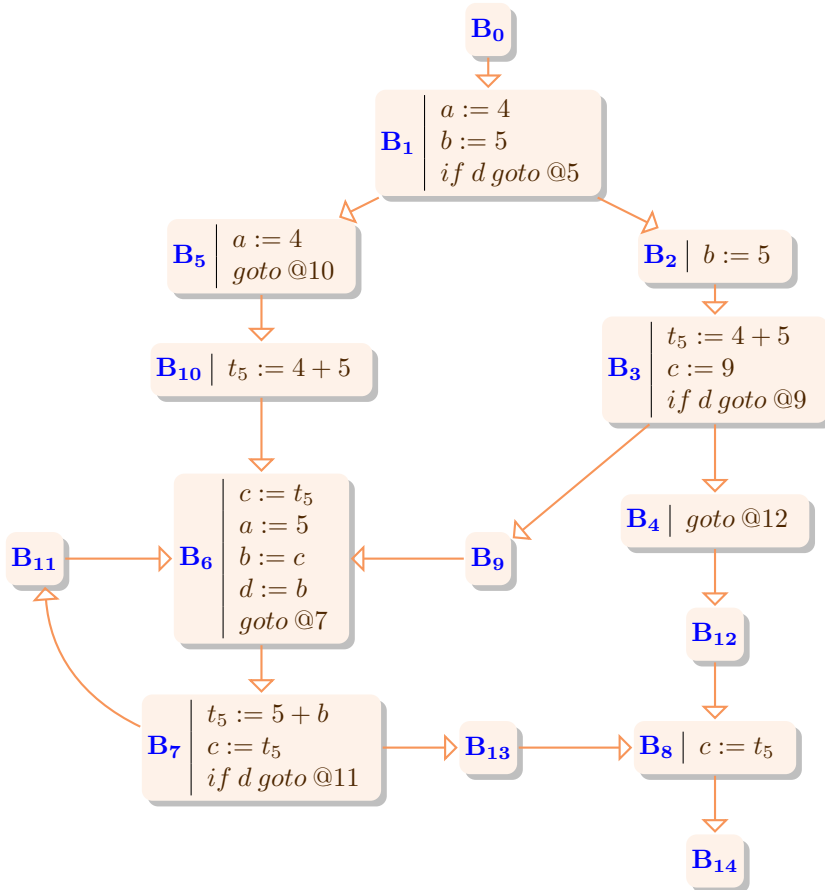


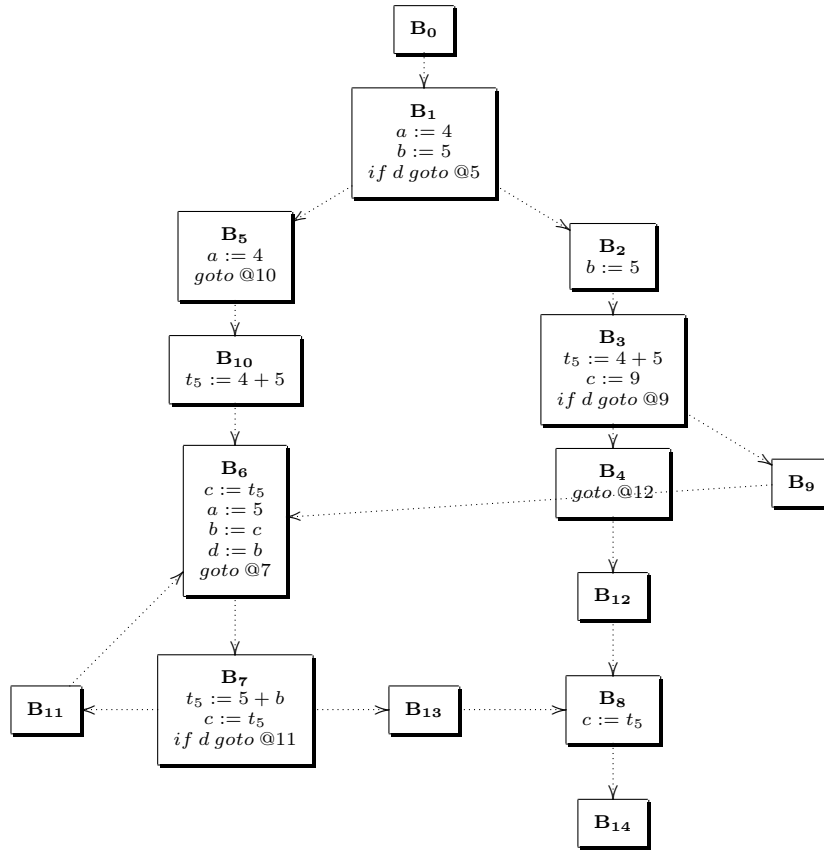
### 9.1.2 Exemplo 3

```

@1 :  a := 4
      b := 5
      if d goto @5
@2 :  b := 5
@3 :  t5 := 4 + 5
      c := 9
      if d goto @9
@4 :  goto @12
@5 :  a := 4
      goto @10
@10 : t5 := 4 + 5
@9 :
@11 :
@6 :  c := t5
      a := 5
      b := c
      d := b
      goto @7
@7 :  t5 := 5 + b
      c := t5
      if d goto @11
@12 :
@13 :
@8 :  c := t5
@14 :

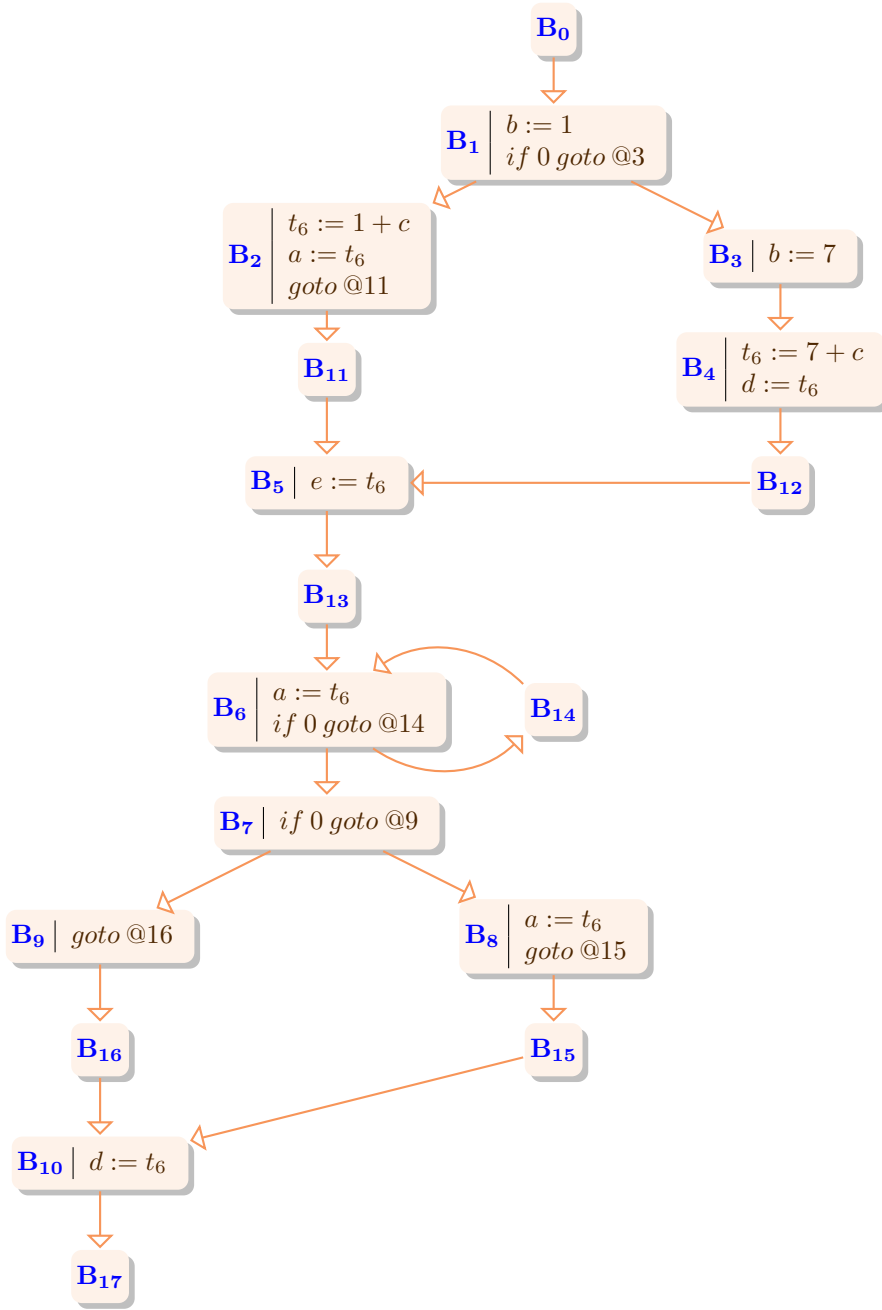
```

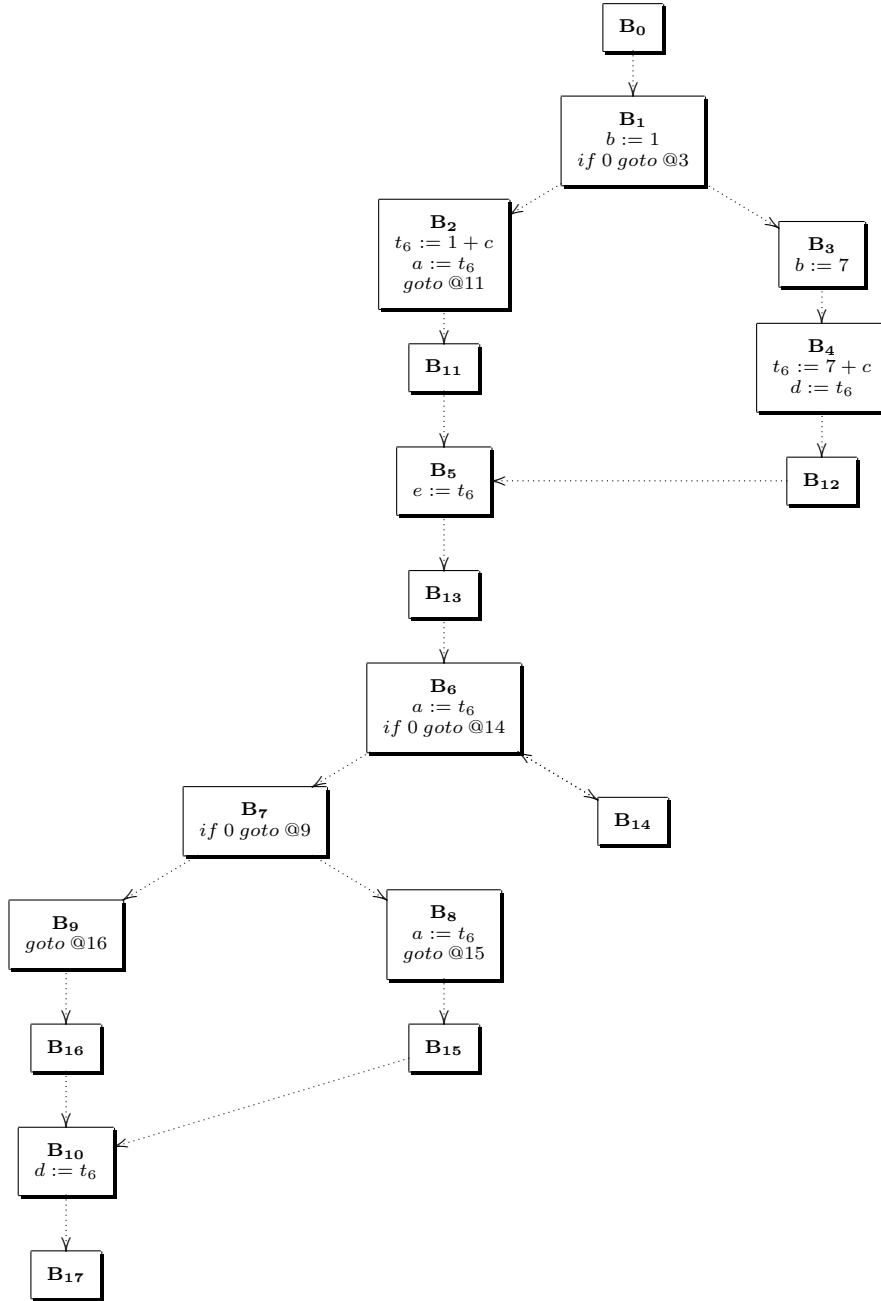




### 9.1.3 Exemplo 7

@1:  $b := 1$   
       if 0 goto @3  
 @2:  $t_6 := 1 + c$   
        $a := t_6$   
       goto @11  
 @3:  $b := 7$   
 @4:  $t_6 := 7 + c$   
        $d := t_6$   
 @11:  
 @12:  
 @5:  $e := t_6$   
 @13:  
 @14:  
 @6:  $a := t_6$   
       if 0 goto @14  
 @7: if 0 goto @9  
 @8:  $a := t_6$   
       goto @15  
 @9: goto @16  
 @15:  
 @16:  
 @10:  $d := t_6$   
 @17:







0	$i := 0$	@0:	<i>nop</i>	@0:	<i>nop</i>	@0:	<i>nop</i>	@0:	<i>nop</i>
1	$j := 0$	@1:	$i := 0$	@1:	$i := 0$	@1:	$i := 0$	@1:	$i := 0$
2	$t_8 := 4 * n$		$j := 0$		$j := 0$		$j := 0$		$j := 0$
3	$v := a[t_8]$		$t_8 := 4 * n$		$t_8 := 4 * n$	@2:	$t_8 := 4 * n$	@2:	$t_8 := 4 * n$
4	$i := i + 1$		$v := a[t_8]$		$v := a[t_8]$		$v := a[t_8]$		$v := a[t_8]$
5	$t_9 := 4 * i$	@2:	$i := i + 1$	@7:		@10:		@4:	
6	$t_{10} := a[t_9]$		$t_9 := 4 * i$	@8:		@11:		@5:	
7	<i>if</i> $t_{10} < v$ <i>goto</i> 4		$t_{10} := a[t_9]$	@9:		@12:		@6:	
8	$j := j - 1$		<i>if</i> $t_{10} < v$ <i>goto</i> @2	@2:	$i := i + 1$	@3:	$i := i + 1$	@3:	$i := i + 1$
9	$t_{11} := 4 * j$	@3:	$j := j - 1$		$t_9 := 4 * i$	@4:	$t_9 := 4 * i$	@7:	$t_9 := 4 * i$
10	$t_{12} := a[t_{11}]$		$t_{11} := 4 * j$		$t_{10} := a[t_9]$		$t_{10} := a[t_9]$		$t_{10} := a[t_9]$
11	<i>if</i> $t_{12} > v$ <i>goto</i> 8		$t_{12} := a[t_{11}]$		<i>if</i> $t_{10} < v$ <i>goto</i> @8		<i>if</i> $t_{10} < v$ <i>goto</i> @11		<i>if</i> $t_{10} < v$ <i>goto</i> @5
12	<i>if</i> $i >= j$ <i>goto</i> 17		<i>if</i> $t_{12} > v$ <i>goto</i> @3	@10:		@13:		@9:	
13	$x := t_{10}$	@4:	<i>if</i> $i >= j$ <i>goto</i> @6	@11:		@14:		@10:	
14	$a[t_9] := t_{12}$	@5:	$x := t_{10}$	@3:	$j := j - 1$	@5:	$j := j - 1$	@8:	$j := j - 1$
15	$a[t_{11}] := x$		$a[t_9] := t_{12}$		$t_{11} := 4 * j$	@6:	$t_{11} := 4 * j$	@11:	$t_{11} := 4 * j$
16	<i>goto</i> 4		$a[t_{11}] := x$		$t_{12} := a[t_{11}]$		$t_{12} := a[t_{11}]$		$t_{12} := a[t_{11}]$
17	$x := t_{10}$		<i>goto</i> @2		<i>if</i> $t_{12} > v$ <i>goto</i> @11		<i>if</i> $t_{12} > v$ <i>goto</i> @14		<i>if</i> $t_{12} > v$ <i>goto</i> @10
18	$t_{13} := a[t_8]$	@6:	$x := t_{10}$	@4:	<i>if</i> $i >= j$ <i>goto</i> @6	@7:	<i>if</i> $i >= j$ <i>goto</i> @9	@12:	<i>if</i> $i >= j$ <i>goto</i> @14
19	$a[t_9] := t_{13}$		$t_{13} := a[t_8]$	@5:	$x := t_{10}$	@8:	$x := t_{10}$	@13:	$x := t_{10}$
20	$a[t_8] := x$		$a[t_9] := t_{13}$		$a[t_9] := t_{12}$		$a[t_9] := t_{12}$		$a[t_9] := t_{12}$
			$a[t_8] := x$		$a[t_{11}] := x$		$a[t_{11}] := x$		$a[t_{11}] := x$
				@6:	<i>goto</i> @9	@9:	<i>goto</i> @12	@14:	<i>goto</i> @6
					$x := t_{10}$		$x := t_{10}$		$x := t_{10}$
					$t_{13} := a[t_8]$		$t_{13} := a[t_8]$		$t_{13} := a[t_8]$
					$a[t_9] := t_{13}$		$a[t_9] := t_{13}$		$a[t_9] := t_{13}$
					$a[t_8] := x$		$a[t_8] := x$		$a[t_8] := x$

