

MO615B - Implementação de
Linguagens II

e

MC900A - Tópicos Especiais em
Linguagem de Programação

Prof. Sandro Rigo

www.ic.unicamp.br/~sandro

Otimizações em Laços

Detecção de Invariantes do Laço

- Invariante:
 - Valor da computação não muda dentro do laço
- $s: x = y + z$
 - Dentro de um laço
 - Porém, todas as definições de y e z que alcançam s estão fora do laço
 - y e/ou z podem ser constantes
- Então s é invariante
 - $t: v = x + w$ dentro do laço
 - Invariante se w é invariante e s é a única definição de x alcançando t

Detecção de Invariantes do Laço

- Detectados usando *ud-chains*
- Fazendo-se várias passadas no laço, até que nenhuma sentença seja marcada como invariante

Variáveis de Indução

- x é uma “variável de indução” em um laço L
 - Quando x é decrementada ou incrementada por um fator constante.
 - O incremento pode ocorrer mais de uma vez em uma mesma iteração
 - O # de incrementos pode ser diferente a cada iteração
- Variáveis de “indução básicas”
 - Somente atribuições da forma $i = i \pm c$
 - onde c é constante

Variáveis de Indução

- “Variáveis adicionais” (ou dependentes)
 - Variável j definida de forma única e dentro de L
 - Valor é função linear de uma “variável de indução básica” i , onde j é definida por:
 - $j = c*i + d$ (c e d constantes)
 - E representada pela tripla (i, c, d)
 - Diz-se que j pertence à família de i
 - Por definição, i pertence à sua própria família
 - Tripla $(i, 1, 0)$

Detecção de Variáveis de Indução

- Informações necessárias
 - Laço L
 - *Reaching Definitions*
 - Invariantes no laço computadas
- Método
 1. Encontre as variáveis básicas de indução, associando triplas do tipo $(i, 1, 0)$
 - $i = i \pm c$, onde c é constante

Detecção de Variáveis de Indução

2. Encontre k tal que k tenha uma única definição em L de uma das formas:

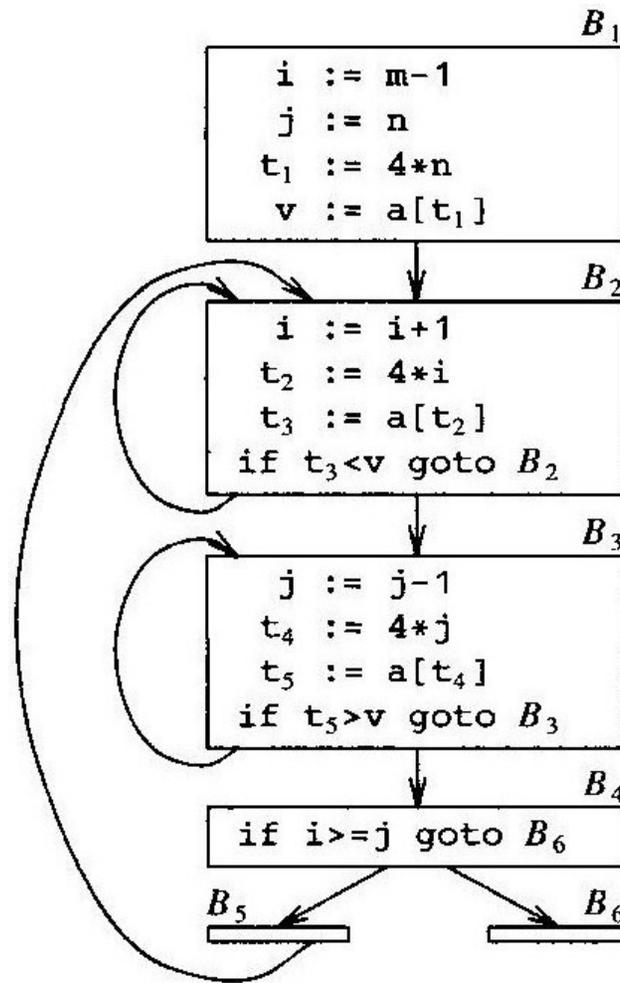
- $k = j * b, k = b * j, k = j / b, k = j \pm b, k = b \pm j$
 - (b constante e j variável de indução, básica ou não)
- Se j é básica, k é da família de j
 - Se $k = j * b$ a tripla é $(j, b, 0)$
- Senão, seja j da família de i . Devemos ter:
 - (a) Não há atribuição a i entre a atribuição a j em L e a atribuição a k ;
 - (b) Nenhuma definição de j de fora do laço alcança k

Detecção de Variáveis de Indução

- No caso comum j e k são temporários no mesmo bloco
 - \Rightarrow torna fácil checar as condições anteriores.
- No caso geral
 - *Reaching definitions* fornece o necessário se analisarmos o CFG do laço L
- Tripla: se tripla de $j = (i, c, d)$ e $k = b * j$
 - $K: (i, b * c, b * d)$ com b, c, d constantes

Exemplo

- Quais as variáveis de indução?



(a) Before

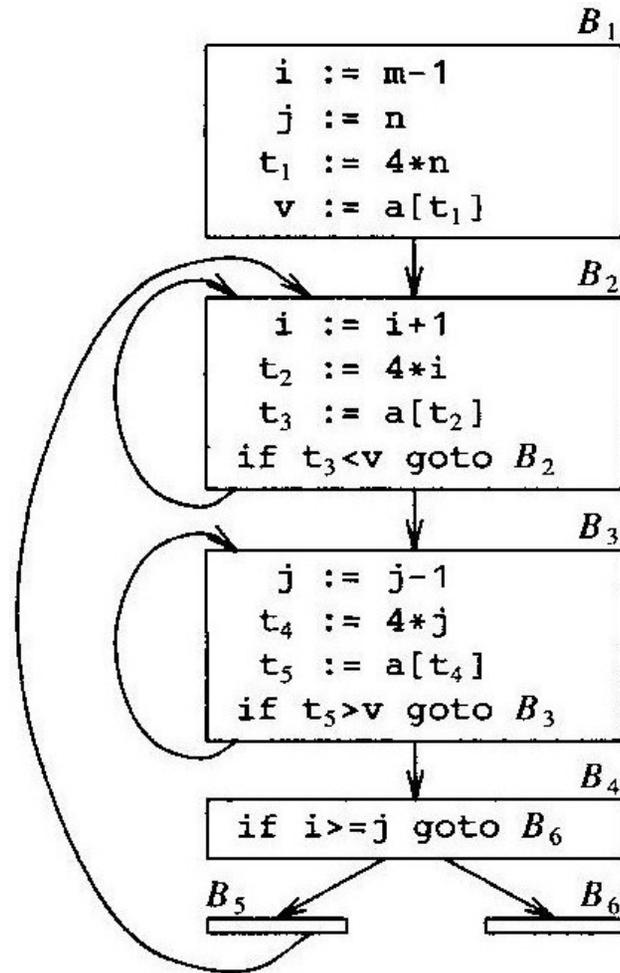
Strength Reduction

- Informações
 - Laço L
 - *Reaching Definitions*
 - Famílias de variáveis de indução
- Método
 - Considere cada variável de indução básica i
 - Para cada j na família de i com tripla (i,c,d)
 - Crie uma nova variável s
 - Se j_1 e j_2 têm a mesma tripla crie apenas uma para ambos
 - Troque a atribuição à variável j por $j=s$

Strength Reduction

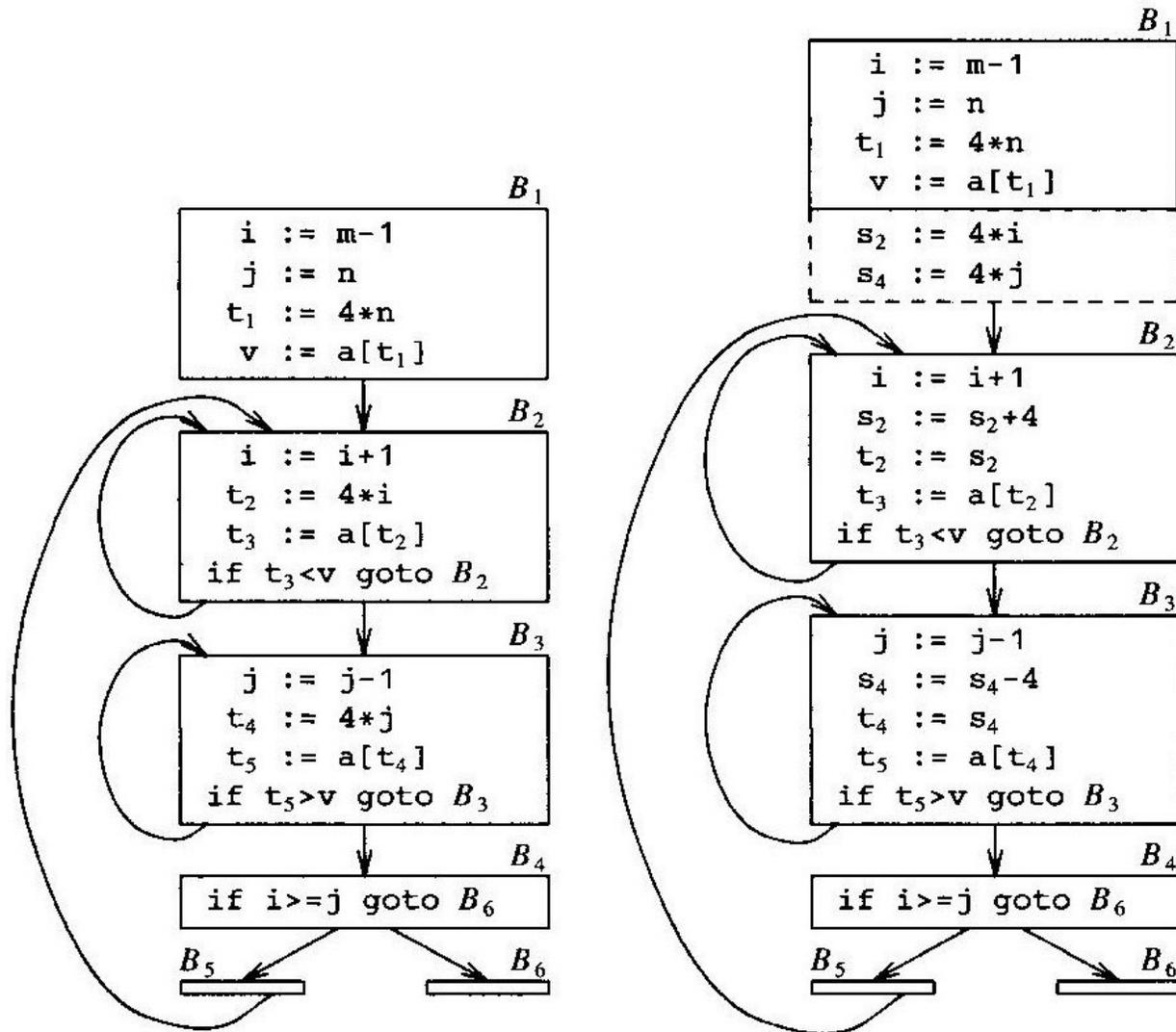
- Método (continuação)
 - Imediatamente após $i = i+n$, n constante, adicione
 - $s = s + c*n$ ($c*n$ é constante)
 - Coloque s na família de i com tripla (i,c,d)
- Falta garantir que $s = c*i + d$ no início do laço
 - No final do *pre-header* coloque
 - $s = c*i$ ($s=i$ se $c=1$)
 - $s = s + d$ (omitir se $d=0$)

Exemplo



(a) Before

Strength Reduction Aplicada



(a) Before

(b) After

Eliminação de Variáveis de Indução

- Variáveis de indução básica (i)
 - Únicos usos são para computar outras variáveis de indução
 - Ou condições em desvios
- Seja j uma variável da família de i
 - Podemos substituir i por j (i,c,d) nos testes
 - Assim acabamos eliminando i
 - É melhor escolher j com c e d simples
 - Ex: (i,1,0)

Eliminação de Variáveis de Indução

- Assumindo que: $t = 4 * i$
 - $i \geq j$ é equivalente a $t \geq 4*j$
- E se $t = -4 * i$?

Eliminação de Variáveis de Indução

- Cuidado com constantes negativas
 - $i \geq j$ é equivalente a $t \geq 4*j$
 - $i \geq j$ é equivalente a $t \leq -4*j$
- Assumindo c positivo
 - *If i relop x goto B*, x não é v.i. Substitui por:
 - $r = c*x$
 - $r = r + d$
 - *If j relop r goto B*
 - r é um novo temporário
 - O teste x relop i é tratado de forma análoga

Eliminação de Variáveis de Indução

- Se x também v.i., ou seja, estamos comparando 2 variáveis de indução. :
 - If $i_1 \text{ relop } i_2 \text{ goto } B$.
 - Verificamos se ambas i_1 e i_2 podem ser removidas
 - O caso bom é quando temos
 - $j_1 = (i_1, c_1, d_1)$ e $j_2 = (i_2, c_2, d_2)$
 - E $c_2 = c_1$ e $d_1 = d_2$
 - Assim $i_1 \text{ relop } i_2$ é equivalente a $j_1 \text{ relop } j_2$
 - Outros casos podem não valer à pena
 - Introduz duas multiplicações e uma adição
 - Elimina apenas 2 passos

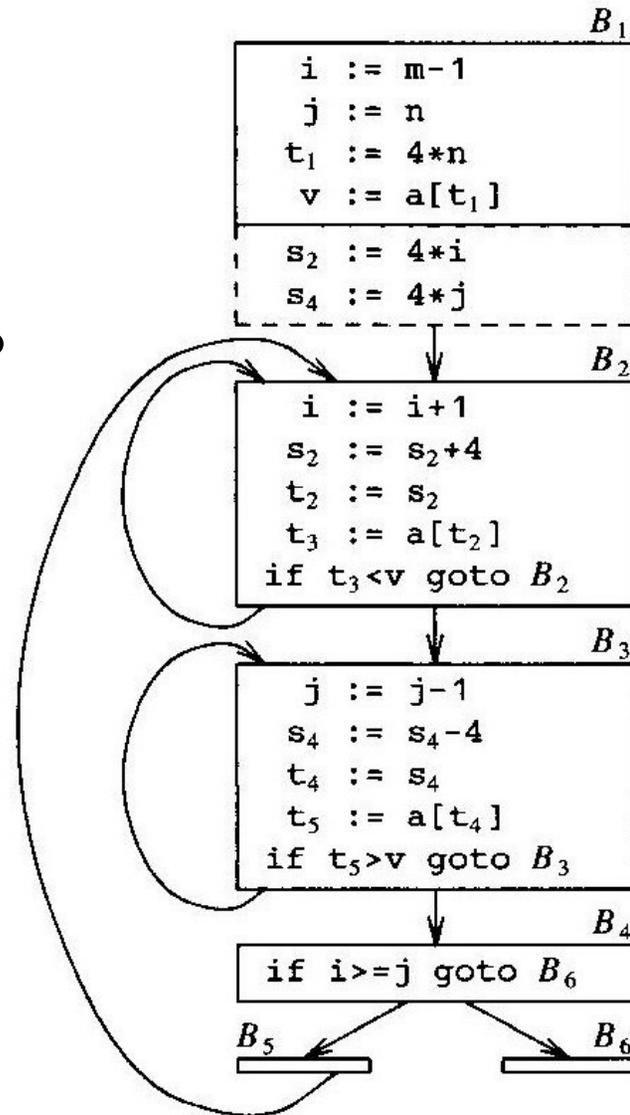
Eliminação de Variáveis de Indução

- Finalmente:
 - Elimine todas as atribuições às v.i. básicas em L
 - Tornaram-se inúteis
- Considere sentenças do tipo
 - *Copy propagation* pode ser aplicada para eliminar as cópias $j = s$ introduzidas pela *strength reduction*.

Strength Reduction Aplicada

Como fica a eliminação das v. i.
nos laços ao redor de B2 e B3?

- Quais as variáveis de indução?
- Podem ser eliminadas?

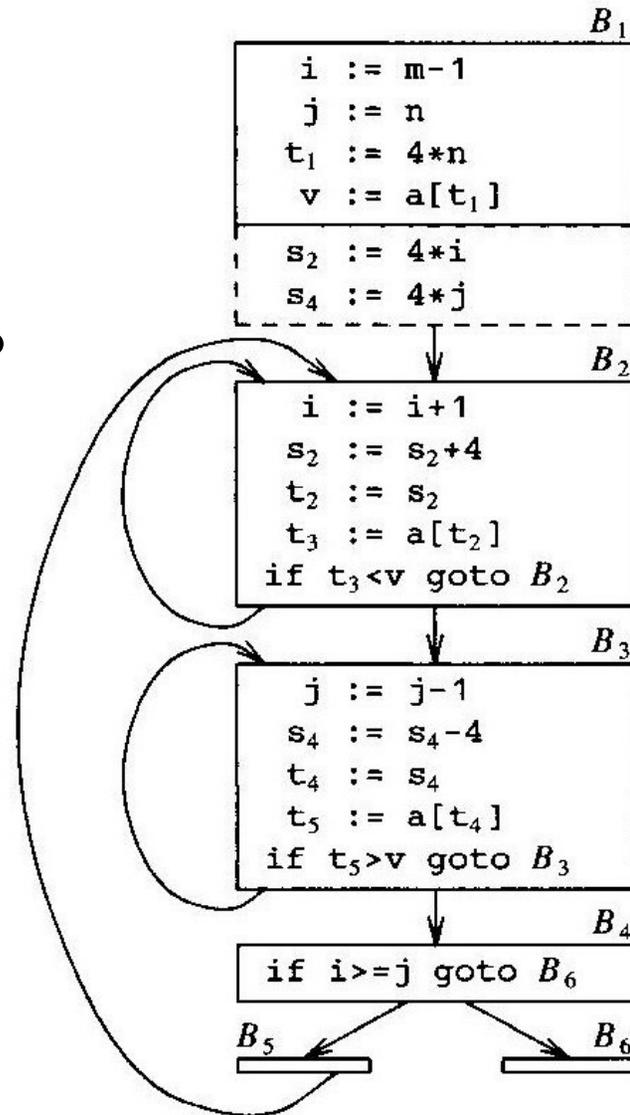


(b) After

Strength Reduction Aplicada

Como fica a eliminação das v. i.
no laço {B2, B3, B4, B5}?

- Quais as variáveis de indução?
- Podem ser eliminadas?



(b) After