

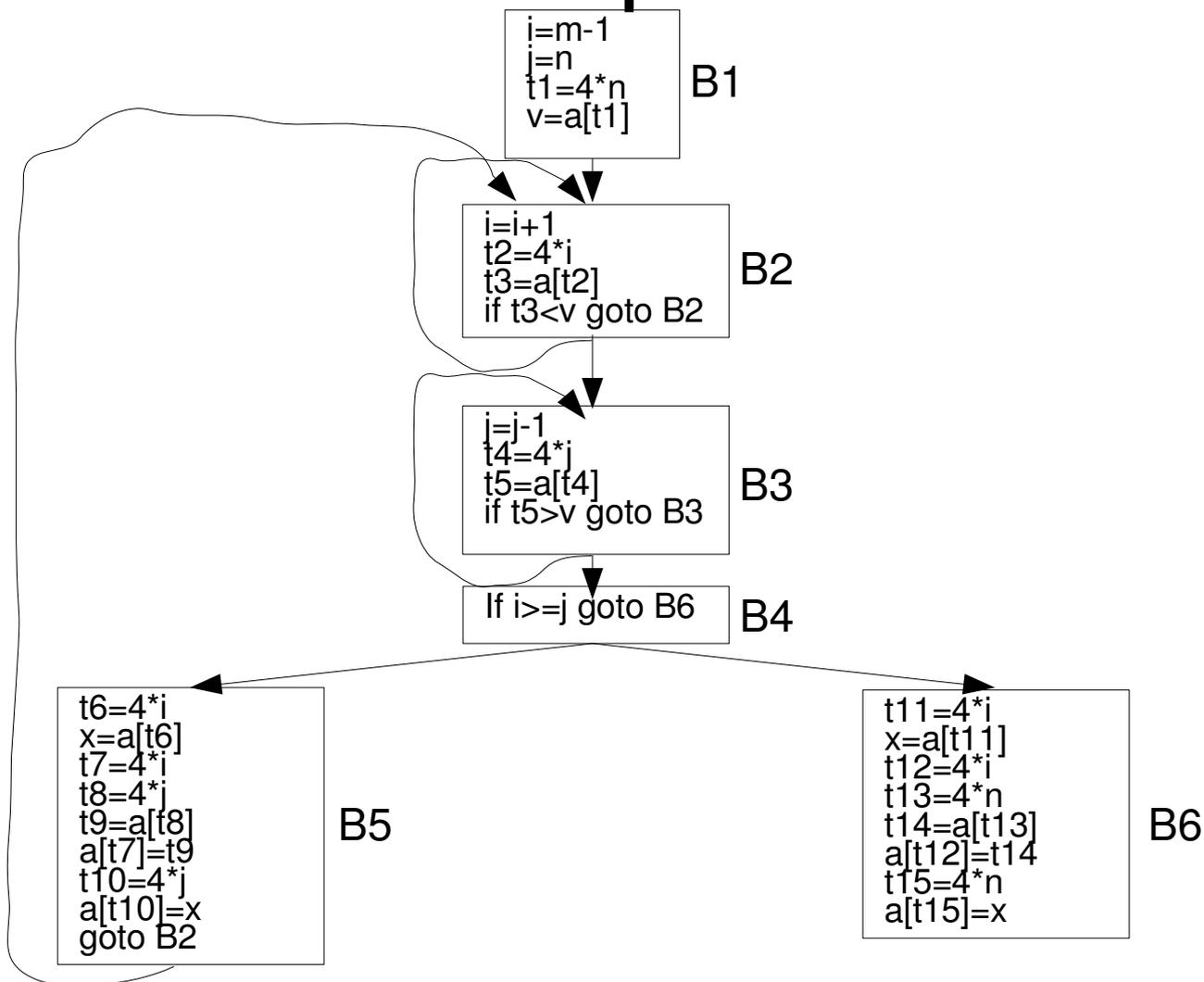
Geração de Código

Otimizações independentes de
máquina

Otimizações independentes de máquina

- São otimizações que visam “transformar” o código, deixando-o mais coeso e com melhor desempenho mas, além de tudo, preservando a semântica
- Eliminação de subexpressões comuns (CSE), propagação de cópias, eliminação de código morto (DCE) são exemplos de tais otimizações

Otimizações independentes de máquina



Eliminação de subexpressões comuns

- Um ocorrência de uma expressão E é chamada uma subexpressão comum se E foi computada previamente e os valores das variáveis em E não foram modificadas desde a última atribuição de E
- Evita-se recomputar E se é possível utilizar seu valor previamente computado
- Exemplo: as atribuições para $t7$ e $t10$ em $B5$ computam as expressões 4^*i e 4^*j , respectivamente
 - Desde que essas expressões já foram computadas em $t6$ e $t8$, podemos eliminar a computação dessas expressões em $t7$ e 10

Eliminação de subexpressões comuns

- As eliminações de subexpressões comuns podem ser feitas além do limite de blocos básicos
- Observe as atribuições de t_6 e t_8 em B5, elas poderiam ser eliminadas?
 - Dica: Observe t_2 em B2 e t_4 em B3
- O bloco B5 (tente também em B6) ainda possui subexpressões que poderiam ser eliminadas?

Propagação de Cópias

- Uma atribuição da forma $u=v$ é chamada atribuição de cópia, ou apenas cópia
- A idéia desta otimização é utilizar v ao invés de u sempre que for possível
- Exemplo: Considere a existência da atribuição $x=t3$ em $B5$, como é uma atribuição de cópia, pode-se trocar $a[t4]=x$ por $a[t4]=t3$
- Há outras cópias que poderiam ser propagadas?

Eliminação de Código Morto

- Uma variável é “viva” em um ponto p se seu valor pode ser usado após o ponto p . De outra forma, o valor dessa variável é denominado “morto”
- Código “morto” pode aparecer como resultado de otimizações/transformações no código
 - Uma vantagem da propagação de cópias é que, freqüentemente, transforma declarações em código morto
- Exemplo: a transformação feita, no exemplo anterior, sobre propagação de cópias deixou x como sendo um código morto
 - Por que x (mesmo sendo usado em B6) pode ser eliminado?

Variáveis de Indução e Redução em Força

- Uma variável x é dita “variável de indução” se existe uma constante c (positiva ou negativa) tal que o valor de x é definido através de operações que utilizam c
- Como exemplo, i e t_2 são variáveis de indução no laço B2
- A substituição de uma operação que consome mais ciclos (cara) por outra que consome menos ciclos (barata) é denominada redução em força

Variáveis de Indução e

Redução em Força

- Exemplo: Considere o bloco básico B3. Logo depois da atribuição $j=j-1$, pode-se notar que o valor de $t4$ é definido como $t4=4*j+4$. Após a atribuição de $t4$ ($t4=4*j$), podemos substituir o valor de $t4$ por $t4=t4-4$
 - A questão aqui é como determinar um valor para $t4$ a primeira vez que o laço for executado?
 - Veja que $t4$ deve manter o relacionamento $t4=4*j$ na entrada do bloco B3
- Por que essa técnica (redução em força) pode gerar ganhos de desempenho para a aplicação?

Variáveis de Indução e Redução em Força

- Tomando como base o exemplo anterior, observe que i e j ficaram sem uso nos blocos B2 e B3, respectivamente
- Note também que i possui relacionamento com $t2$, pois $t2=4*i$, enquanto j possui relacionamento com $t4$, pois $t4=4*j$
- Com isso, poderíamos substituir i por $t2$ e j por $t4$ em B4?
 - Se isso for possível, i e j poderiam ser eliminados de seus blocos?