

Um compilador simples de uma passagem

Um tradutor para expressões simples

Prof. Edson Alves

Faculdade UnB Gama

Sumário

1. Máquinas de Pilha Abstratas

Máquinas de Pilha Abstratas

- ▶ A interface de vanguarda do compilador produz uma representação intermediária do programa fonte, que será usada pela interface de retaguarda para produzir o programa alvo

Máquinas de Pilha Abstratas

- ▶ A interface de vanguarda do compilador produz uma representação intermediária do programa fonte, que será usada pela interface de retaguarda para produzir o programa alvo
- ▶ Uma possível forma para a representação intermediária é a máquina de pilha abstrata

Máquinas de Pilha Abstratas

- ▶ A interface de vanguarda do compilador produz uma representação intermediária do programa fonte, que será usada pela interface de retaguarda para produzir o programa alvo
- ▶ Uma possível forma para a representação intermediária é a máquina de pilha abstrata
- ▶ Uma máquina de pilha abstrata possui memórias separadas para dados e instruções, e todas as operações aritméticas são realizadas sobre os valores em uma pilha

Máquinas de Pilha Abstratas

- ▶ A interface de vanguarda do compilador produz uma representação intermediária do programa fonte, que será usada pela interface de retaguarda para produzir o programa alvo
- ▶ Uma possível forma para a representação intermediária é a máquina de pilha abstrata
- ▶ Uma máquina de pilha abstrata possui memórias separadas para dados e instruções, e todas as operações aritméticas são realizadas sobre os valores em uma pilha
- ▶ As instruções são divididas em três classes: aritmética inteira, manipulação de pilha e fluxo de controle

Máquinas de Pilha Abstratas

- ▶ A interface de vanguarda do compilador produz uma representação intermediária do programa fonte, que será usada pela interface de retaguarda para produzir o programa alvo
- ▶ Uma possível forma para a representação intermediária é a máquina de pilha abstrata
- ▶ Uma máquina de pilha abstrata possui memórias separadas para dados e instruções, e todas as operações aritméticas são realizadas sobre os valores em uma pilha
- ▶ As instruções são divididas em três classes: aritmética inteira, manipulação de pilha e fluxo de controle
- ▶ O ponteiro *pc* indica qual é a próxima instrução a ser executada

Instruções aritméticas

- ▶ A máquina de pilha abstrata precisa implementar cada operador da linguagem intermediária

Instruções aritméticas

- ▶ A máquina de pilha abstrata precisa implementar cada operador da linguagem intermediária
- ▶ Operações elementares, como adição e subtração, são suportadas diretamente

Instruções aritméticas

- ▶ A máquina de pilha abstrata precisa implementar cada operador da linguagem intermediária
- ▶ Operações elementares, como adição e subtração, são suportadas diretamente
- ▶ Operações mais sofisticadas devem ser implementadas como uma sequência de instruções da máquina

Instruções aritméticas

- ▶ A máquina de pilha abstrata precisa implementar cada operador da linguagem intermediária
- ▶ Operações elementares, como adição e subtração, são suportadas diretamente
- ▶ Operações mais sofisticadas devem ser implementadas como uma sequência de instruções da máquina
- ▶ A título de simplificação, assuma que existe uma instrução para cada operação aritmética

Instruções aritméticas

- ▶ A máquina de pilha abstrata precisa implementar cada operador da linguagem intermediária
- ▶ Operações elementares, como adição e subtração, são suportadas diretamente
- ▶ Operações mais sofisticadas devem ser implementadas como uma sequência de instruções da máquina
- ▶ A título de simplificação, assuma que existe uma instrução para cada operação aritmética
- ▶ O código de uma máquina de pilha abstrata para uma expressão simula a avaliação de uma representação posfixa, usando uma pilha

Instruções aritméticas

- ▶ A máquina de pilha abstrata precisa implementar cada operador da linguagem intermediária
- ▶ Operações elementares, como adição e subtração, são suportadas diretamente
- ▶ Operações mais sofisticadas devem ser implementadas como uma sequência de instruções da máquina
- ▶ A título de simplificação, assuma que existe uma instrução para cada operação aritmética
- ▶ O código de uma máquina de pilha abstrata para uma expressão simula a avaliação de uma representação posfixa, usando uma pilha
- ▶ A avaliação segue da esquerda para a direita, empilhando os operandos

Instruções aritméticas

- ▶ A máquina de pilha abstrata precisa implementar cada operador da linguagem intermediária
- ▶ Operações elementares, como adição e subtração, são suportadas diretamente
- ▶ Operações mais sofisticadas devem ser implementadas como uma sequência de instruções da máquina
- ▶ A título de simplificação, assuma que existe uma instrução para cada operação aritmética
- ▶ O código de uma máquina de pilha abstrata para uma expressão simula a avaliação de uma representação posfixa, usando uma pilha
- ▶ A avaliação segue da esquerda para a direita, empilhando os operandos
- ▶ Quando um operador é encontrado, seus operandos são extraídos da pilha (do último para o primeiro), a operação é realizada e o resultado é inserido no topo da pilha

Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

Ações

Pilha

Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

Ações

1 2 + 3 *



Pilha

Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

Ações

1 2 + 3 *



Empilhar o valor 1

Pilha

Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

1 2 + 3 *
↑

Ações

Empilhar o valor 1

Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

Ações

1 2 + 3 *



Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

Ações

1 2 + 3 *



Empilhar o valor 2

Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

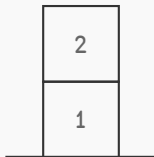
1 2 + 3 *



Ações

Empilhar o valor 2

Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

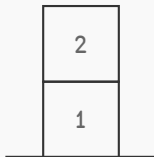
Instrução

Ações

1 2 + 3 *



Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

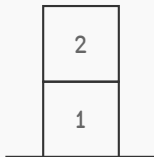
1 2 + 3 *



Ações

Adicionar 1+2

Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

1 2 + 3 *

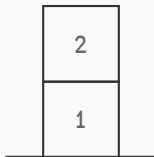


Ações

Adicionar 1+2

Empilhar a soma

Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

1 2 + 3 *



Ações

Adicionar 1+2

Empilhar a soma

Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

1 2 + 3 *



Ações

Empilhar o valor 3

Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

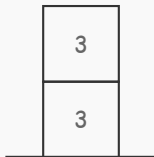
1 2 + 3 *

↑

Ações

Empilhar o valor 3

Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

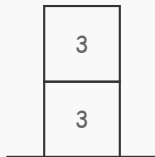
Instrução

Ações

1 2 + 3 *



Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

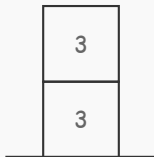
1 2 + 3 *



Ações

Multiplicar $3*3$

Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

1 2 + 3 *

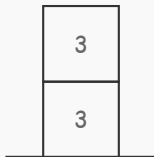
↑

Ações

Multiplicar $3*3$

Empilhar o produto

Pilha



Exemplo de avaliação da expressão $12+3*$ por máquina abstrata de pilha

Instrução

1 2 + 3 *
 ↑

Ações

Multiplicar $3*3$

Empilhar o produto

Pilha



Valores-L e valores-R

- ▶ O significado de um identificador depende da posição onde ele ocorre em uma atribuição

Valores-L e valores-R

- ▶ O significado de um identificador depende da posição onde ele ocorre em uma atribuição
- ▶ No lado esquerdo, o identificador se refere à localização de memória onde o valor deve ser armazenado

Valores-L e valores-R

- ▶ O significado de um identificador depende da posição onde ele ocorre em uma atribuição
- ▶ No lado esquerdo, o identificador se refere à localização de memória onde o valor deve ser armazenado
- ▶ No lado direito, o identificador se refere ao valor armazenado na localização de memória associada ao identificador

Valores-L e valores-R

- ▶ O significado de um identificador depende da posição onde ele ocorre em uma atribuição
- ▶ No lado esquerdo, o identificador se refere à localização de memória onde o valor deve ser armazenado
- ▶ No lado direito, o identificador se refere ao valor armazenado na localização de memória associada ao identificador
- ▶ Valor-L e valor-R se referem aos valores apropriados para os lados esquerdo e direito de uma atribuição, respectivamente

Valores-L e valores-R

- ▶ O significado de um identificador depende da posição onde ele ocorre em uma atribuição
- ▶ No lado esquerdo, o identificador se refere à localização de memória onde o valor deve ser armazenado
- ▶ No lado direito, o identificador se refere ao valor armazenado na localização de memória associada ao identificador
- ▶ Valor-L e valor-R se referem aos valores apropriados para os lados esquerdo e direito de uma atribuição, respectivamente
- ▶ Um mesmo identificador pode ser um valor-L e um valor-R na mesma atribuição (por exemplo, o identificador x em $x = x + 1$)

Manipulação da pilha

Uma máquina de pilha abstrata suporta as seguintes instruções para a manipulação da pilha:

Instrução	Significado
<code>push v</code>	empilha v
<code>pop</code>	desempilha o valor do topo da pilha
<code>valor-r p</code>	empilha o valor armazenado no endereço de memória p
<code>valor-l p</code>	empilha o endereço de memória p
<code>:=</code>	o valor-R do topo da pilha é armazenado no valor-L do subtopo (elemento que está abaixo do topo) da pilha
<code>copiar</code>	empilha o valor do topo da pilha