
Verificação de Tipos

Análise Semântica: Verificação de Tipos

O que é um tipo?

- A definição de tipo pode mudar de linguagem para linguagem.

Consenso:

- Um conjunto de valores
- Um conjunto de operações sobre estes valores

Classes são uma instanciação moderna da noção de tipo.

Análise Semântica: Verificação de Tipos

Por que os tipos são necessários?

Considere o seguinte fragmento de linguagem *assembly*:

```
add $r1, $r2, $r3
```

Quais os tipos de `$r1`, `$r2` e `$r3` ?

Análise Semântica: Verificação de Tipos

Certas operações são válidas para valores de alguns tipos

- Em C **não** faz sentido somar dois ponteiros para inteiros
- Em C faz sentido somar dois inteiros

Ambas as construções gerariam o mesmo código *assembly*:

```
add $r1, $r2, $r3
```

O sistema de tipos de uma linguagem serve para especificar quais operações são válidas para cada um dos tipos.

O objetivo da verificação de tipos é assegurar que as operações da linguagem estão sendo usadas com os tipos corretos.

Análise Semântica: Verificação de Tipos

As linguagens podem ser basicamente divididas em 3 tipos:

- Estaticamente tipadas: Toda ou quase toda a verificação de tipos é feita em tempo de compilação (C, Java, Pascal, etc)
- Dinamicamente tipadas: Toda a verificação de tipos é feita em tempo de execução do programa (basic)
- Não-tipadas: Não existe verificação de tipos (*assembly*)

Análise Semântica: Verificação de Tipos

Verificação de tipos

- O programador declara tipos para os identificadores
- O compilador infere o tipo para cada expressão

A verificação de tipos, verifica se para cada operação no programa se os tipos utilizados são válidos para a mesma.

No processo de inferência de tipo, o compilador preenche as informações que não estão presentes.

Análise Semântica: Verificação de Tipos

Verificação de tipos: Regras de Inferência

Se duas expressões E_1 e E_2 possuem um determinado tipo, então uma operação sobre elas irá gerar uma expressão E_3 com um determinado tipo.

Ex.:

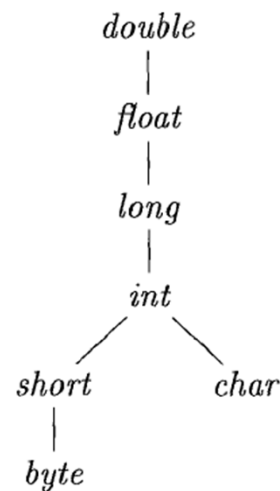
Se e_1 tem tipo **int** e e_2 tem tipo **int**,
Então $e_1 + e_2$ irá ter tipo **int**.

Se e_1 tem tipo **int*** e e_2 tem tipo **int**,
Então $e_1 + e_2$ irá ter tipo **int***.

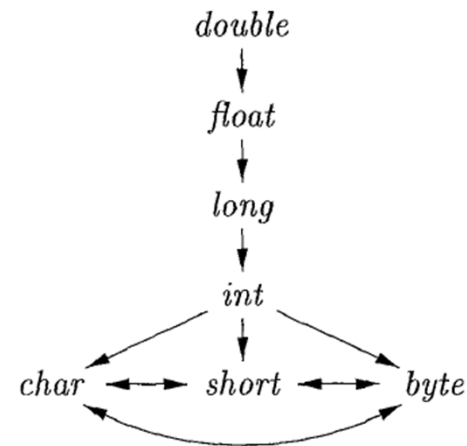
Análise Semântica: Verificação de Tipos

Verificação de tipos: Regras de Inferência

Se duas expressões E_1 e E_2 possuem tipos diferentes, pode ser necessário converter um deles, para verificar a validade da operação entre as expressões. Ex.:



(a) Widening conversions



(b) Narrowing conversions

Análise Semântica: Verificação de Tipos

Verificação de tipos

A verificação de tipos é feita realizando-se um percurso em pós-ordem na AST, ou seja, das folhas até a raiz, onde os tipos são passados dos nós filhos para os nós pais.

Análise Semântica: Verificação de Tipos

Como seria a verificação de tipos para:

```
void foo(int a, char b)
{
    int i, *p;
    float f;

    i = a + b + *p;
    f = i + a + b;
    i = p + a;
    i = p + (f + a);
}
```