Programação de Computadores:

Subprogramação: Sub-rotinas e Funções

Luis Martí
Instituto de Computação
Universidade Federal Fluminense
lmarti@ic.uff.br - http://lmarti.com

Considere o Programa a Seguir

```
program p17codre

real a, b, c

read *, a, b
c = a * b / 2.0
print *, 'a vale: ', a
print *, 'b vale: ', b
print *, 'a * b / 2.0 vale: ', c

a = a / 2.0
b = b / 2.0
c = a * b / 2.0
print *, 'a vale: ', a
print *, 'b vale: ', b
print *, 'a * b / 2.0 vale: ', c

end
```

É muito comum que vários trechos de código se repitam ao longo de um programa.

Como fazer para economizar código, evitando a repetição?

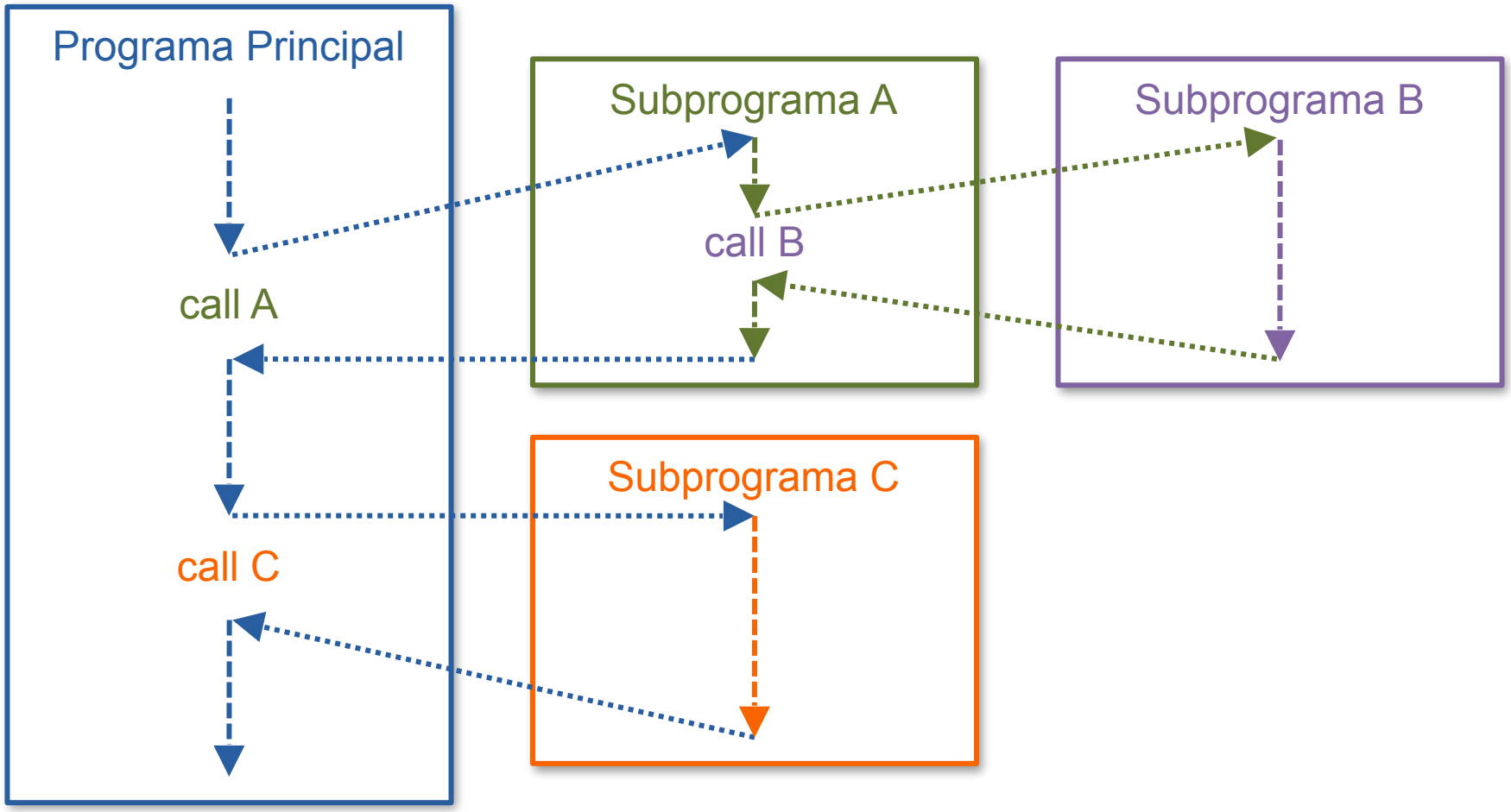
Subprogramas

- Pedaços de código que são chamadas pelo programa principal ou por outros subprogramas
- Vantagens
  - Defeitos podem ser facilmente corrigidos, pois as modificações são feitas em um único local
  - Defeitos podem ser evitados, pois a replicação de código pode levar a modificações incompletas
  - O código fica mais legível
  - Facilita o design de algoritmos, pois um problema é dividido em subproblemas

Exemplo de Subprogramação em FORTRAN

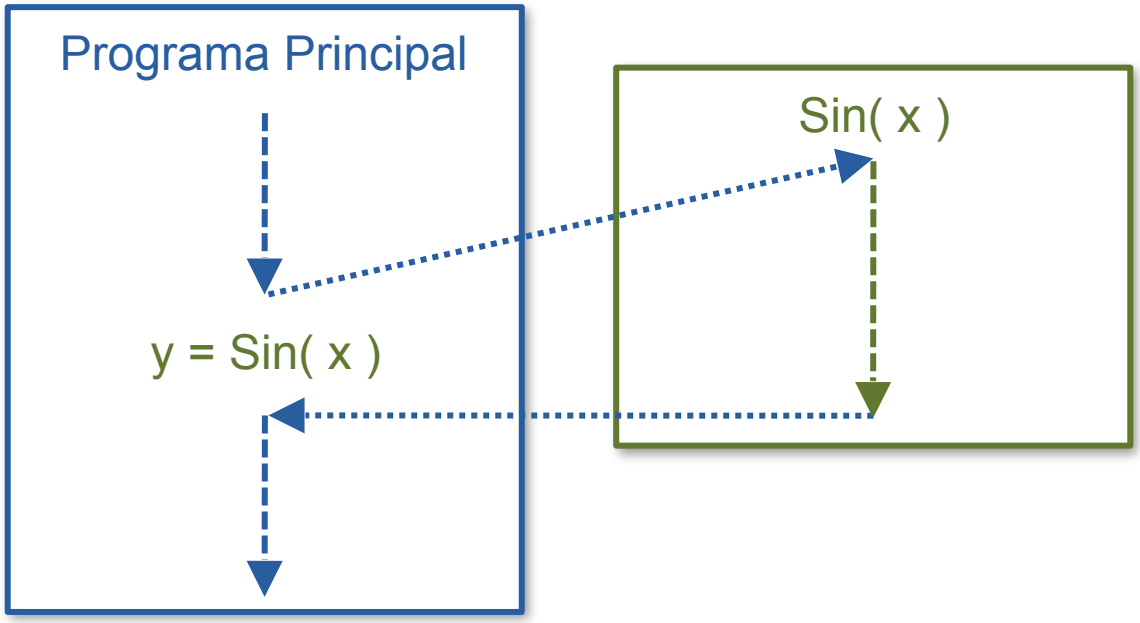
```
program p18subpr

real a, b
read *, a, b
call mostra(a, b)
a = a / 2.0
b = b / 2.0
call mostra(a, b)

end


subroutine mostra(a, b)

real a, b, c
c = a * b / 2.0
print *, 'a vale: ', a
print *, 'b vale: ', b
print *, 'a * b / 2.0 vale: ', c

end
```

Fluxo de Controle



Fluxo de Controle

- O mesmo acontece quando usamos uma das funções pré-definidas do FORTRAN

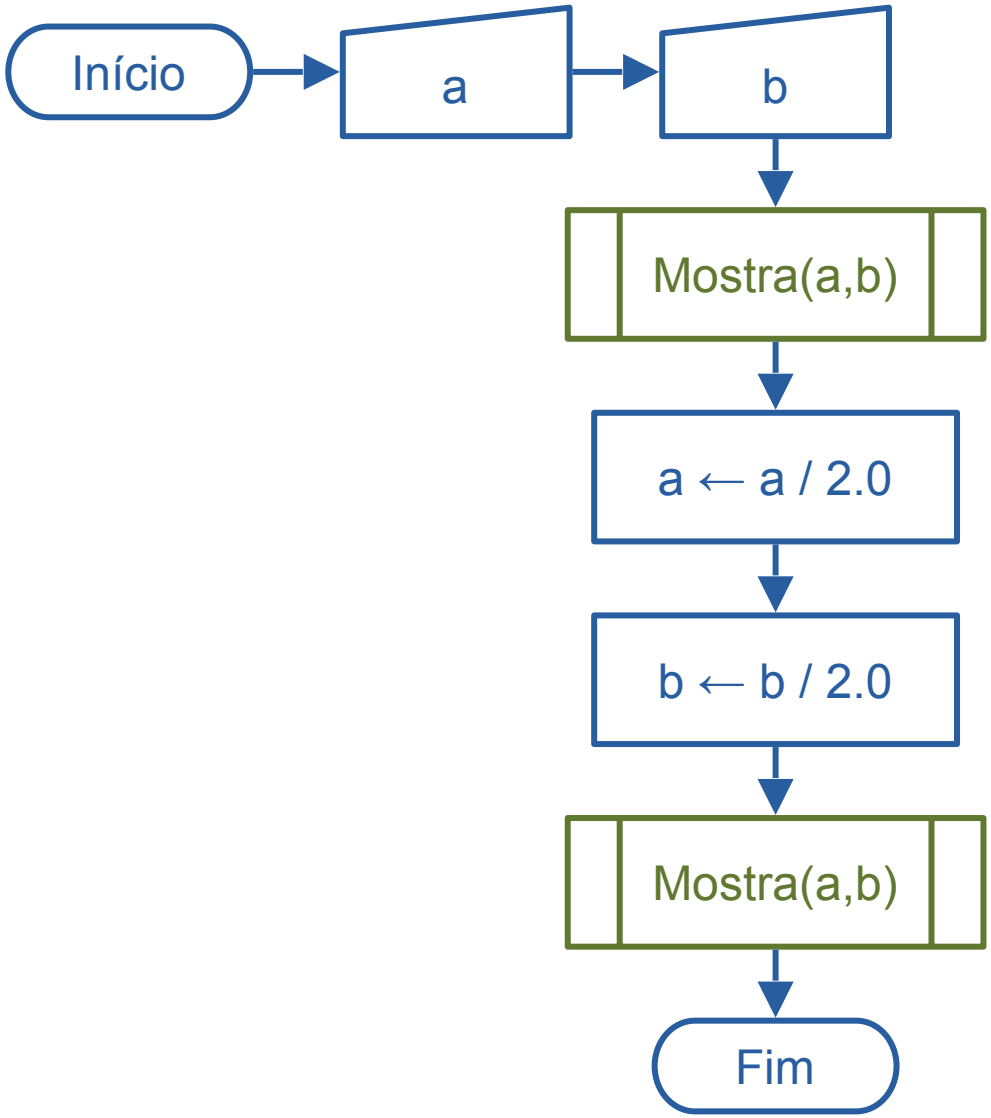


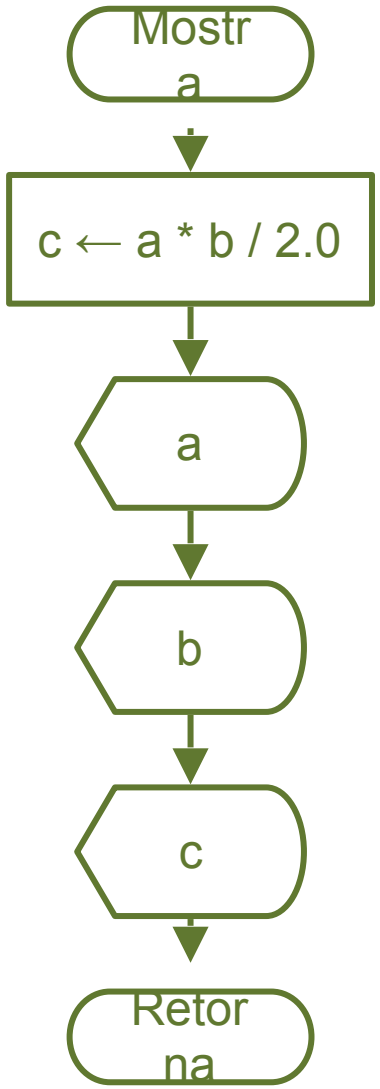
Fluxograma

- Um novo símbolo é utilizado na representação de subprograma chamado em um fluxograma

- O subprograma passa a ter um fluxograma próprio
  - O símbolo de início contém o nome do subprograma
  - O símbolo de término contém a palavra "Retorno"

Exemplo de Fluxogramas





Tipos de Subprogramas

- Função
  - Sempre retorna um valor
  - A chamada é feita pelo identificador (nome) da função e argumentos entre parênteses, separados por vírgula
    - Exemplo: **`sin(x)`**
  - O valor retornado é do tipo especificado na declaração da função
- Sub-rotina
  - Não retorna um valor, necessariamente
  - A chamada é feita por CALL seguido pelo identificado da sub-rotina e argumentos entre parênteses, separados por vírgula
    - Exemplo: **`call mostra(a,b)`**

Exemplo de Função e de Sub-Rotina

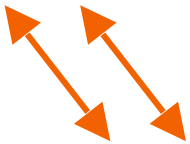
Veja o arquivo p19fusu.f

Observações Importantes!

1. O nome dos argumentos de funções e sub-rotinas não precisa ser igual ao nome das variáveis

   O FORTRAN faz o casamento dos nomes

```
call troca(a,b)

subroutine troca(x,y)
```



2. Modificações no valor dos argumentos são visíveis fora do subprograma

Observações Importantes!

3. Variáveis declaradas no programa principal não podem ser acessadas pelo subprograma, apenas aquelas passadas como argumento
4. Variáveis declaradas no subprograma não podem ser acessadas pelo programa principal
5. Funções devem ser declaradas no programa principal de forma semelhante a uma variável

SUBROUTINE

- Declaração ocorre logo após o END do programa principal
- Sintaxe

```
subroutine <nome>(<arg1>, <arg2>, ..., <argn>)

  <declaração dos tipos do parâmetros>
  <declaração de variáveis locais>

  <corpo do subprograma>

end
```

Exemplos Simples de SUBROUTINE

```
subroutine s1(a)
  integer a
  ...
end
```

```
subroutine s2(a,b)
  integer a, b
  ...
end
```

```
subroutine s3(a,b)
  real a
  integer b
  ...
end
```

```
subroutine s4
  ...
end
```

SUBROUTINE

- Declaração de variáveis em sub-rotina
  - As variáveis declaradas dentro de uma sub-rotina pertencem somente a ela (variáveis locais)
- Passagem de argumentos por referência
  - A mesma região de memória do programa principal é utilizada pelos argumentos da sub-rotina
  - Alterações no valor das variáveis passadas como argumento são refletidas no programa principal

SUBROUTINE

- Chamada feita do programa principal, ou de outro subprograma, com o comando CALL
- Sintaxe

```
call <nome>(<v_arg1>, <v_arg2>, ..., <v_argn>)
```

- `v_argi` é o valor passado como argumento, pode ser um valor constante ou uma variável
  - Se for modificado dentro da sub-rotida, `v_argi` deverá ser uma variável

Exemplo de SUBROUTINE

Veja o arquivo p20tempe.f

FUNCTION

- Declaração ocorre logo após o END do programa principal
- Sintaxe

```
<tipo de retorno> function <nome>(<arg1>, ...)

  <declaração dos tipos do parâmetros>
  <declaração de variáveis locais>

  <corpo do subprograma>

end
```

Exemplos Simples de FUNCTION

```
integer function f1(a)
  integer a
  ...
  f1 = ...
end

real function f2(a,b)
  real a
  integer b
  ...
  f2 = ...
end
```

O FORTRAN cria automaticamente uma variável com o nome da função, que deve ser utilizada para definir o valor de retorno

FUNCTION

- Chamada feita do programa principal, ou de outro subprograma
- Precisa ser declarada como uma variável
- Retorna um único valor
- Sintaxe

```
var = <nome>(<v_arg1>, <v_arg2>, ..., <v_argn>)
```

Exemplo de FUNCTION

Veja o arquivo p21volum.f

Material adaptado por Luis Martí a partir dos slides de Leandro Augusto Frata Fernandes.