

# Programação Concorrente

## Programação Paralela e Distribuída

### *Conceito de Programa e Execução*



Graduação em Ciência da Computação  
Universidade do Vale do Rio dos Sinos

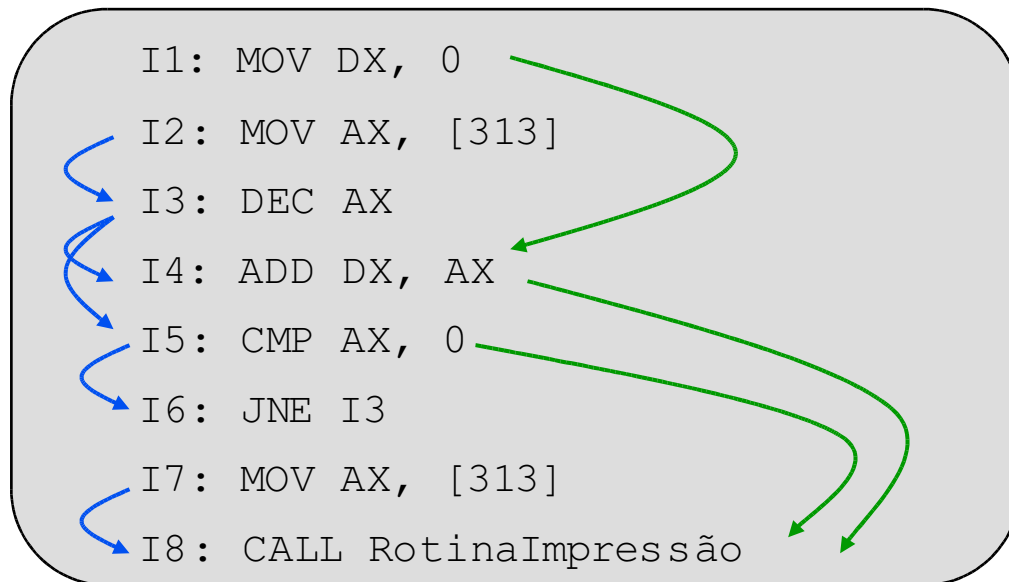
# Programa

## Compreensão do conceito de *programa*

- O que é um programa ?
- O que é concorrência ?
- O que é programação concorrente ?
- Concorrência e paralelismo representam a mesma coisa ?
- Programar concorrente é diferente de programar seqüencial ?
- O que é uma tarefa ?
- O que é sincronização ?

# O que é um programa ?

Um conjunto de *instruções* ao indicando uma  
ação a ser executada

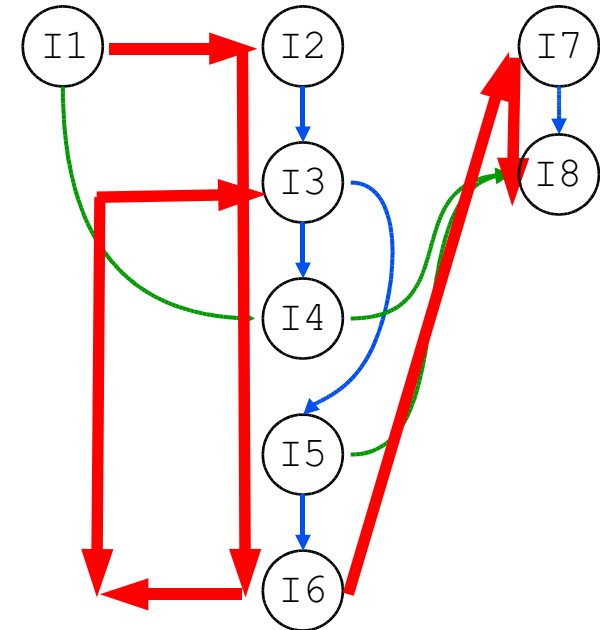
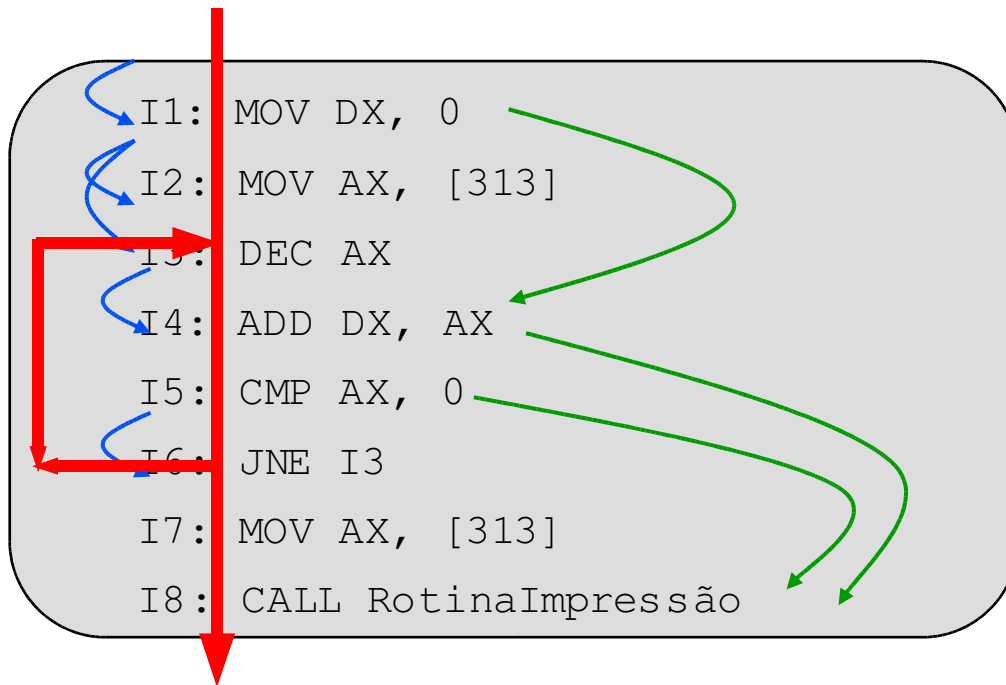


**Efeito colateral**

Escrita em memória

# O que é um programa ?

Um conjunto de *instruções* ao indicando uma  
ação a ser executada



Controle **semântico** através de sincronizações entre instruções

# O que é concorrência ?

- Tradicionalmente:

- Disputa por recursos.

- E também:

- Concorrer: juntar-se para uma ação ou fim comum; contribuir, cooperar.

- Outra abordagem:

- Independência temporal

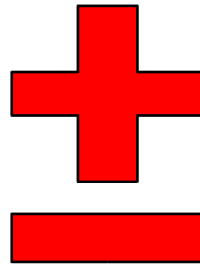
# O que é programação concorrente ?

Técnica de programação que explora a **independência temporal** de *atividades* definidas por uma aplicação.

**Compartilhamento**  
de dados

**Cooperação**  
para evolução da execução

# Concorrência e paralelismo representam a mesma coisa ?



Atividades paralelas são atividades concorrentes,  
contudo são tratadas sobre conjuntos de recursos  
*independentes*.

# Programar concorrente é diferente de programar seqüencial ?

**NÃO**

O problema da programação continua sendo: dado um problema, **definir as instruções a serem executadas pelo computador**

**SIM**

Novos itens devem ser tratados, como **identificar as atividades concorrentes**, o **compartilhamento dos recursos** (dados) e a **estrutura da colaboração** entre tarefas

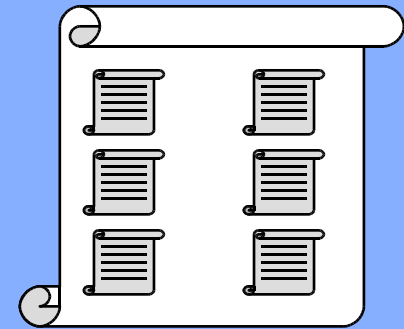
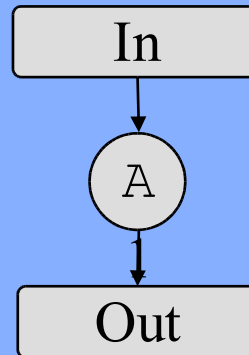
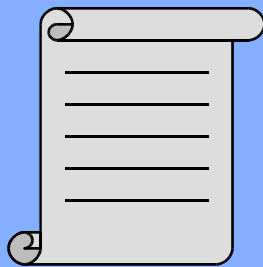


# O que é tarefa ?

## Tarefa

Unidade de trabalho

Conjunto de instruções que descrevem uma tarefa específica no contexto de um programa.



Uma instrução (ou uma função, ...) na programação seqüencial.  
Uma abstração na programação concorrente.

# O que é sincronização ?

## Sincronização

Troca de informações entre duas tarefas

Mecanismo que permite controlar o compartilhamento de dados entre tarefas.

Comunicação

Implícita na programação seqüencial.  
Explícita na programação concorrente.

# Mecanismos de Sincronização

## Relações de Dependência

### Fraca

Não existe uma ordem específica no acesso aos dados. A execução do programa será correta **independente da ordem** em que as operações forem realizadas.

### Exemplos

- Compartilhamento de buffer (produtor/consumidor)
- Caixa de banco

# Mecanismos de Sincronização

## Relações de Dependência

### Forte

Existe uma ordem específica no acesso aos dados. A execução do programa será correta somente se **a ordem definida for respeitada**. Esta dependência identifica uma relação de entrada e saída de dados entre as tarefas.

### Exemplos

- Programação seqüencial
- Programação dinâmica
- Métodos iterativos
- Cálculos com matrizes
- ...

# Mecanismos de Sincronização

## Relações de Dependência

### Predição de desempenho

**Fraca:** Não é possível prever o momento em que uma tarefa será executada

**Forte:** É possível conhecer o instante em que uma tarefa será executada a partir do conhecimento de suas dependências

### Resultado final

**Fraca:** Dependente de decisões tomadas fora do programa, refletindo no escalonamento das tarefas

**Forte:** Dependente da ordem estabelecida no programa para execução das tarefas

# Tarefa e Sincronização

## Exemplo 1

```
Baskara( IN a, b, c; OUT x1, x2 ) {  
    ...  
    x1 = (-b + sqrt(b2 - 4*a*c)) / 2*a;  
    x2 = (-b - sqrt(b2 - 4*a*c)) / 2*a;  
    Sync;  
    ...  
}
```

**Tarefas**

**Sincronização**

# Tarefa e Sincronização

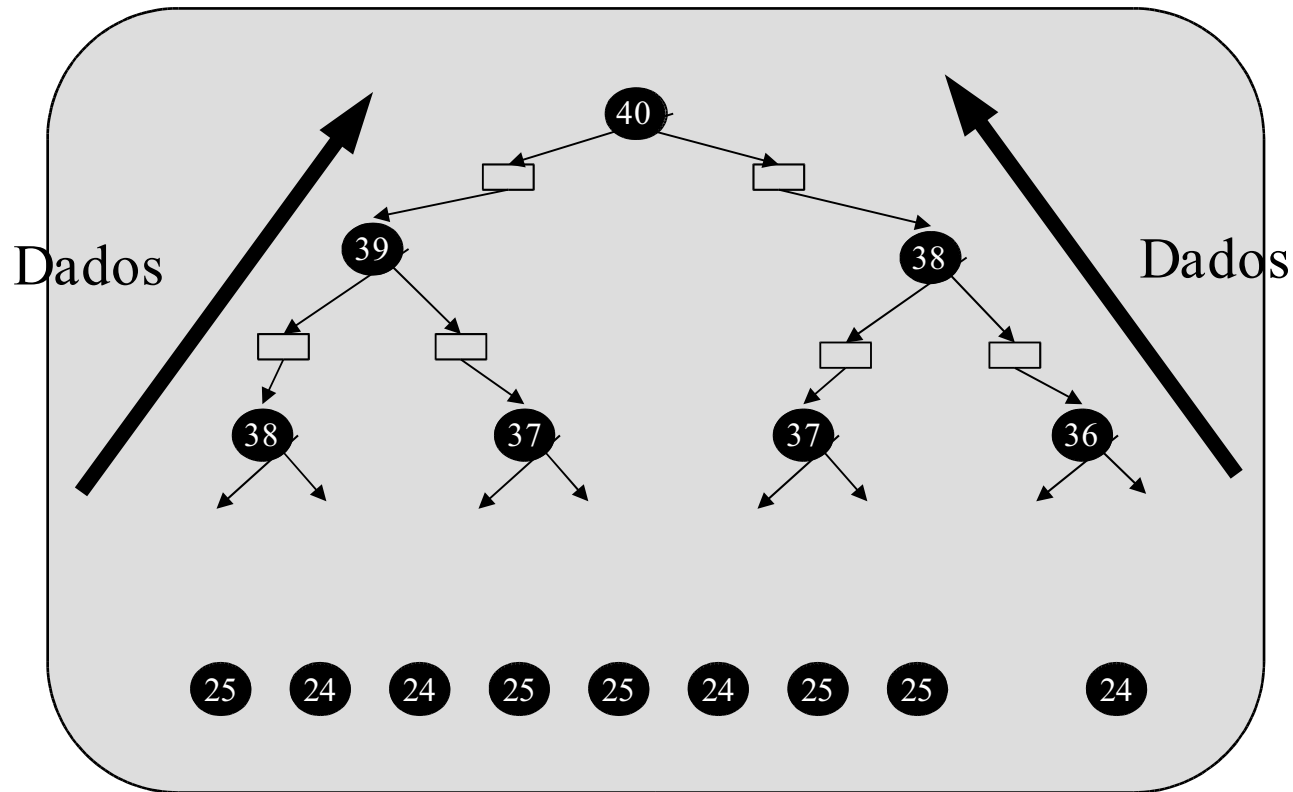
## Exemplo 2

```
Fibonacci( IN n, OUT res ) {  
    if( n <= 2 ) res = 1;  
    else {  
        Fibonacci( n - 1, r1 );  
        Fibonacci( n - 2, r2 );  
        Sync;  
        res = r1 + r2;  
    }  
}
```

Sincronização

# Tarefa e Sincronização

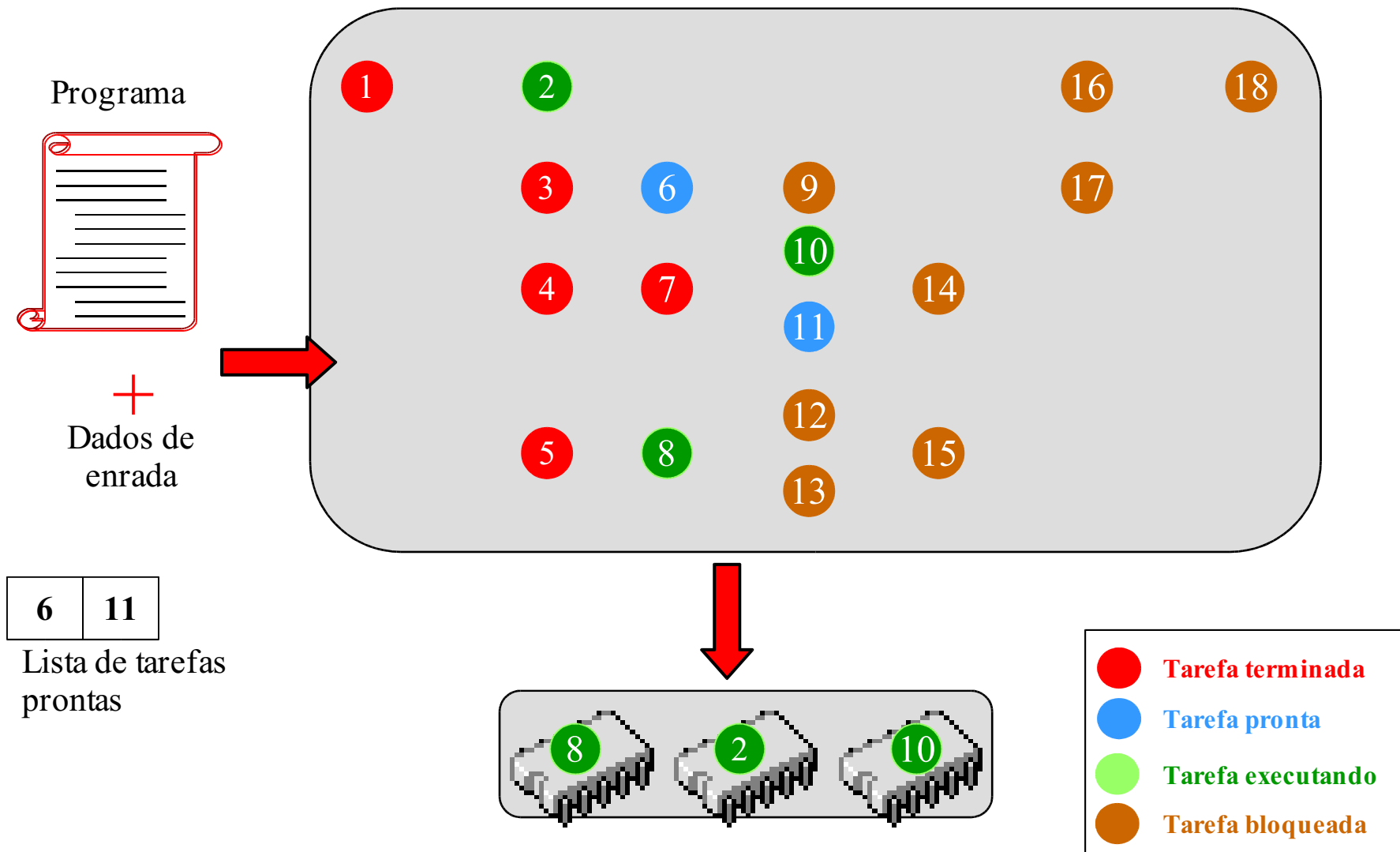
## Exemplo 2





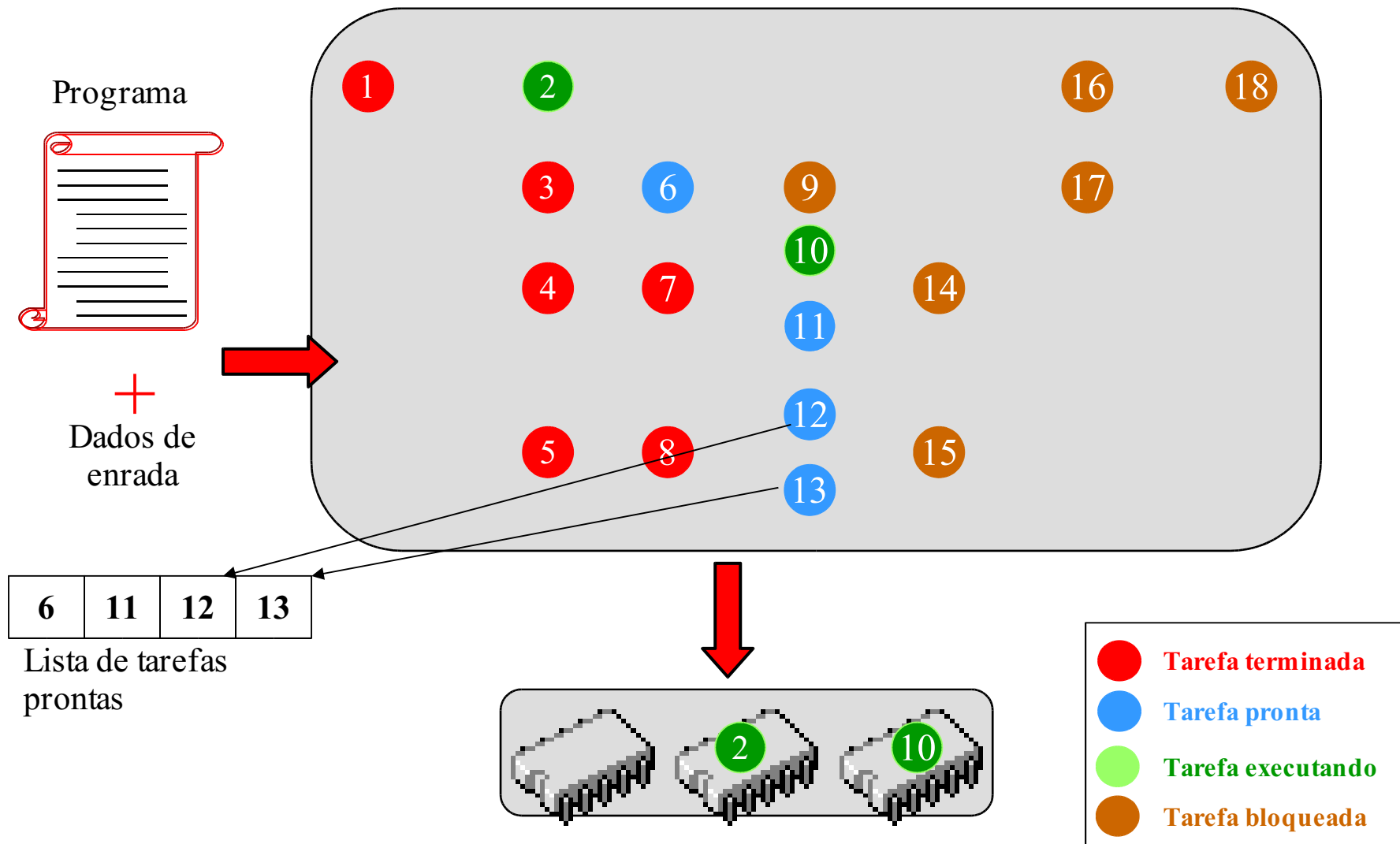
# Tarefa e Sincronização

## Generalizando



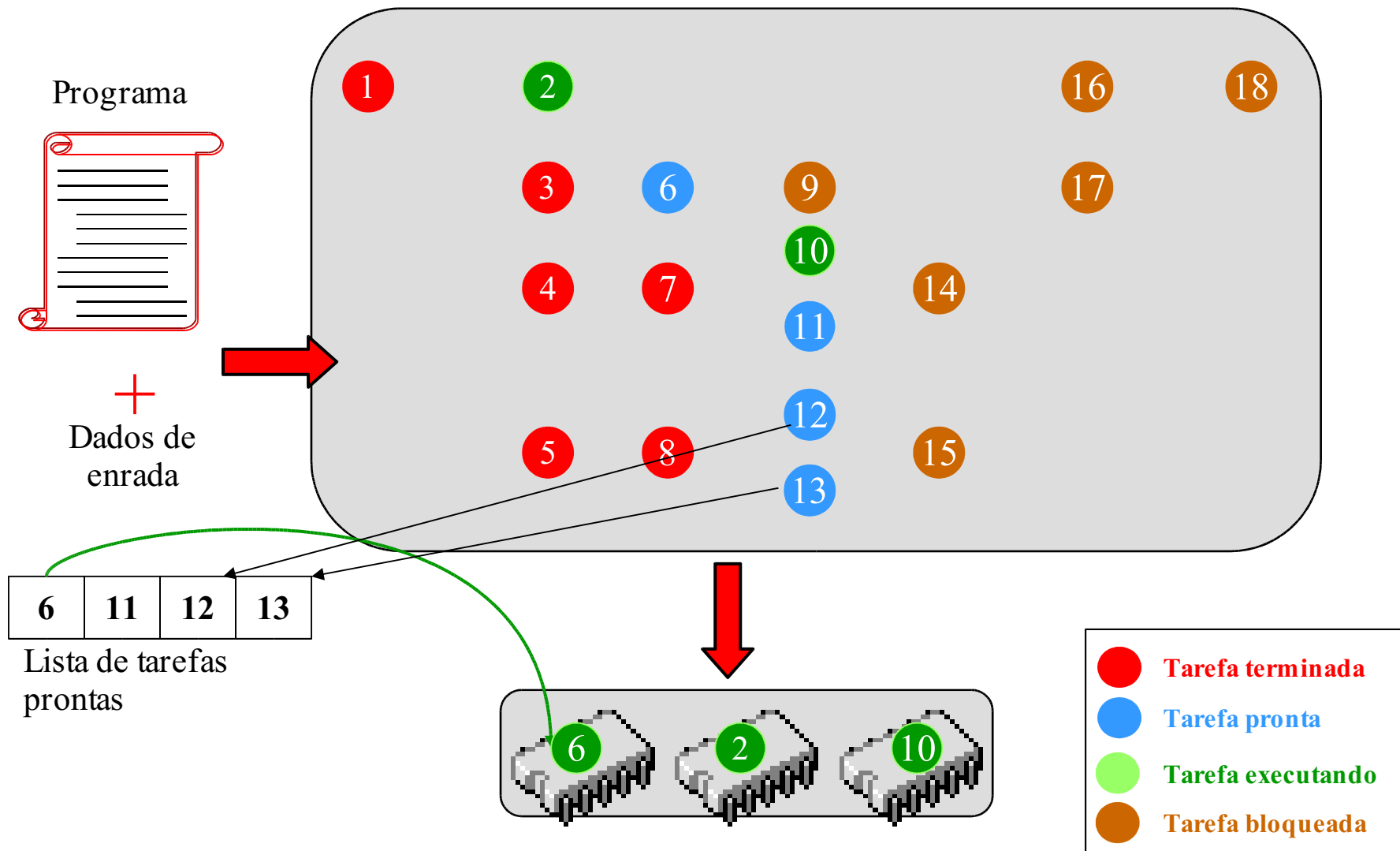
# Tarefa e Sincronização

## Generalizando



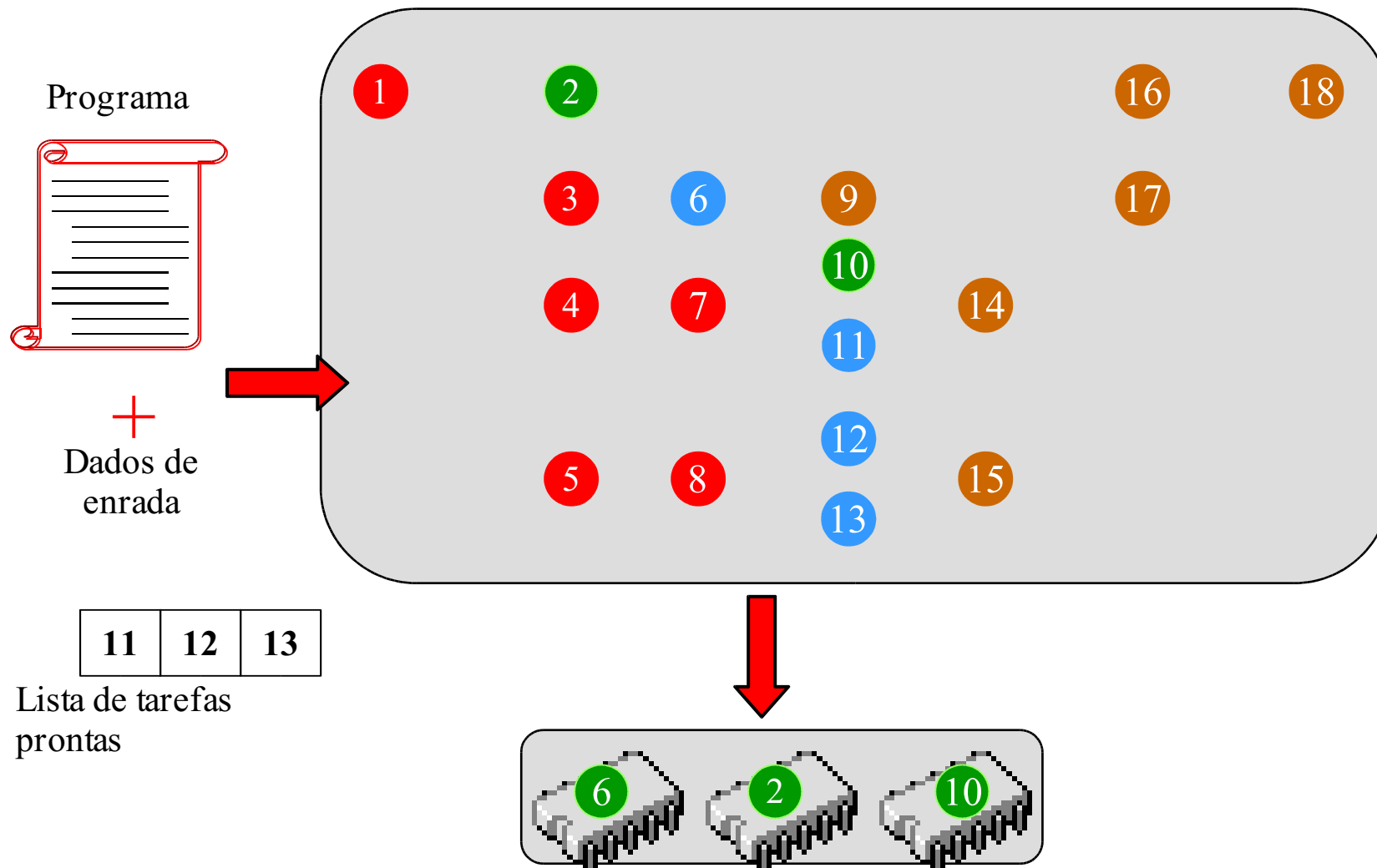
# Tarefa e Sincronização

## Generalizando



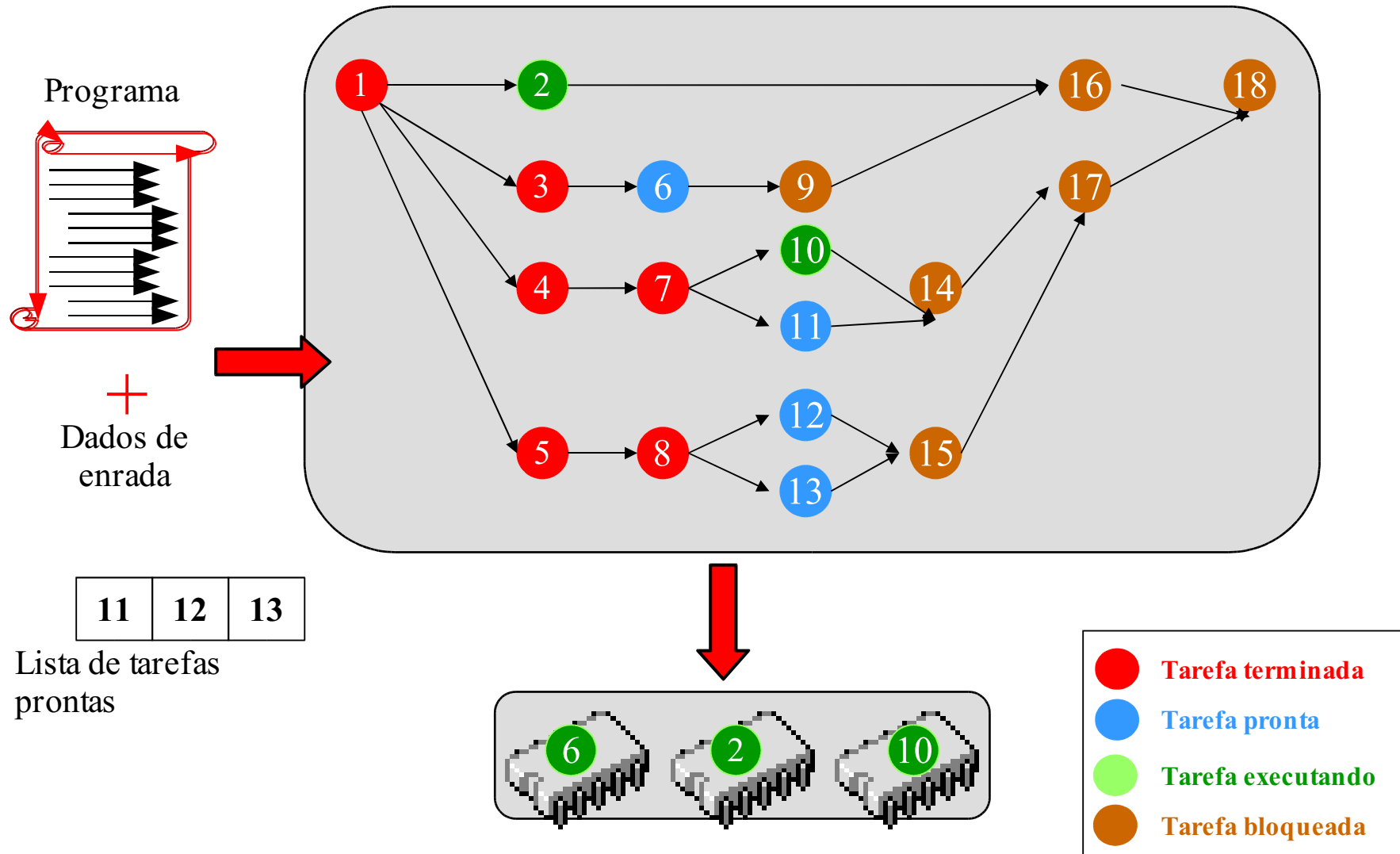
# Tarefa e Sincronização

## Generalizando



# Tarefa e Sincronização

## Generalizando



# Tarefa e Sincronização

## Generalizando

