

Capítulo 5: Algoritmos

**Ciência da computação: Uma visão abrangente
11a Edition**

**Autor:
J. Glenn Brookshear**

Capítulo 5: algoritmos

- 5.1 O conceito de um algoritmo
- 5.2 Representação de algoritmo
- 5.3 Descoberta de algoritmo
- 5.4 Estruturas iterativas
- 5.5 Estruturas recursivas
- 5.6 Eficiência e correção

Definição de algoritmo

Um algoritmo é um conjunto **ordenado** de passos **executáveis** **não-ambíguos** que definem um processo finalizável.

Representação de algoritmo

- Requer primitivas bem-definidas
- Uma coleção de primitivas constitui uma linguagem de programação.

Figura 5.2 Produzir um pássaro através do dobramento de pedaço quadrado de papel

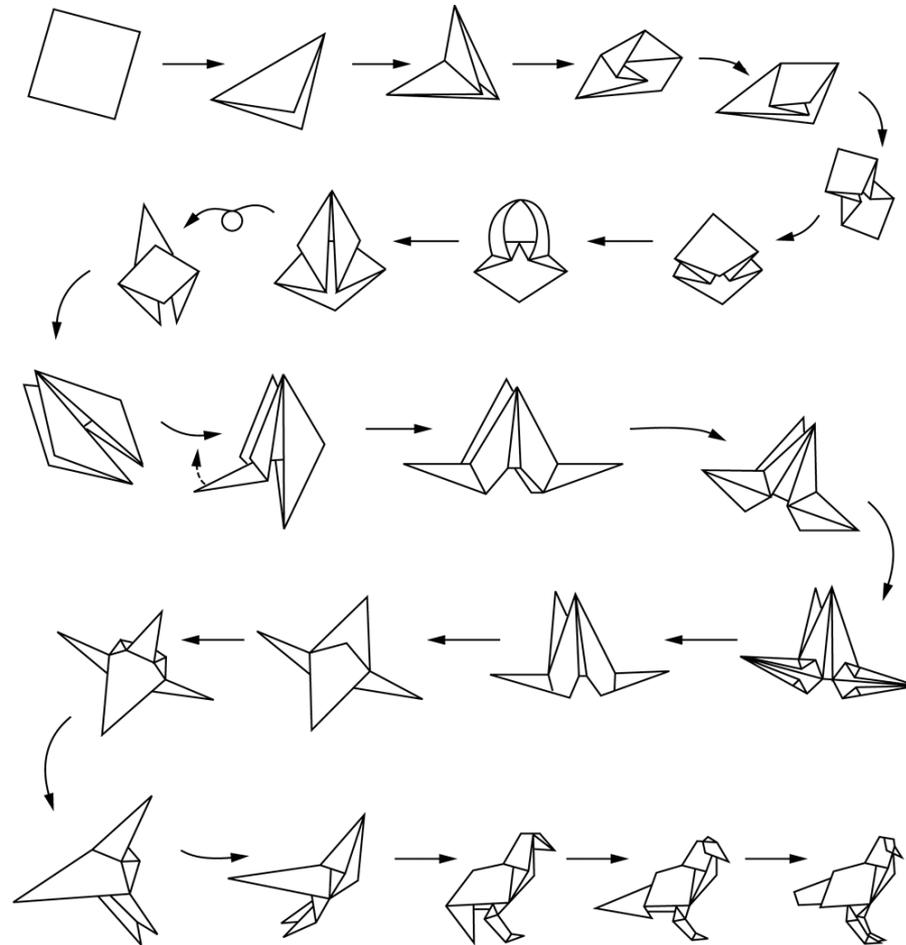
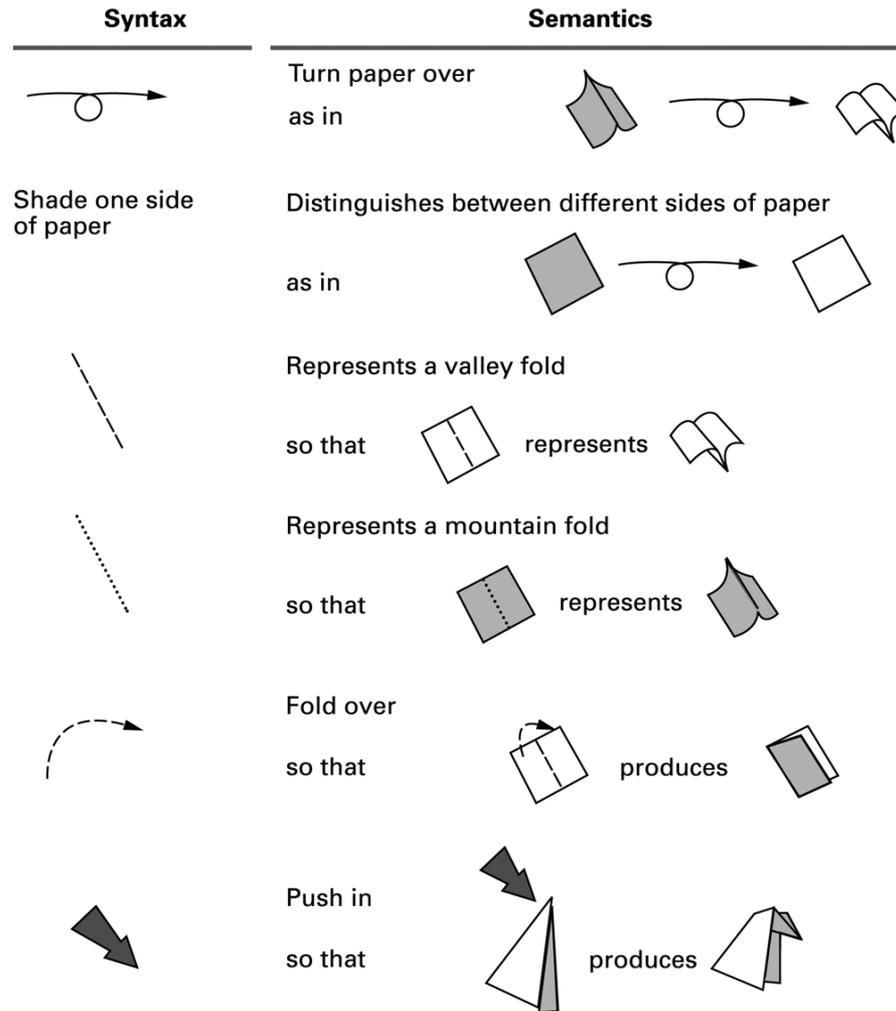


Figura 5.3 Primitivas de Origami



Primitivas de pseudocódigo

- Assignment (Atribuição)

nome ← *expressão*

- Seleção condicional

if *condição* **then** *ação* (*se – então*)

Primitivas de Pseudocódigo

- Execução repetida

while *condição do ação (enquanto – faça)*

- Procedure (procedimento)

procedure *name (nomes genéricos – não usar primitivas)*

Figura 5.4 Procedimento em pseudocódigo de saudações (hello)

procedure Greetings

Count \leftarrow 3;

while (Count > 0) **do**

(print the message "Hello" and

Count \leftarrow Count - 1)

Passos de resolução de problemas de Polya

- 1. Entenda o problema.
- 2. Elabore um plano para resolver o problema.
- 3. Realize o plano.
- 4. Avaliar a solução quanto à precisão e o seu potencial como uma ferramenta para resolver outros problemas.

Tradução para o contexto da programação

1. Entender o problema
2. Idealizar um algoritmo que possa resolver o problema
3. Formular o algoritmo e implementá-lo
4. Avaliar a precisão dos resultados e o potencial para resolver outros problemas

O Problema das idades das crianças

- Pessoa é encarregada da tarefa de determinar as idades dos três filhos do B.
 - B diz a A que o produto da idade das crianças é 36.
 - A responde que outra pista é necessária.
 - B diz a A a soma das idades das crianças.
 - A responde que outra pista é necessária.
 - B diz a A que o filho mais velho toca piano.
 - A descobre e diz a idade de cada criança.
- Quantos anos tem as três crianças?

Figure 5.5

a. Triples whose product is 36

$$(1,1,36) \quad (1,6,6)$$

$$(1,2,18) \quad (2,2,9)$$

$$(1,3,12) \quad (2,3,6)$$

$$(1,4,9) \quad (3,3,4)$$

b. Sums of triples from part (a)

$$1 + 1 + 36 = 38$$

$$1 + 2 + 18 = 21$$

$$1 + 3 + 12 = 16$$

$$1 + 4 + 9 = 14$$

$$1 + 6 + 6 = 13$$

$$2 + 2 + 9 = 13$$

$$2 + 3 + 6 = 11$$

$$3 + 3 + 4 = 10$$

Estruturas iterativas

- Laço pré-teste
 - enquanto (condição) faça (ação):
while (*condition*) **do**
(loop body)
- Laço pós-teste:
 - Repetir (ação) até (condição)
repeat (*loop body*)
until (*condition*)

O algoritmo de busca sequencial em pseudocódigo

Procedimento Buscar (*Lista*, *ValorAlvo*)

Se (*Lista* Vazia)

Então

(Declare que a busca falhou)

Senão

(Selecione o primeiro da lista como *EntradadeTeste*)

Enquanto (*ValorAlvo* > *EntradadeTeste* **E** houver entradas em *Lista* a testar)

Faça (Selecione a próxima entrada de *Lista* como *EntradadeTeste*)

Se (*ValorAlvo* = *EntradadeTeste*)

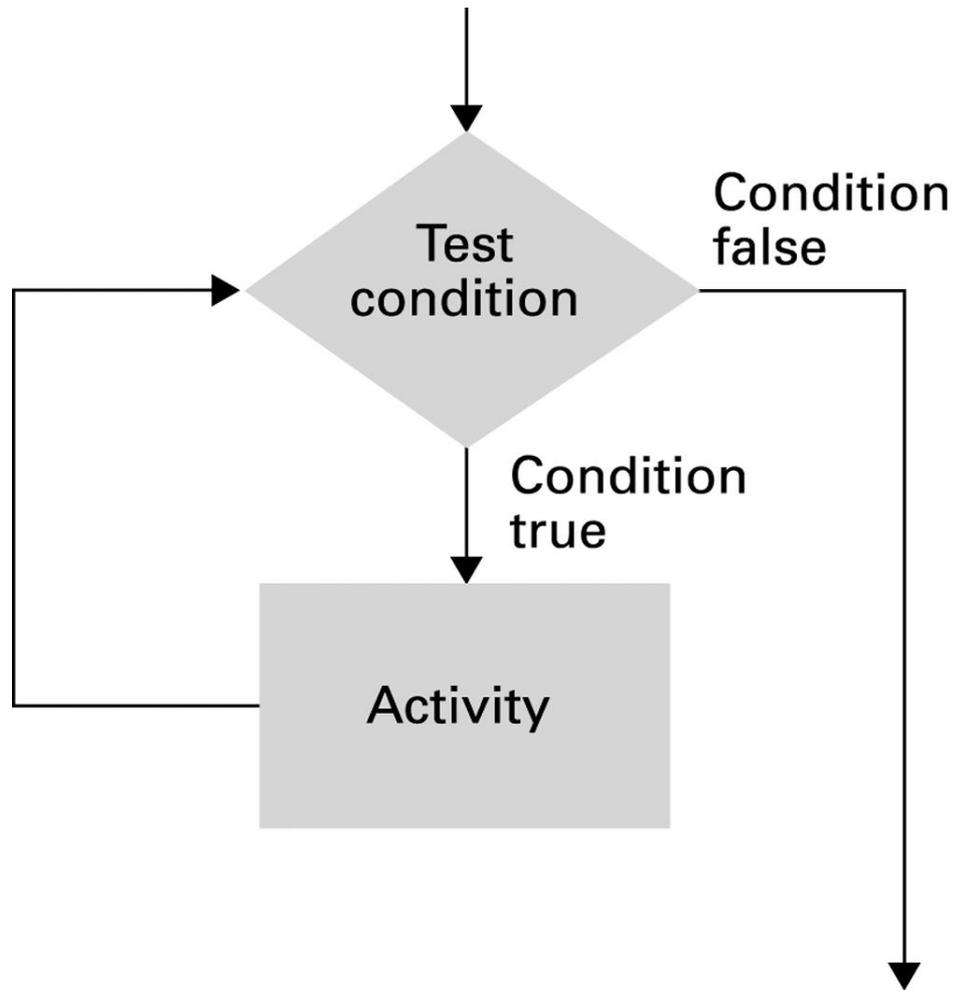
Então (Declare que a busca foi bem sucedida)

Senão (Declare que a busca falhou)

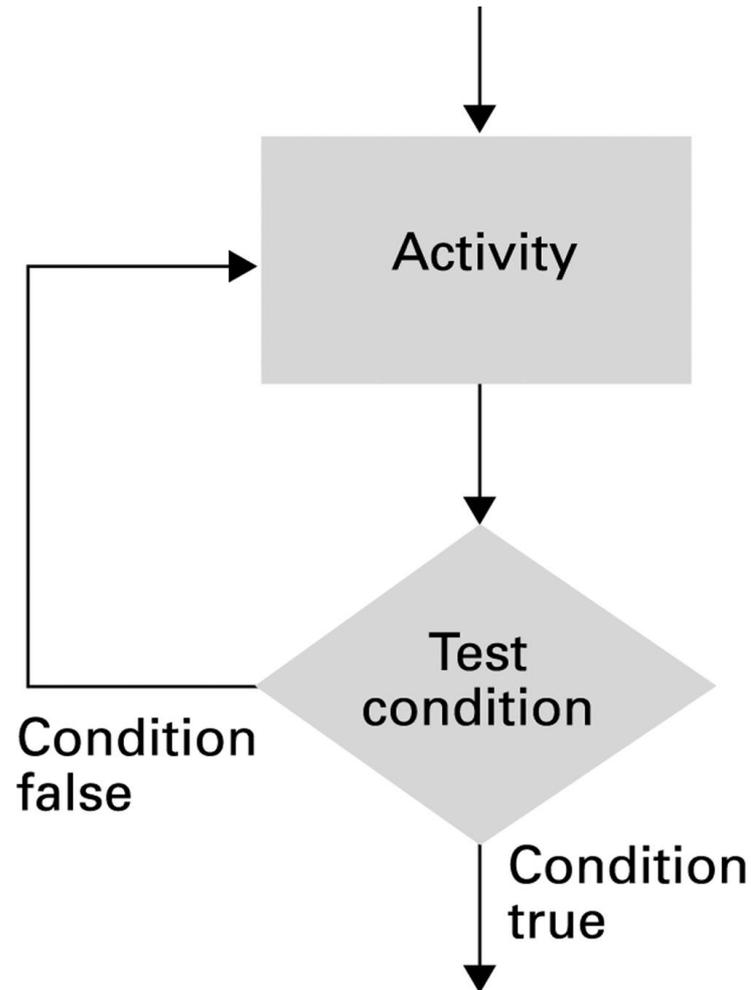
Componentes de Controle Repetitivo

- **Inicializar** – Estabelecer um estado inicial que será modificado em direção à condição de término
- **Testar** – Comparar o estado atual com a condição de término e terminar a repetição se a condição for alcançada
- **Modificar** – Modificar o estado de forma que ele de mova para a condição de término

A estrutura do laço (loop) FAÇA – ENQUANTO (while - do)



A estrutura do laço de repetição REPETIR ATÉ – repeat until)



Recursividade

- A execução de um procedimento leva a outra execução do procedimento.
- São executadas múltiplas ativações do procedimento, mas uma espera a outra terminar para prosseguir.