

## Fluxograma

### Símbolos inicial e terminal

indica os pontos de início e de fim do fluxograma



### Símbolos de entrada/leitura e saída/escrita (estruturas de sequência)

indica passos envolvendo troca de dados com o exterior



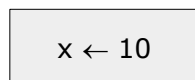
### Símbolo de sentido do fluxo

indica a origem e o destino do fluxo



### Símbolo de processamento (estruturas de sequência)

indica cálculo ou manuseio de dados



### Símbolo de seleção (estruturas de seleção e de repetição)

indica a seleção de uma das 2 ações, segundo o resultado de uma condição lógica (**sim** ou **não**)



### Símbolo de conexão

indica a ligação de dois pontos do fluxograma (pode-se não usar)



### Pseudocódigo

Um algoritmo deve iniciar-se com

```
algoritmo <nome-do-algoritmo>
```

Um algoritmo deve terminar com

```
fim_algoritmo
```

Os dados necessários à execução do algoritmo são indicados pelo comando (opcional)

```
parâmetros de entrada: <lista-de-variáveis>
```

onde <lista-de-variáveis> são os nomes das variáveis que recebem os dados iniciais

Os dados determinados (obtidos) pelo algoritmo são indicados pelo comando (opcional)

```
parâmetros de saída: <lista-de-variáveis>
```

onde <lista-de-variáveis> são os nomes das variáveis que recebem os resultados

A escrita de comentários para clarificar o algoritmo (parte não executável)

```
{ comentários }
```

O símbolo ← é usado para atribuir o resultado de uma expressão a uma variável

```
<variável> ← <expressão>
```

#### Estrutura de entrada/leitura de dados

```
ler: <lista-de-variáveis>
```

indica a <lista-de-variáveis> que está disponível para receber os dados iniciais (valores)

#### Estrutura de saída/escrita de dados

```
escrever: <lista-de-variáveis, mensagens>
```

para indicar quais e onde certos valores devem ser escritos e mostrar mensagens

#### Estrutura condicional simples

```
se <condição> então
```

```
    <comandos>
```

```
fim_se
```

#### Estrutura condicional composta

```
se <condição> então
```

```
    <comandos_1>
```

```
senão
```

```
    <comandos_2>
```

```
fim_se
```

**Estrutura de repetição** que verifica a condição à entrada do ciclo

```
enquanto <condição> fazer
```

```
    <comandos>
```

```
fim_enquanto
```

**Estrutura de repetição** que verifica a condição à saída do ciclo

```
fazer
```

```
    <comandos>
```

```
enquanto <condição>
```