

# Estruturas condicionais

Marina Andretta

ICMC-USP

2 de março de 2016

Em pseudo-código, a sintaxe do comando **se** é:

**se** (<condição>) **entao**

<comando 1>

<comando 2>

⋮

<comando k>

**senao**

<comando k+1>

<comando k+2>

⋮

<comando n>

**fimse**

A <condição> deve ser uma expressão lógica, ou seja, deve ter como resultado um valor verdadeiro ou falso.

Se o resultado da expressão for **verdadeiro**, os comandos <comando 1>, <comando 2>, ..., <comando k> são executados (nesta ordem) e, depois de executados, passa-se ao **fimse**.

Se o resultado da expressão for **falso**, apenas os comandos <comando k+1>, <comando k+2>, ..., <comando n> são executados (nesta ordem) e, depois de executados, passa-se ao **fimse**.

Um exemplo do uso do comando **se** é o seguinte:

```
se (x >= 5) entao  
    escreva('0 aluno passou direto')  
senao  
    escreva('0 aluno ainda nao passou')  
fimse
```

# Comando **if**

Em linguagem C, o equivalente ao **se** é o comando **if**, que tem a seguinte sintaxe:

```
if (<condição>) {  
    <comando 1>;  
    <comando 2>;  
    ⋮  
    <comando k>;  
}  
else {  
    <comando k+1>;  
    <comando k+2>;  
    ⋮  
    <comando n>;  
}
```

O exemplo dado em pseudo-código traduzido para linguagem C fica:

```
if (x < 0) {  
    printf(“0 numero eh negativo”);  
}  
else {  
    printf(“0 numero eh positivo”);  
}
```

Em algumas linguagens de programação, como em C, existe um comando de seleção. No caso da linguagem C, ele é chamado de **`case`**.

Estes comandos, que podem ser substituídos por uma sequência de comandos **`if`**, verificam qual o valor de uma variável (que, no caso do C, deve ser do tipo inteiro ou caracter) e, para cada valor diferente, define uma sequência de comandos a serem executados.

# Comando case

```
switch (<variável>) {  
    case <valor 1>:  
        <comando 1>;  
        ⋮  
        <comando k>;  
        break;  
        ⋮  
    default:  
        <comando j+1>;  
        ⋮  
        <comando n>;  
}
```

# Exemplo

```
switch (op) {  
    case 'w':  
        printf("para cima");  
        break;  
    case 's':  
        printf("para baixo");  
        break;  
    case 'a':  
        printf("para a esquerda");  
        break;  
    case 'd':  
        printf("para a direita");  
        break;  
    default:  
        printf("comando invalido");  
}
```

# Exemplo de um algoritmo completo: em pseudo-código

Vamos escrever um algoritmo, em pseudo-código, para resolver o seguinte problema: dadas as 2 notas de um aluno, escrever uma mensagem dizendo se o aluno foi aprovado ou não.

A média usada deve ser a aritmética e um aluno é considerado aprovado se sua média for maior ou igual a 5.

# Exemplo de um algoritmo completo: em pseudo-código

```
programa media
  declaracoes
    real nota1, nota2, media
inicio
  escreva(“Digite as duas notas:  ”)
  leia(nota1, nota2)
  media ← (nota1 + nota2)/2
  se (media >= 5) entao
    escreva(“0 aluno foi aprovado”)
  senao
    escreva(“0 aluno ainda nao foi aprovado”)
  fimse
fim
```

## Exemplo do mesmo algoritmo completo: em linguagem C

```
int main () {
    float nota1, nota2, media;

    printf("Digite as duas notas: ");
    scanf("%f %f", &nota1, &nota2);

    media = (nota1 + nota2)/2;

    if (media >= 5) {
        printf("O aluno foi aprovado");
    }
    else {
        printf("O aluno ainda nao foi aprovado");
    }

    return(0);
}
```