

# LP1

*Prof. Luciano Bernardes de Paula*



## Estruturas de decisão

As estruturas de decisão são utilizadas para se criar condições para a execução ou não de um determinado trecho de código.



## Exemplo:

Um programa que recebe a média do aluno e retorna “Aprovado” ou “Reprovado” segundo o seguinte critério:

Se a média é igual ou maior que 6,0, o aluno está “Aprovado”, caso contrário, o aluno está “Reprovado” .



## Estrutura de decisão *if* (“se”)

Sintaxe

```
if (exp){  
    comandos localizados aqui serão  
    executados se exp é verdadeira  
}
```



## Exemplo

...

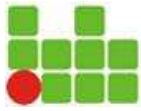
```
if (var > 10){  
    printf("O valor de var é maior que  
    10");  
}
```

...



## Bloco if-else (*se-senão*)

```
if(exp1){  
    comandos localizados aqui são executados se  
    exp1 é verdadeira  
} else {  
    comandos localizados aqui são executados se  
    exp1 é falsa  
}
```

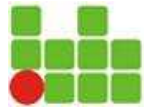


## Exemplo

...

```
if(media >= 6.0){  
    printf("Aprovado");  
} else {  
    printf("Reprovado");  
}
```

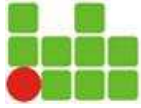
...



- Operadores relacionais

>	Maior que
>=	Maior ou igual
<	Menor que
<=	Menor igual
==	Igual
!=	Diferente





## Exemplos de expressões lógicas

`x > y`

`x < 10`

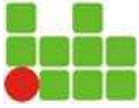
`num1 != num2`

`var1 >= var2`

`var1 <= var2`

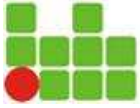
`a == b`

## Exemplo



```
x = 10;  
y = 5;
```

```
if((x+y) > 20){  
    printf("x + y eh maior que 20!\n");  
}  
else {  
    printf("x + y nao eh maior que 20!\n");  
}
```



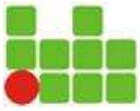
`x = 10;`

```
if(x > 5){  
    if(x < 15){
```

`...`

`}`

`}`



Para considerarmos duas ou mais expressões lógicas juntas, é preciso usar **operadores lógicos**.



## Operadores lógicos

A linguagem C possui três operadores lógicos.

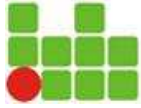
`&&` → Lógico E

`||` → Lógico OU

`!` → Lógico Não

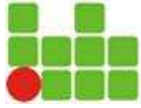


As operações lógicas E, OU e NÃO (negação) possuem resultados definidos em tabelas-verdade.



- **Operator E (AND)**

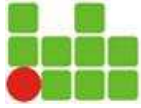
Exp1	Exp2	R
V	V	V
V	F	F
F	V	F
F	F	F



- **Operator OU (OR)**

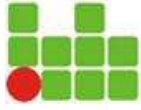
Exp1	Exp2	R
V	V	V
V	F	V
F	V	V
F	F	F





- **Operador NÃO (NOT)**

Exp1	R
V	F
F	V



Sendo E1 e E2 duas expressões de avaliação, por exemplo  $x > 3$ , as regras para as operações lógicas são as seguintes:

- E1 & E2 resulta verdadeiro (1) somente se E1 e E2 forem verdadeiras. Em qualquer outra situação resulta falso (0).
- E1 || E2 resulta falso (0) somente se as duas expressões são falsas. Caso contrário retorna verdadeiro (1).
- !E1 resulta verdadeira se E1 é falso e falso se E1 é verdadeiro.



## Combinando expressões lógicas

Exemplo: qual seria a expressão lógica para “*um número entre 5 e 10*”.

Para ser entre 5 e 10, o número precisa ser “maior que 5 ( $x > 5$ )” **E** “menor que 10 ( $x < 10$ )”.

### Exemplo de expressão

$(x > 5) \ \&\& \ (x < 10)$



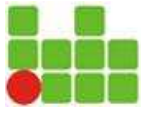
A seguir um *if* que testa se um valor está entre 5 e 10.

```
if((x > 5) && (x < 10))  
{  
    ...  
}
```



E como seria o contrário, ou seja, um *if* que testa se um valor NÃO está entre 5 e 10?

```
if((x <= 5) || (x >= 10))  
{  
    ...  
}
```



É possível também inverter o sentido de uma expressão lógica usando o operador de negação (!).

```
((x <= 5) || (x >= 10)) == !( (x > 5) && (x < 10) )
```

Ou seja, se uma expressão resultaria em V, com o ! ela resulta em F e vice-versa.

```
#include <stdio.h>
```

```
int main(){
```

```
    float nota1, nota2, media;
```

```
    printf("Entre com a primeira nota: ");
```

```
    scanf("%f", &nota1);
```

```
    printf("Entre com a segunda nota: ");
```

```
    scanf("%f", &nota2);
```

```
    media = (nota1 + nota2) / 2;
```

```
    if(media >= 6.0){
```

```
        printf("Aprovado!");
```

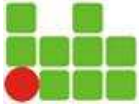
```
    } else {
```

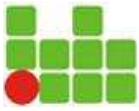
```
        printf("Reprovado!");
```

```
    }
```

```
    return 0;
```

```
}
```



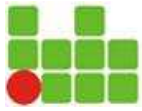


## Exemplo:

Um programa que recebe duas notas do aluno e retorna:

- “Aprovado” se a média das notas for maior ou igual a 6,0;
- “Reprovado”, se a média for menor que 4,0;
- “Recuperação” se a média for menor que 6,0, porém, maior que 4,0.





## Exercícios.