

if(aula_comecou == 1)



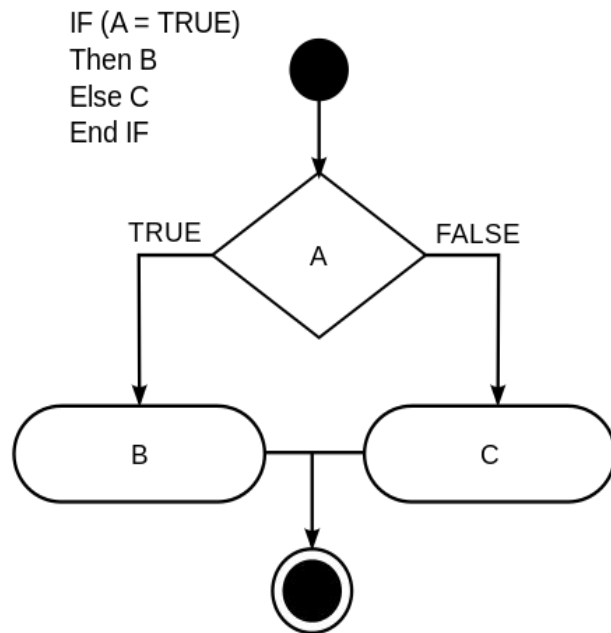
printf("Inicio da Aula\n");

Condicionais:

if-else

Condicionais

- Ou estruturas de decisão
- Vão decidir qual **parte do seu código** vai executar
 - Baseando-se em uma **expressão lógica**
- Pontos onde seu algoritmo se divide em dois possíveis caminhos
- Exemplo: Matrícula nas disciplinas
 - **Se** você estiver em condição
Procure o coordenador
 - Senão**
Utilize o Matrícula Web



se-senão (if-else)

```
...  
if ( Expressão ) {  
    // código a ser executada caso a expressão seja verdadeira  
}  
  
else {  
    // código a ser executada caso a expressão seja falsa  
}  
  
...
```

Expressão

Combinação de:

- Variáveis
- Operadores
- Valores

Resulta em Verdadeiro ou Falso

True or False

```
int x = 56;  
int y = 100;
```

```
x + 100 > y;
```

```
x > y;
```

```
y == 42;
```

se-senão (if-else)

...

```
if (x > 3) {  
    printf("x eh maior que 3\n");  
}  
else {  
    printf("x eh menor ou igual que 3\n");  
}  
...
```

x = 10

...

```
if (x > 3) {  
    printf("x eh maior que 3\n");  
}  
else {  
    printf("x eh menor ou igual que 3\n");  
}  
...
```

x = 10

...

```
if (x > 3) { // Verdadeiro x é maior que 3
    printf("x eh maior que 3\n");
}
else {
    printf("x eh menor ou igual que 3\n");
}
...
```


x = 10

...

```
if (x > 3) { // Verdadeiro x é maior que 3
    printf("x eh maior que 3\n");
}
else {
    printf("x eh menor ou igual que 3\n");
}
...
```

x = 2

...

```
if (x > 3) {  
    printf("x eh maior que 3\n");  
}  
else {  
    printf("x eh menor ou igual que 3\n");  
}  
...
```

x = 2

...

```
if (x > 3) { // Falso x é menor que 3
    printf("x eh maior que 3\n");
}
else {
    printf("x eh menor ou igual que 3\n");
}
...
```

x = 2

...

```
if (x > 3) { // Falso x é menor que 3
    printf("x eh maior que 3\n");
}
```

```
else {
    printf("x eh menor ou igual que 3\n");
}
...
```

if - else if - else

```
if (x > 10) {  
    printf("x eh maior que 10\n");  
}  
else if (x > 3) {  
    printf("x eh maior que 3\n");  
}  
else {  
    printf("x eh <= 3\n");  
}
```

if - else if - else

```
if (x > 10) {
```

Você checa se x é maior que 10

```
}
```

```
else if (x > 3) {
```

Você checa se x é maior que 3

```
}
```

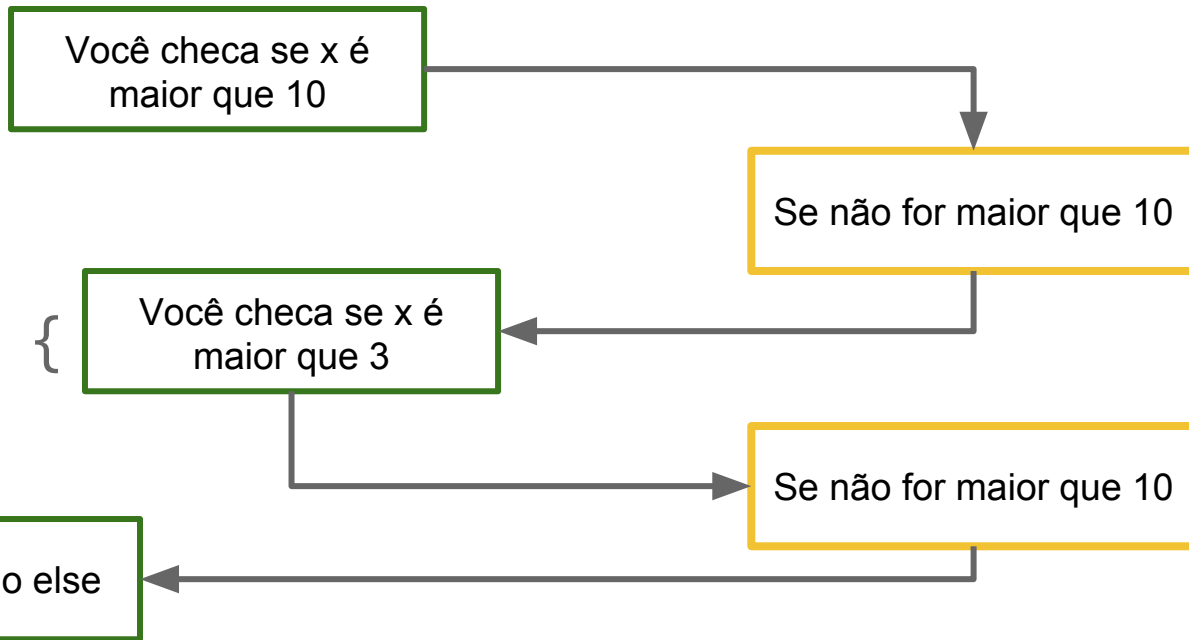
```
else {
```

Você executa o else

```
}
```

Se não for maior que 10

Se não for maior que 10



x = 11

```
if (x > 10) {  
    printf("x eh maior que 10\n");  
}  
else if (x > 3) {  
    printf("x eh maior que 3\n");  
}  
else {  
    printf("x eh <= 3\n");  
}
```

x = 11

```
if (x > 10) {  
    printf("x eh maior que 10\n");  
}  
else if (x > 3) {  
    printf("x eh maior que 3\n");  
}  
else {  
    printf("x eh <= 3\n");  
}
```


x = 9

```
if (x > 10) {  
    printf("x eh maior que 10\n");  
}  
else if (x > 3) {  
    printf("x eh maior que 3\n");  
}  
else {  
    printf("x eh <= 3\n");  
}
```

x = 9

```
if (x > 10) {  
    printf("x eh maior que 10\n");  
}  
else if (x > 3) {  
    printf("x eh maior que 3\n");  
}  
else {  
    printf("x eh <= 3\n");  
}
```

x = 2

```
if (x > 10) {  
    printf("x eh maior que 10\n");  
}  
else if (x > 3) {  
    printf("x eh maior que 3\n");  
}  
else {  
    printf("x eh <= 3\n");  
}
```

x = 2

```
if (x > 10) {  
    printf("x eh maior que 10\n");  
}  
else if (x > 3) {  
    printf("x eh maior que 3\n");  
}  
else {  
    printf("x eh <= 3\n");  
}
```

x = 10

```
if (x > 10) {  
    printf("x eh maior que 10\n");  
}  
else if (x > 3) {  
    printf("x eh maior que 3\n");  
}  
else {  
    printf("x eh <= 3\n");  
}
```

x = 10

```
if (x > 10) {  
    printf("x eh maior que 10\n");  
}  
else if (x > 3) {  
    printf("x eh maior que 3\n");  
}  
else {  
    printf("x eh <= 3\n");  
}
```

Regras Básicas

Sempre começar com o **if**

```
if (x > 3) {  
    ...  
}  
else if (x > 10) {  
    ...  
}  
else {  
    ...  
}
```



```
else if (x > 10) {  
    ...  
}  
else {  
    ...  
}
```



```
else {  
    ...  
}
```



O **else** sempre é o último

```
if (x > 3) {  
    ...  
}  
else if (x > 10) {  
    ...  
}  
else {  
    ...  
}
```



```
else {  
    ...  
}  
if (x > 10) {  
    ...  
}
```



```
else {  
    ...  
}  
else if (x > 10) {  
    ...  
}
```



O **else** não é obrigatório

```
if (x > 3) {  
    ...  
}  
else if (x > 10) {  
    ...  
}
```



```
if (x > 10) {  
    ...  
}
```



```
if (x > 3) {  
    ...  
}  
else if (x > 10) {  
    ...  
}  
else if (x > 20) {  
    ...  
}
```



Operadores

Operadores Aritiméticos

+ - * / % ++ --

Operadores Relacionais

== Igual

!= Diferente

< Menor que

> Maior que

<= Menor ou igual

>= Maior ou igual

```
int x = 4;  
int y = 5;  
int z = 4;
```

x == z

x != z

x < y

x > y

x >= z

x > z

x < z

y <= x

y >= x

```
int x = 4;  
int y = 5;  
int z = 4;
```

x == z Verdadeiro

x != z Falso

x < y Verdadeiro

x > y Falso

x >= z Verdadeiro

x > z Falso

x < z Falso

y <= x Falso

y >= x Verdadeiro

Operadores Lógicos

&& and / e

|| or / ou

! not / negação

AND

Verdadeiro && Verdadeiro = Verdadeiro

Verdadeiro && Falso = Falso

Falso && Verdadeiro = Falso

Falso && Falso = Falso

AND - Passei em APC?

mencao >= MM && presenca >= 75%

MS && 83% = Passei!

SS && 23% = Reprovei =/

MI && 100% = Reprovei =/

II && 2% = Reprovei =/

OR

Verdadeiro || Verdadeiro = Verdadeiro

Verdadeiro || Falso = Verdadeiro

Falso || Verdadeiro = Verdadeiro

Falso || Falso = Falso

OR - Entrei na UnB?

Passei no PAS || Passei no Vestibular

Passei no PAS || Passei no Vestibular = Entrei o/

Passei no PAS || Bombei no Vestibular = Entrei o/

Bombei no PAS || Passei no Vestibular = Entrei o/

Bombei no PAS || Bombei no Vestibular = Bombei =/

NOT

`!Verdadeiro = Falso`

`!Falso = Verdadeiro`

Mistura de Operadores


```
int x = 4;  
int y = 5;  
int z = 4;
```

$x + y * 3 - z;$

$x + (y * 3) - z;$

```
int x = 4;  
int y = 5;  
int z = 4;
```

Ordem dos operadores:

1. () parenteses
2. ++ --
3. * / %
4. + -

x + y * 3 - z; // 15

x + (y * 3) - z; // 15

```
int x = 4;  
int y = 5;  
int z = 4;
```

```
x + y * 3 - z;    // 15  
x + (y * 3) - z; // 15
```

Ordem dos operadores:

1. () parenteses
2. ++ --
3. * / %
4. + -
5. <= >= < >
6. == !=

$x + y \geq y * z;$

```
int x = 4;  
int y = 5;  
int z = 4;
```

```
x + y * 3 - z;    // 15
```

```
x + (y * 3) - z; // 15
```

Ordem dos operadores:

1. () parenteses
2. ++ --
3. * / %
4. + -
5. <= >= < >
6. == !=

$$\underbrace{x + y}_{9} \geq \underbrace{y * z}_{20};$$

```
int x = 4;  
int y = 5;  
int z = 4;
```

```
x + y * 3 - z;    // 15
```

```
x + (y * 3) - z; // 15
```

Ordem dos operadores:

1. () parenteses
2. ++ --
3. * / %
4. + -
5. <= >= < >
6. == !=

$\underbrace{x + y}_{9} \geq \underbrace{y * z}_{20};$

9 \geq 20

```
int x = 4;  
int y = 5;  
int z = 4;
```

```
x + y * 3 - z;    // 15  
x + (y * 3) - z; // 15
```

Ordem dos operadores:

1. () parenteses
2. ++ --
3. * / %
4. + -
5. <= >= < >
6. == !=

$\underbrace{x + y}_{9} \geq \underbrace{y * z}_{20} // \text{False}$

9 ≥ 20

```
int x = 4;  
int y = 5;  
int z = 4;
```

```
x + y * 3 - z;    // 15  
x + (y * 3) - z; // 15
```

Ordem dos operadores:

1. () parenteses
2. ++ --
3. * / %
4. + -
5. <= >= < >
6. == !=

$\underbrace{x + y}_{9} \geq \underbrace{y * z}_{20}; \quad // \text{ False}$

$9 \geq 20$

$!(x + y \geq y * z);$

```
int x = 4;  
int y = 5;  
int z = 4;
```

```
x + y * 3 - z;    // 15  
x + (y * 3) - z; // 15
```

Ordem dos operadores:

1. () parenteses
2. ++ -- !
3. * / %
4. + -
5. <= >= < >
6. == !=

$\underbrace{x + y}_{9} \geq \underbrace{y * z}_{20}; // \text{ Falso}$

$9 \geq 20$

$!(x + y \geq y * z); // \text{ Verdadeiro}$


```
int x = 4;  
int y = 5;  
int z = 4;
```

```
x + y * 3 - z;    // 15  
x + (y * 3) - z; // 15
```

```
x + y >= y * z; // Falso  
!(x + y >= y * z); // Verdadeiro
```

Ordem dos operadores:

1. () parenteses
2. ++ -- !
3. * / %
4. + -
5. <= >= < >
6. == !=
7. &&
8. ||

```
(x * x != z * z) && (y > 6);
```

```
int x = 4;  
int y = 5;  
int z = 4;
```

```
x + y * 3 - z;    // 15  
x + (y * 3) - z; // 15
```

```
x + y >= y * z; // Falso  
!(x + y >= y * z); // Verdadeiro
```

Ordem dos operadores:

1. () parenteses
2. ++ -- !
3. * / %
4. + -
5. <= >= < >
6. == !=
7. &&
8. ||

$$\underbrace{(x * x)}_{16} \neq \underbrace{(z * z)}_{16} \ \&\& \ (y > 6);$$

```
int x = 4;  
int y = 5;  
int z = 4;
```

```
x + y * 3 - z;    // 15  
x + (y * 3) - z; // 15
```

```
x + y >= y * z; // Falso  
!(x + y >= y * z); // Verdadeiro
```

Ordem dos operadores:

1. () parenteses
2. ++ -- !
3. * / %
4. + -
5. <= >= < >
6. == !=
7. &&
8. ||

```
(x * x != z * z) && (y > 6);  
16    !=    16    // Falso
```

```
int x = 4;  
int y = 5;  
int z = 4;
```

```
x + y * 3 - z;    // 15  
x + (y * 3) - z; // 15
```

```
x + y >= y * z; // Falso  
!(x + y >= y * z); // Verdadeiro
```

Ordem dos operadores:

1. () parenteses
2. ++ -- !
3. * / %
4. + -
5. <= >= < >
6. == !=
7. &&
8. ||

```
(x * x != z * z) && (y > 6);
```

```
// Falso
```

```
int x = 4;  
int y = 5;  
int z = 4;
```

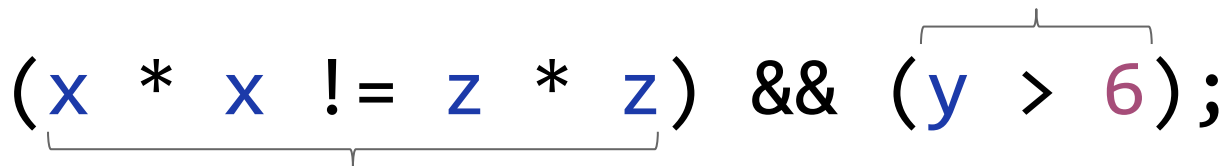
```
x + y * 3 - z;    // 15  
x + (y * 3) - z; // 15
```

```
x + y >= y * z; // Falso  
!(x + y >= y * z); // Verdadeiro
```

Ordem dos operadores:

1. () parenteses
2. ++ -- !
3. * / %
4. + -
5. <= >= < >
6. == !=
7. &&
8. ||

// Falso


(x * x != z * z) && (y > 6);

// Falso

```
int x = 4;  
int y = 5;  
int z = 4;
```

```
x + y * 3 - z;    // 15  
x + (y * 3) - z; // 15
```

```
x + y >= y * z; // Falso  
!(x + y >= y * z); // Verdadeiro
```

Ordem dos operadores:

1. () parenteses
2. ++ -- !
3. * / %
4. + -
5. <= >= < >
6. == !=
7. &&
8. ||

// Falso

$(x * x \neq z * z) \ \&\& \ (y > 6);$ // Falso

// Falso

```
int x = 4;  
int y = 5;  
int z = 4;
```

```
x + y * 3 - z;    // 15  
x + (y * 3) - z; // 15
```

```
x + y >= y * z; // Falso  
!(x + y >= y * z); // Verdadeiro
```

```
(x * x != z * z) && (y > 6); // Falso
```

Ordem dos operadores:

1. () parenteses
2. ++ -- !
3. * / %
4. + -
5. <= >= < >
6. == !=
7. &&
8. ||

Problema

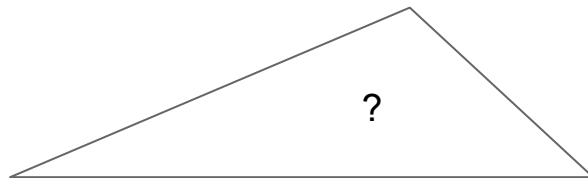
Problema do Triângulo

Dado o tamanho de 3 lados, diga se eles formam um triângulo válido ou não.

a _____

b _____

c _____



Problema do Triângulo

A entrada do programa serão 3 números inteiros que representam o tamanho dos lados do triângulo, em um intervalo de 1 a 1000.

7 4 10

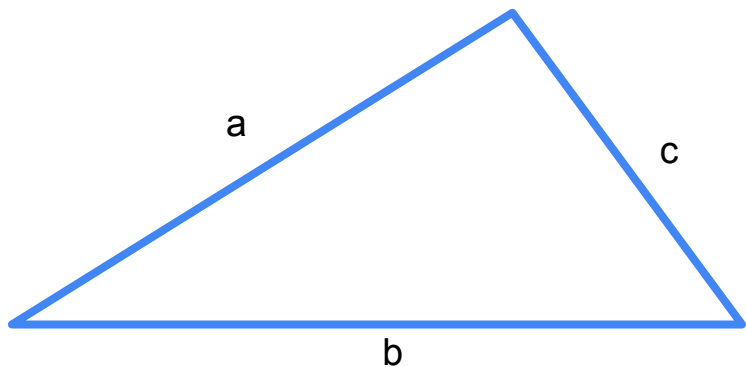
5 3 8

A saída do programa deve ser “Yes” se o triângulo for válido, e “No” se o triângulo for inválido.

Yes

No

Regra do Triângulo

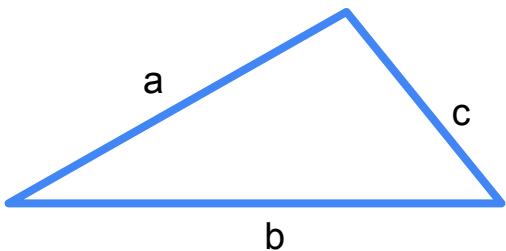


$$a + b > c$$

$$a + c > b$$

$$b + c > a$$

Tentem Resolver



$$a + b > c$$

$$a + c > b$$

$$b + c > a$$

Entrada

7 4 10

5 3 8

Saída

Yes

No

Resolução

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

```
int main() {
```

```
    int a = 0;
```

```
    int b = 0;
```

```
    int c = 0;
```

```
    scanf("%d %d %d", &a, &b, &c);
```

```
    if ((a + b > c) && (a + c > b) && (b + c > a)) {  
        printf("Yes\n");
```

```
    }
```

```
    else {  
        printf("No\n");
```

```
    }
```

```
    return 0;
```

```
}
```

**E uma
calculadora?
Como faz?**

else



```
printf("Fim da Aula\n");
```