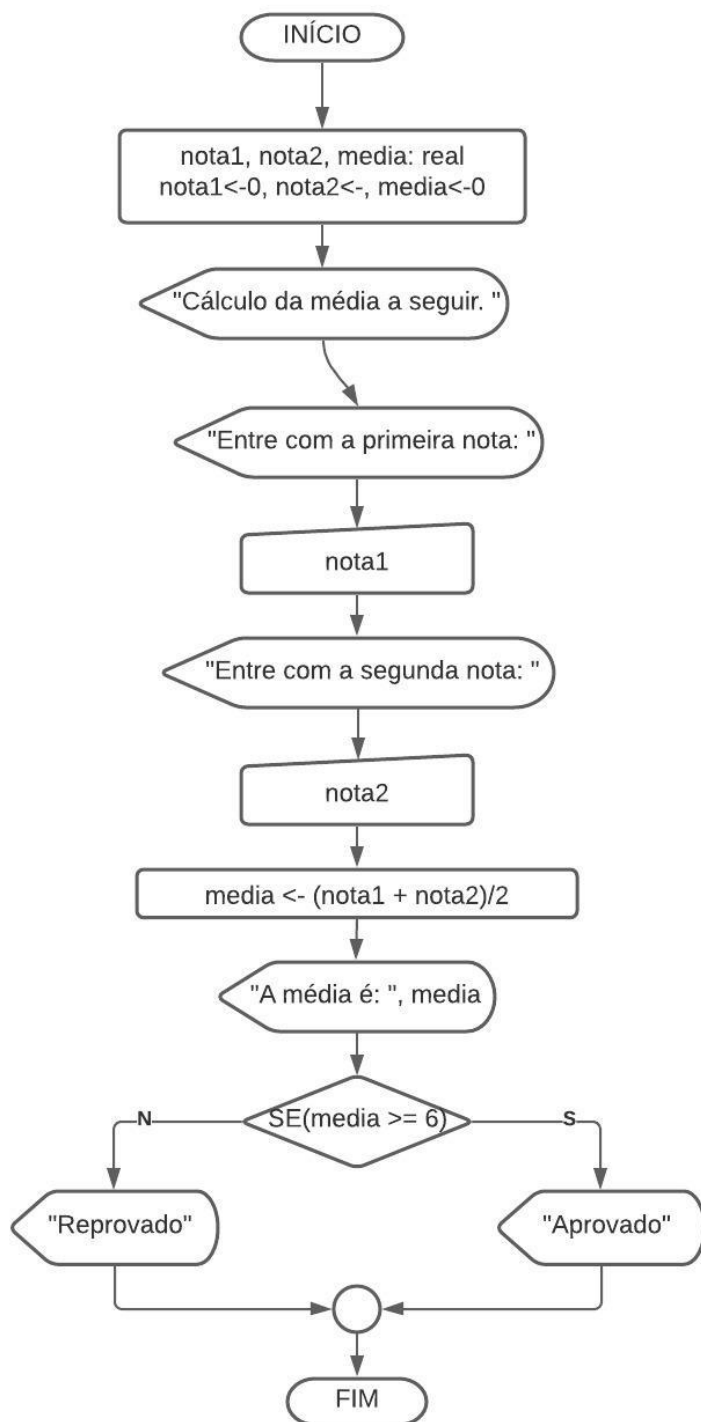


Exercícios envolvendo estruturas de decisão

Desenvolver o diagrama de blocos, teste de mesa , português estruturado e codificação em C++ para os seguintes enunciados:

1) Solicitar duas notas para o cálculo da média , mostrar a média e se a média for Maior ou igual a 6, mostrar "Aprovado", caso contrário mostrar "Reprovado".



TESTE DE MESA

n1	n2	media	mostra
4	5,5	4,75	Reprovado
6	6,5	6,25	Aprovado

Pseudocódigo

Variáveis

n1,n2,media:real

Início

n1←0,n2←0,media←0

escreva "Calcula media "

escreva "Digite primeira nota: "

leia n1

escreva "Digite segunda nota: "

leia n2

media←(n1+n2)/2

escreva "A media eh: ", media

se (media >= 6) então

escreva "Aprovado"

senão

escreva "Reprovado"

fim-se

Fim

Codificação C++

```
using namespace std;
```

```
int main()
```

```
{
```

```
    float n1=0,n2=0,media=0;
```

```
        cout <<"Calcula media";
```

```
    cout<<"Digite primeira nota: ";
```

```
    cin>> n1;
```

```
    cout<<"digite segunda nota: ";
```

```
    cin>>n2;
```

```
    media=(n1+n2)/2;
```

```
    cout<<"A media eh: "<<media;
```

```
    if (media>=6)
```

```
    {
```

```
        cout<<"Aprovado";
```

```
    }
```

	<pre>else { cout<<"Reprovado"; } Return 0; // não precisa colocar no seu código, pois o dev já inclui automaticamente }</pre>
--	--

2) Solicitar 3 valores inteiros para o cálculo da equação de segundo grau,

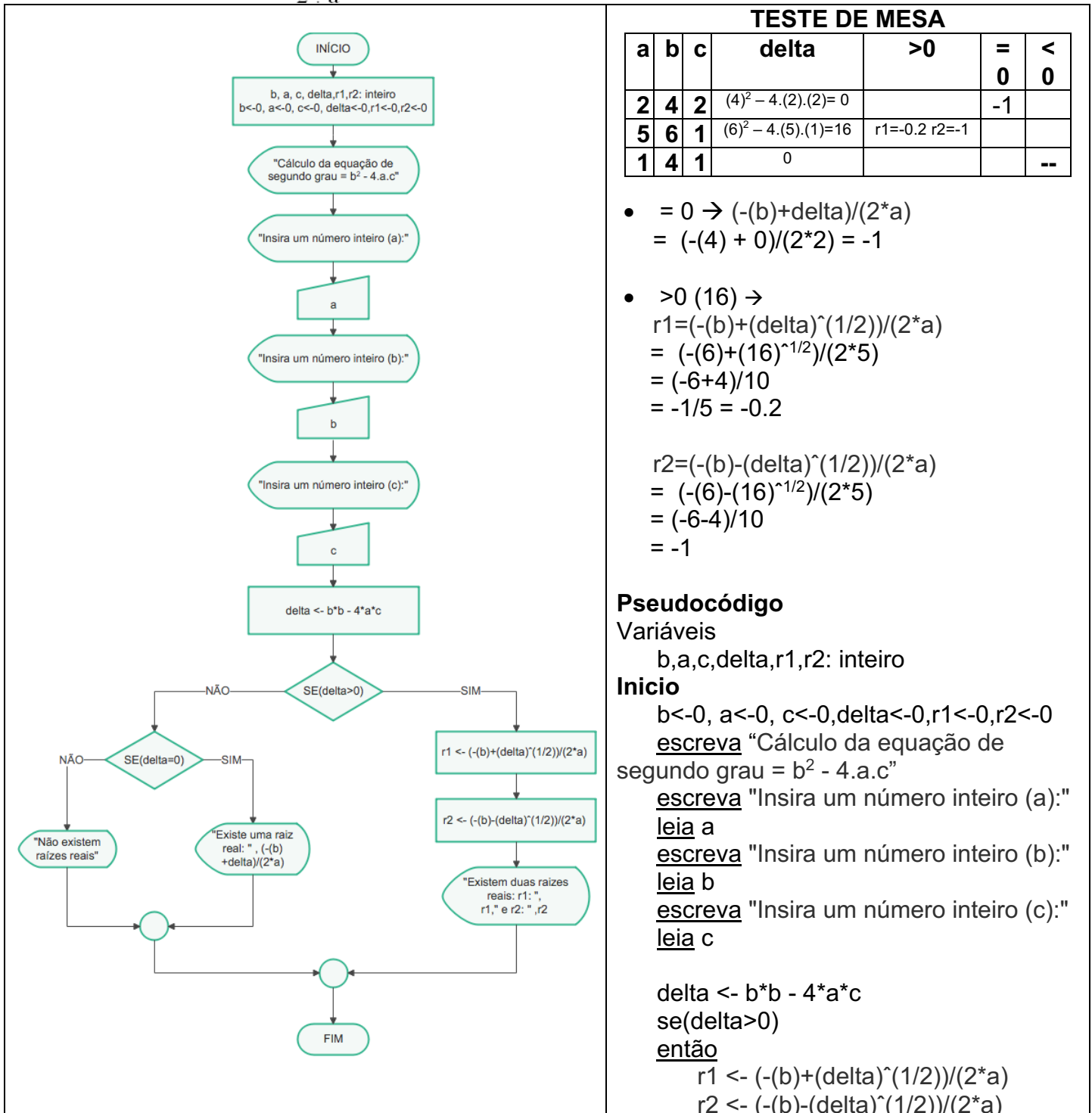
$$\text{delta} = b^2 - 4 \cdot a \cdot c$$

Se $\text{delta} < 0 \rightarrow$ não existem raízes reais

Se $\text{delta} = 0 \rightarrow$ existe uma raiz real

Se $\text{delta} > 0 \rightarrow$ existem duas raízes reais

$$\text{Raízes} \rightarrow \frac{-b \pm \sqrt{\Delta}}{2 \cdot a}$$



```

        escreva "Existem duas raizes reais:
                r1: ", r1," e r2: ",r2
    senão
        se(delta=0)
            então
                escreva("Existe uma raiz real: " ,
(-b)+delta)/(2*a))
            senão
                escreva("Não existem raízes
reais")
        fim-se
    fim-se
Fim

```

Codificação C++

```

#include <iostream>
#include <math.h>

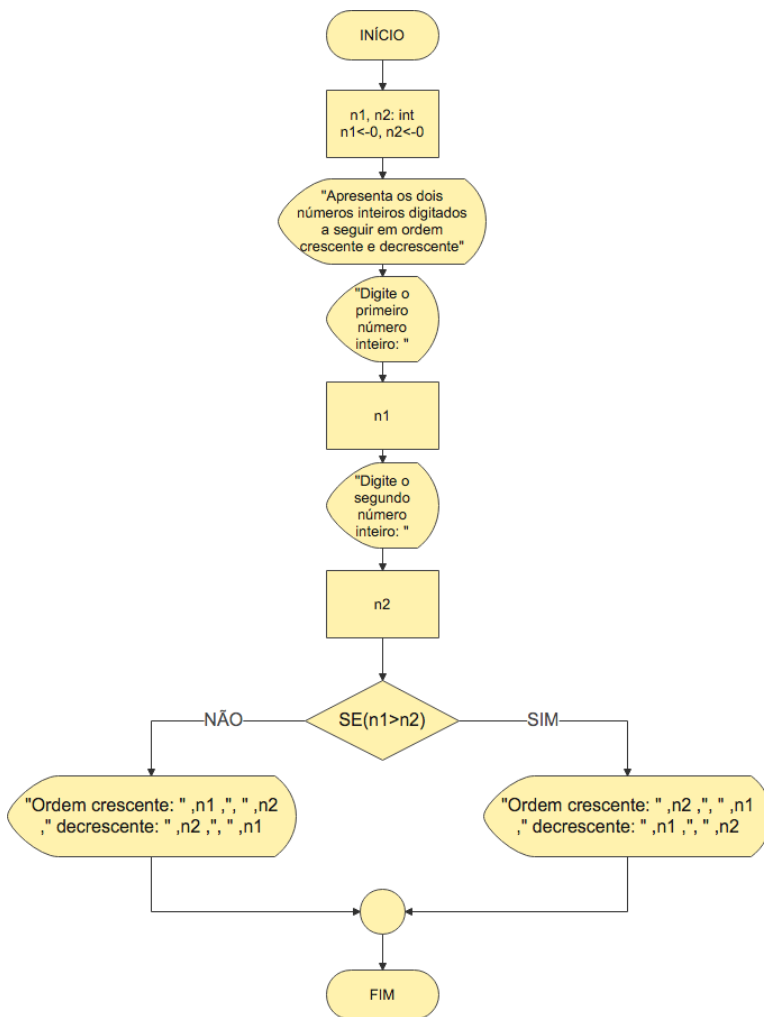
int main() {
    float b=0, a=0, c=0, r1=0, r2=0,
delta=0;
    std::cout << "Cálculo da equação
de segundo grau = b^(2) - 4.a.c\n";
    std::cout << "Insira um número
inteiro (a):";
    std::cin>> a;
    std::cout << "Insira um número
inteiro (b):";
    std::cin>> b;
    std::cout << "Insira um número
inteiro (c):";
    std::cin>> c;

    delta = b*b - 4*a*c;
    if(delta>0) {
        r1=(-b+sqrt(delta))/(2*a);
        r2=(-b-sqrt(delta))/(2*a);
        std::cout <<"Existem duas raizes
reais: r1: " <<r1 <<" e r2: " <<r2;
    } else {
        if(delta==0){
            std::cout <<"Existe uma raiz
real: " <<(-b+ delta)/(2*a);
        }
        else {

```

	<pre>std::cout <<"Não existem raízes reais"; } } }</pre>
--	--

3) Solicitar dois valores inteiros, e mostrá-los na ordem crescente e decrescente.



TESTE DE MESA

n1	n2	n1>n2	show
2	4	NÃO	C: 2,4 D: 4,2
5	1	SIM	C: 1,5 D: 5,1

Pseudocódigo

Variáveis

n1, n2: inteiro

Início

n1<-0, n2<-0

escreva "Apresenta os dois números inteiros digitados a seguir em ordem crescente e decrescente"

escreva "Digite o primeiro número inteiro: "

leia n1

escreva "Digite o segundo número inteiro: "

leia n2

se(n1>n2)

então

escreva "Ordem crescente: ", n2, ", ", n1, " decrescente: ", n1, ", ", n2

senão

escreva "Ordem crescente: ", n1, ", ", n2, " decrescente: ", n2, ", ", n1

fim-se

Fim

Codificação C++

```
#include <iostream>
```

```
int main() {
    int n1=0, n2=0;
```

```
    std::cout << "Apresenta os dois  
números inteiros digitados a seguir  
em ordem crescente e decrescente\n";
```

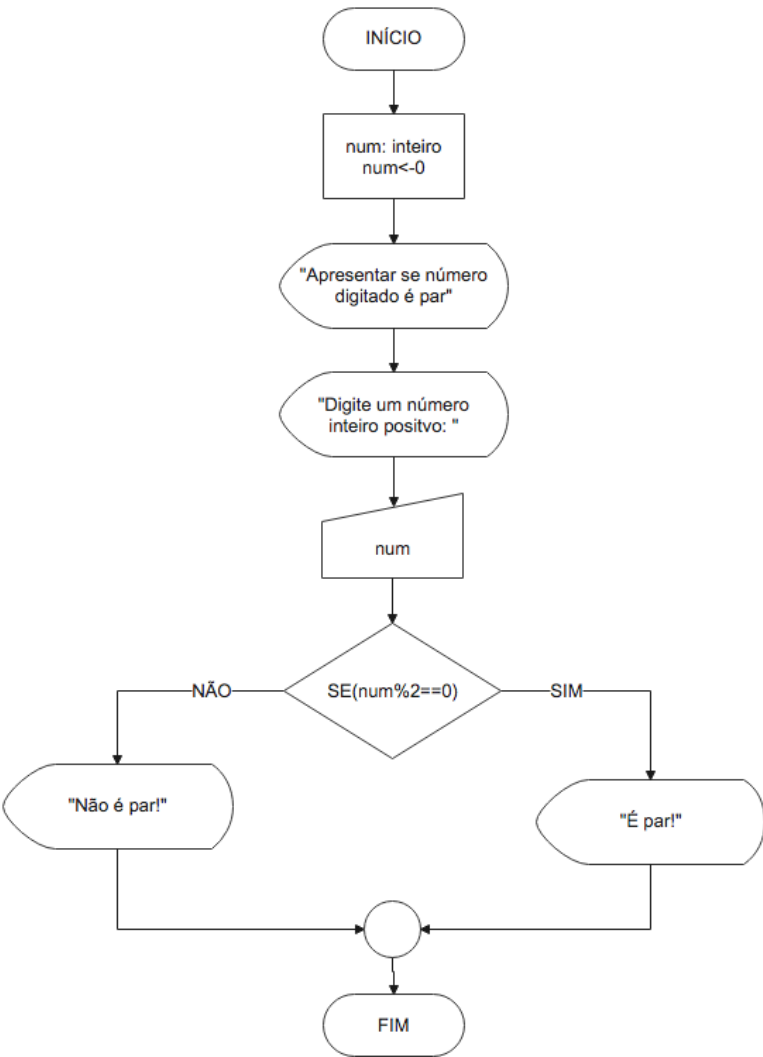
```
    std::cout << "Digite o primeiro  
número inteiro: \n";
    std::cin >> n1;
```

```
    std::cout << "Digite o segundo  
número inteiro: \n";
```

```
std::cin >> n2;

if(n1>n2){
    std::cout <<"Ordem crescente: "
<<n2 <<" , " <<n1 <<" decrescente: "
<<n1 <<" , " <<n2;
} else {
    std::cout <<"Ordem crescente: "
<<n1 <<" , " <<n2 <<" decrescente: "
<<n2 <<" , " <<n1;
}
}
```

4) Solicitar um valor inteiro positivo (consistir) e mostrar se esse número é par ou não par



TESTE DE MESA

num	num%2	condição	msg
3	3/2 = r= 1	1 ≠ 0	não é par
4	4/4= r= 0	0 = 0	é par

Pseudocódigo

Variáveis
num: inteiro

Início
num<-0
escreva "Apresentar se número digitado é par"
escreva "Digite um número inteiro positivo: "
leia num
se(num%2==0)
então
escreva "É par!"
senão
escreva "Não é par!"
fim-se

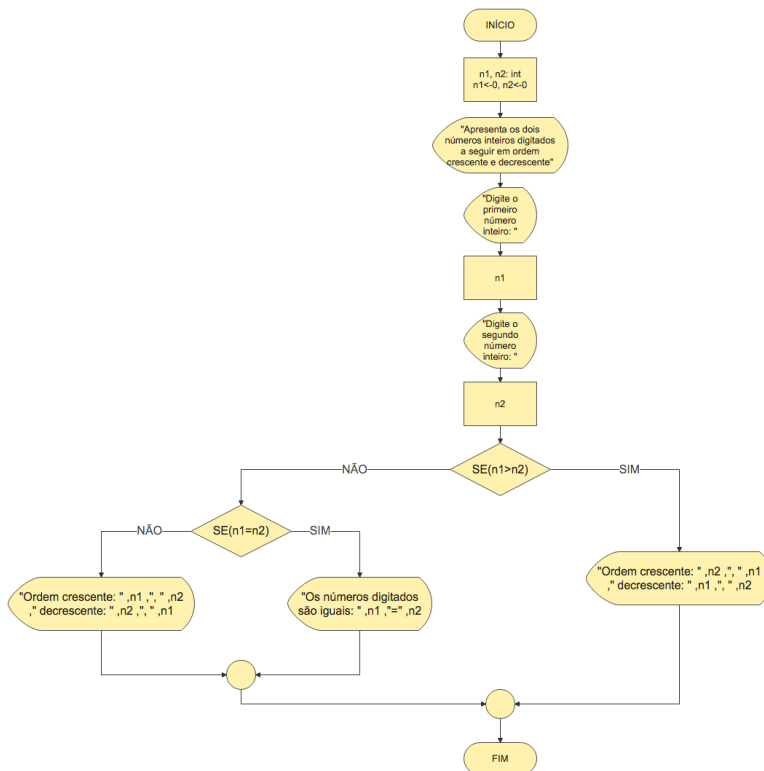
Fim

Codificação C++

```
#include <iostream>

int main() {
    int num=0;
    std::cout << "Apresentar se número digitado é par\n";
    std::cout << "Digite um número inteiro positivo: ";
    std::cin >> num;
    if(num%2==0) {
        std::cout << "É par!";
    } else {
        std::cout << "Não é par!";
    }
}
```


5) Solicitar dois valores inteiros, e mostrá-los na ordem crescente e decrescente.(prever para quando os valores forem iguais.



TESTE DE MESA

n1	n2	n1>n2	n1=n2	show
2	4	NÃO	NÃO	C: 2,4 D: 4,2
5	1	SIM	-	C: 1,5 D: 5,1
2	2	NÃO	SIM	2 = 2

Pseudocódigo

Variáveis

n1, n2: inteiro

Início

n1<-0, n2<-0

escreva "Apresenta os dois números inteiros digitados a seguir em ordem crescente e decrescente"

escreva "Digite o primeiro número inteiro: "

leia n1

escreva "Digite o segundo número inteiro: "

leia n2

se(n1>n2)

então

escreva "Ordem crescente: " ,n2 , " ,n1 , " decrescente: " ,n1 , " , " ,n2

senão

se(n1=n2)

então

escreva "Os números digitados são iguais: " ,n1 , "=" ,n2

senão

escreva "Ordem crescente: " ,n1 , " , " ,n2 , " decrescente: " ,n2 , " , " ,n1

fim-se

fim-se

Fim

Codificação C++

```
#include <iostream>
```

```
int main() {
    int n1=0, n2=0;
```

```
std::cout << "Apresenta os dois  
números inteiros digitados a seguir  
em ordem crescente e decrescente\n";
```

```
std::cout << "Digite o primeiro  
número inteiro: \n";  
std::cin >> n1;
```

```
std::cout << "Digite o segundo  
número inteiro: \n";  
std::cin >> n2;
```

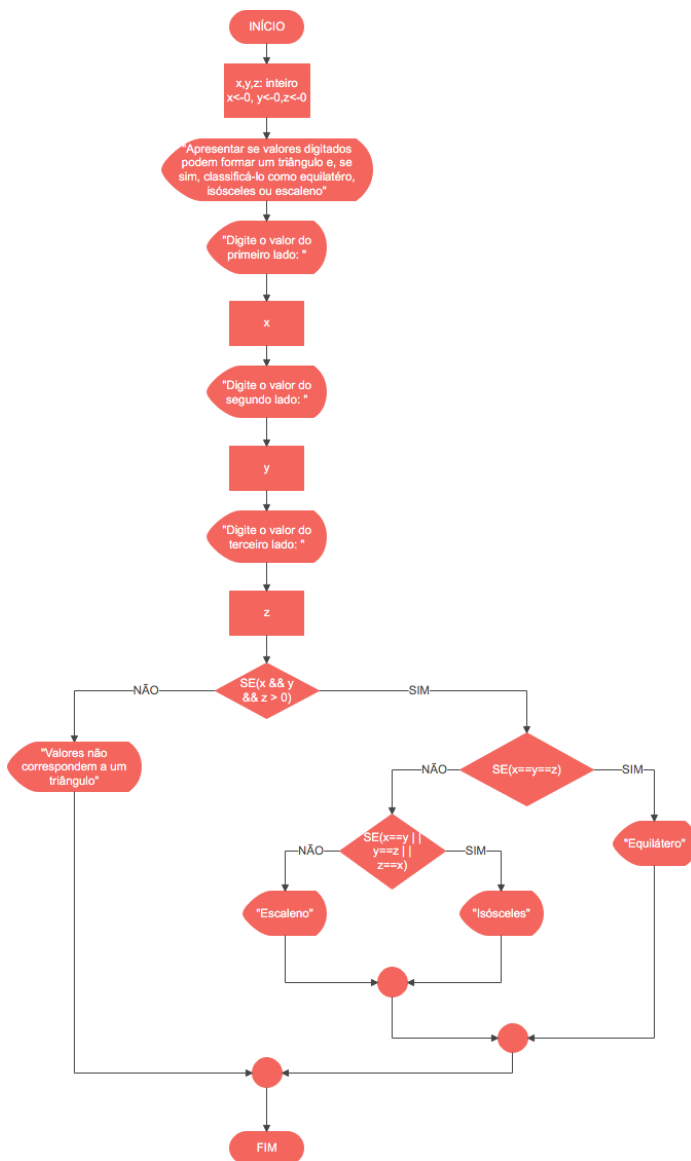
```
if(n1>n2){  
    std::cout <<"Ordem crescente: "  
<<n2 <<" , " <<n1 <<" decrescente: "  
<<n1 <<" , " <<n2;  
} else {  
    if (n1==n2) {  
        std::cout <<"Os números  
digitados são iguais: " <<n1 <<"=" <<n2;  
    } else {  
        std::cout <<"Ordem crescente:  
" <<n1 <<" , " <<n2 <<" decrescente: "  
<<n2 <<" , " <<n1;  
    }  
}  
}
```

6) Receber 3 valores numéricos, X, Y e Z, e verificar se esses valores podem corresponder aos lados de um triângulo. Em caso afirmativo, informar ao usuário se o triângulo é equilátero, isósceles ou escaleno.

ENTRADA \leftarrow x, y e z

PROCESSO \leftarrow S/P = se x,y e z >0 então válido

SAÍDA \leftarrow se triângulo é eq., iso ou esc.



TESTE DE MESA

x	y	z	x, y e z > 0	x= (y e z) equilátero	x=y ou y=z ou z=x isósceles	senão escaleno
1	1	1	(V)	(V)	-	-
3	2	2	(V)	(F)	(V)	-
1	2	3	(V)	(F)	(F)	(V)

Pseudocódigo

Variáveis

x,y,z: inteiro

Início

x<-0, y<-0, z<-0

escreva "Apresentar se valores digitados podem formar um triângulo e, se sim, classificá-lo como equilátero, isósceles ou escaleno"

escreva "Digite o valor do primeiro lado: "

leia x

escreva "Digite o valor do segundo lado: "

leia y

escreva "Digite o valor do terceiro lado: "

leia z

se(x&&y&&z>0)

então

se(x==y && x==z)

então

escreva "equilátero"

senão

se(x==y | y==z || z=x)

então

escreva "isósceles"

senão

escreva "Escaleno"

fim-se

fim-se

senão

escreva "Valores não correspondem a um triângulo"

fim-se

Fim

Codificação C++

```
#include <iostream>
```

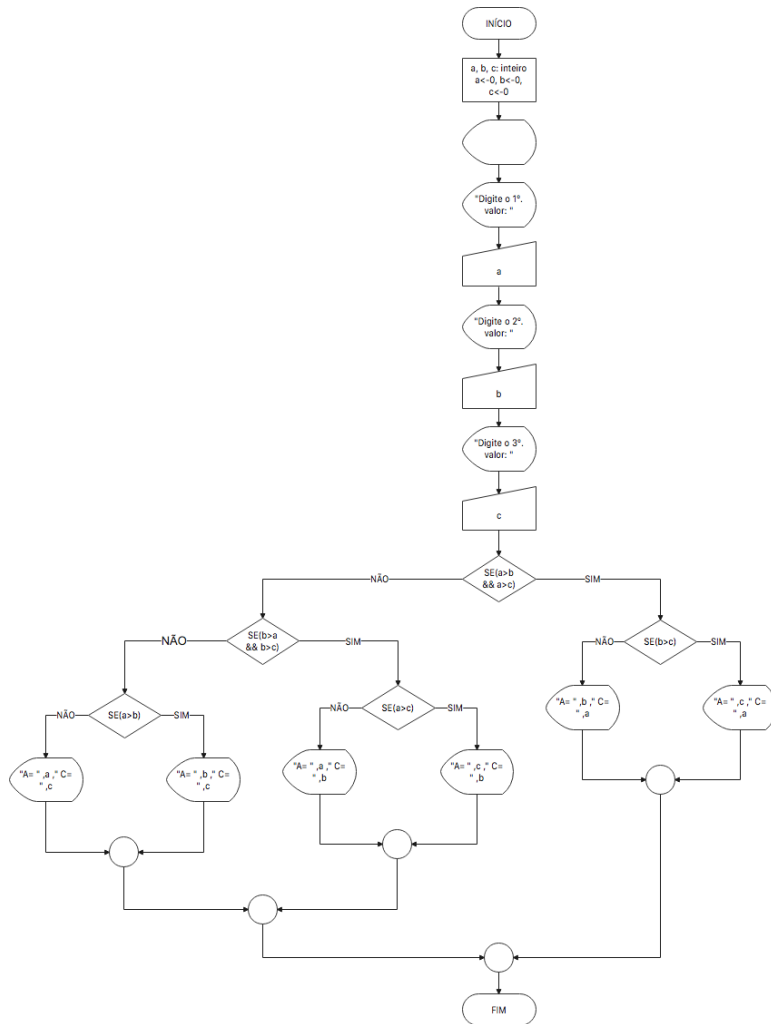
```
int main() {  
    int x=0, y=0, z=0;  
    std::cout << "Apresentar se valores  
digitados podem formar um triângulo e, se  
sim, classificá-lo como equilátero,  
isósceles ou escaleno\n";  
    std::cout << "Digite o valor do primeiro  
lado: \n";  
    std::cin >> x;  
    std::cout << "Digite o valor do segundo  
lado: \n";  
    std::cin >> y;    std::cout << "Digite o  
valor do terceiro lado: \n";  
    std::cin >> z;  
  
    if(x&&y&&z>0) { //verifica se >0  
        if((x==y) && (x==z) ){ // eq  
            std::cout <<"equilátero";  
        }  
        else{  
            if((x==y)|| (x==z)|| (y==z)){ // iso  
                std::cout <<"isósceles";  
            }  
            else{  
                std::cout <<"escaleno";  
            } //fim iso  
        }  
    }  
    else {  
        std::cout <<"Valores não correspondem  
a um triângulo";  
    } // final verif. se >-  
  
}
```

7) Receber 3 valores numéricos em 3 variáveis, A, B e C, e trocar os valores entre as variáveis de forma que, ao final do algoritmo, a variável A possua o menor valor e a variável C o maior.

Entrada → A,B e C

Processamento → $A \leftarrow$ menor valor e $C \leftarrow$ maior valor

Saída →



TESTE DE MESA

Pseudocódigo

Variáveis

a,b,c: inteiro

Início

a<-0, b<-0, c<-0

escreva ""

escreva "Digite o 1º. valor: "

leia a

escreva "Digite o 2º. valor: "

leia b

escreva "Digite o 3º. valor: "

leia c

se(a>b&& a>c)

então

se(b>c)

então

escreva "A= ",c,"C= ",a

senão

escreva "A= ",b,"C= ",a

fim-se

senão

se(b>a&&b>c)

então

se(a>c)

então

escreva "A= ",c,"C= ",b

senão

escreva "A= ",a,"C= ",b

fim-se

fim-se

senão

se(a>b)

então

escreva "A= ",b,"C= ",c

senão

escreva "A= ",a,"C= ",c

fim-se

fim-se

Fim

Codificação C++

```
#include <iostream>
```

```
int main() {
    int a=0, b=0, c=0;
    std::cout <<"\n";
    std::cout <<"\nDigite o primeiro
valor: ";
    std::cin >> a;
    std::cout <<"Digite o segundo
valor: ";
    std::cin >> b;
    std::cout <<"Digite o terceiro
valor: ";
    std::cin >> c;

    if( a>b&& a>c) {
        if(b>c) {
            std::cout <<"A= " <<c <<" C=
" <<a;
        } else {
            std::cout <<"A= " <<b <<" C=
" <<a;
        }
    } else {
        if(b>a && b>c) {
            if(a>c) {
                std::cout <<"A= " <<c <<"
C= " <<b;
            } else {
                std::cout <<"A= " <<a <<"
C= " <<b;
            }
        } else {
            if(a>b){
                std::cout <<"A= " <<b <<"
C= " <<c;
            } else{
                std::cout <<"A= " <<a <<"
C= " <<c;
            }
        }
    }
}
```

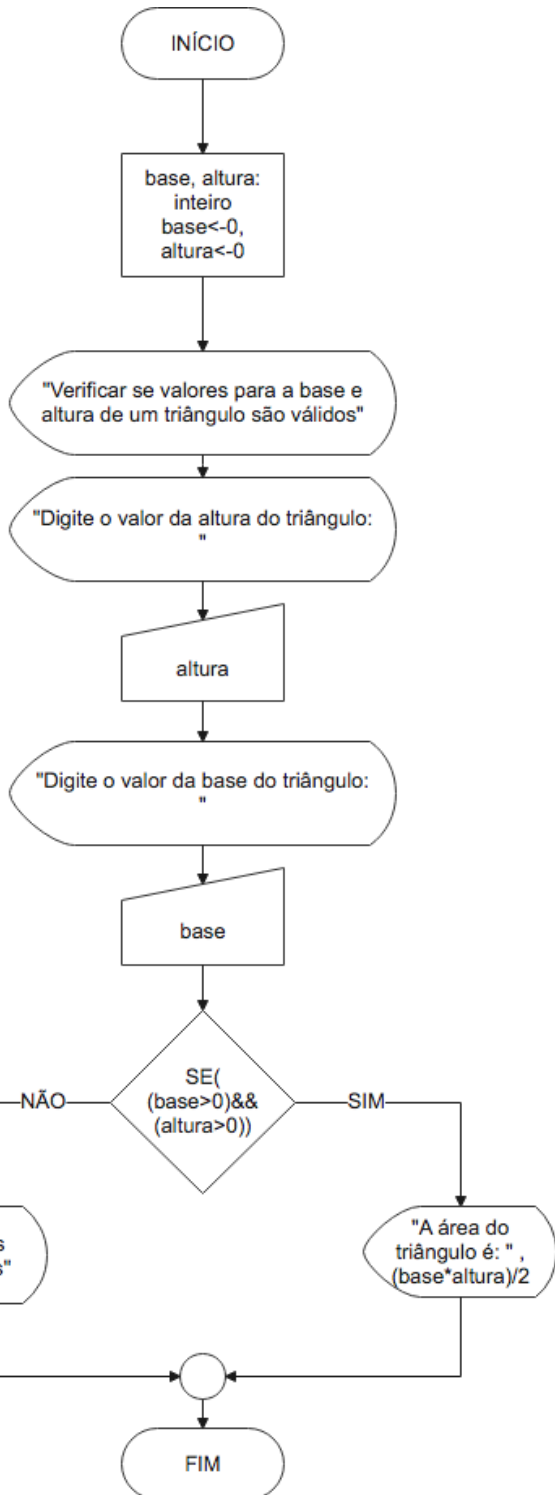
	<div> <div>}</div> <div>}</div> <div>}</div> </div>
--	---

8) Receber valores de base e altura de um triângulo e verificar se são valores válidos (positivos maiores que zero). Em caso afirmativo, calcular a área do triângulo.

Entrada ← Base e altura de um triângulo (>0)

Processamento ← área do triângulo

Saída ← verificar se valores são válidos



TESTE DE MESA

Pseudocódigo

Variáveis

base, altura: inteiro

Início

base <- 0, altura <- 0

escreva "Verificar se valores para a base e altura de um triângulo são válidos"

escreva "Digite o valor da altura do triângulo: "

leia altura

escreva "Digite o valor da base do triângulo: "

leia base

se((base>0)&&(altura>0))

então

escreva "A área do triângulo é: ", (base*altura)/2

senão

escreva "Valores inválidos"

fim-se

Fim

Codificação C++

```
#include <iostream>
```

```
int main() {
```

```
    int base=0, altura=0;
```

```
    std::cout << "Verificar se valores  
para a base e altura de um triângulo  
são válidos\n";
```

```
    std::cout <<"Digite o valor da  
altura do triângulo: ";
```



```
std::cin >>altura;

std::cout <<"Digite o valor da
base do triângulo: ";
std::cin >>base;

if((base>0)&&(altura>0)){
    std::cout << " A área do
triângulo é: " <<(base*altura)/2;
} else {
    std::cout << "Valores
inválidos";
}
}
```

9) Dado um número inteiro de segundos, mostrar a quantas horas, minutos e segundos ele corresponde.

Entrada ← número inteiro de segundos

Processamento ← conversão segundos para horas, minutos e segundos

Saída ← segundos convertidos

TESTE DE MESA

Pseudocódigo

Variáveis

num: inteiro

Início

num←-0

Fim

Codificação C++

```
#include <iostream>

int main() {
    int sec=0;
    std::cout << "Digite o valor em
segundos: ";
    std::cin >> sec;

    //if(sec>=3600) {
        if( ((sec%3600)>0) ||
            (((sec%3600/60)%60)>0)){

            if(sec%60>0){
                std::cout
                <<( (sec-(sec%3600))/3600 )
                <<": "
                <<((sec%3600)/60) <<": "
                <<(sec%60);
            } else{
                std::cout
                <<( (sec-(sec%3600))/3600 )
                <<": "
                <<((sec%3600)/60)
                <<":00";
            }

        } else {
```

```
std::cout << sec/3600
<<":00:00" ;
}
//} else {}

/* 1:30:15    5415 segundos
   3:42:00    13320 segundos
   3:00:00    10800 segundos */
}
```

10) Credito especial em função do saldo médio.

- ler o saldo medio do cliente,
- calcular o valor do credito em função da tabela a seguir:

SALDO MÉDIO	% CREDITO
0 a 200	0%
201 a 400	20%
401 a 600	30%
Acima de 601	40%

- mostrar como saída uma mensagem informando o saldo médio e o valor do crédito

Entrada ← saldo médio do cliente

Processamento ← $\text{credito} = \text{saldo} * [1 + (\text{porcentagem}/100)]$

Saída ← exibir crédito disponível de acordo com a % da tabela

TESTE DE MESA

Pseudocódigo

Variáveis

saldo: inteiro

Inicio

num<-0

escreva "Informar valor do crédito para o saldo médio disponível"

escreva "Digite seu saldo: "

leia saldo

se(saldo>0 && saldo<=600)

então

se(saldo<=200)

então

escreva "Seu crédito é de R\$",
saldo * [1 + (0/100)]

senão

escreva "Não é par!"

fim-se

Fim

Codificação C++

```
#include <iostream>
```

```
int main() {
```

```
    float saldo=0, porc=0;
```

```
std::cout << "\nValor do crédito  
disponível com base no saldo médio";  
std::cout << "\nDigite o saldo  
médio: ";  
std::cin >> saldo;  
  
if(saldo>=0 && saldo<=600){  
    if(saldo<=200){  
        porc = 0;  
    } else{  
        if(saldo<=400){  
            porc=20;  
        } else{  
            porc=30;  
        }  
    }  
} else{  
    porc= 40;  
}  
  
std::cout << "\n porcentagem : "  
<<porc <<"  % crédito disponível R$"  
<<(saldo*(1 + (porc/100) ));  
  
}
```

11) Escrever um algoritmo que lê um conjunto de 4 valores i, a, b, c, onde i é um valor inteiro e positivo e a, b, c, são quaisquer valores reais e os escreva. A seguir: Se i=1 escrever os três valores a, b, c em ordem crescente.

Se i=2 escrever os três valores a, b, c em ordem decrescente.

Se i=3 escrever os três valores a, b, c de forma que o maior entre a, b, c fique dentre os dois.

14) Escrever um algoritmo que lê um conjunto de 4 valores i, a, b, c, onde i é um valor inteiro e positivo e a, b, c, são quaisquer valores reais e os escreva. A seguir:

Se i=1 escrever os três valores a, b, c em ordem crescente.

Se i=2 escrever os três valores a, b, c em ordem decrescente.

Se i=3 escrever os três valores a, b, c de forma que o maior entre a, b, c fique dentre os dois.

TESTE DE MESA

Pseudocódigo

Variáveis

num: inteiro

Início

num<-0

Fim

Codificação C++

12) Um vendedor necessita de um algoritmo que calcule o preço total devido por um cliente. O algoritmo deve receber o código de um produto e a quantidade comprada e calcular o preço total, usando a tabela abaixo:

Código do produto Preço unitário

1	R\$ 32,00
2	R\$ 45,00
3	R\$ 37,00
4	R\$ 33,00

	TESTE DE MESA			
Pseudocódigo				
Variáveis				
num: inteiro				
Início				
num<-0				
Fim				
Codificação C++				

13) Escrever um algoritmo que lê um valor em Reais (R\$) e calcule qual o menor número possível de notas de 100, 50, 10, 5 e 1 em que o valor lido pode ser decomposto. Escrever o valor lido e a relação de notas necessárias.

TESTE DE MESA

Pseudocódigo

Variáveis
num: inteiro

Início
num<-0

Fim

Codificação C++

```
#include <iostream>
using namespace std;

int main() {
    int valor=0, nota100=0, resto50=0,
    nota50, resto10=0, nota10=0,
    resto5=0,nota5=0;

    cin >> valor;

    if(valor%100>0) {
        nota100 = (valor-valor%100)/100;
        if(valor%50>0){
            resto50= valor%100;
            nota50 = (resto50-
valor%50)/50;
            if(valor%10>0){
                resto10= valor%50;
                nota10 = (resto10 -
valor%5)/10;
                if(valor%5>0){
                    resto5= valor%10;
                    nota5= (resto5-
valor%5)/5;
                } else {
                    resto5= valor%10;
                    nota5= (resto5-
valor%5)/5;
```



```
    }  
    } else {  
        resto10= valor%50;  
        nota10 = (resto10 -  
valor%5)/10;  
    }  
    }  
    else {  
        resto50= valor%100;  
        nota50 = (resto50-  
valor%50)/50;  
    }  
} else {  
    nota100 = (valor-valor%100)/100;  
}  
  
cout <<"\n notas de 100 " <<nota100;  
cout <<"\n notas de 50 " <<nota50;  
cout <<"\n notas de 10 " <<nota10;  
cout <<"\n notas de 5 " <<nota5;  
cout <<"\n notas de 1 " <<valor%5;  
}
```

15) As maçãs custam R\$ 1,30 cada se forem compradas menos de uma dúzia, e R\$ 1,00 se forem compradas pelo menos 12. Escreva um programa que leia o número de maçãs compradas, calcule e escreva o custo total da compra.