

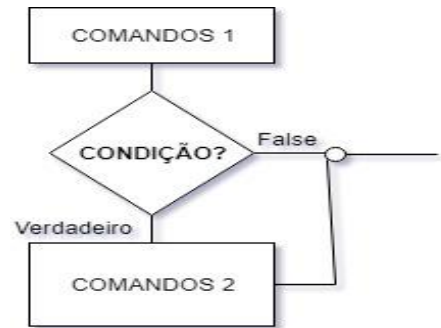
Roteiro 3 - Estrutura Condicional

Uma das tarefas fundamentais de qualquer programa é decidir o que deve ser executado, para isso, temos comandos de decisão que permitem determinar qual a ação deve ser tomada a partir de um resultado de uma expressão condicional. Em C/C++ temos três comandos de decisão:

if
if-else
switch-case

Estrutura Condicional Simples

```
//Bibliotecas
public class condicionalSimples{
    public static void main (String [ ] args )
    {
        if( condição ) {
            bloco de comandos 1;
        }
    }
}
```



- Observe que o **bloco de comandos** só será executado se a **condição** for verdadeira. Uma condição é uma comparação que possui dois valores possíveis (verdadeiro ou falso).
- Necessário a utilização de **chaves** { } quando houver mais de um comando no **Bloco de comandos**. Casos quando o Bloco de comandos possui apenas um comando, não precisa utilizar **chaves**.
- No exemplo acima, o bloco de comandos entre chaves só será executado se a condição for verdadeira; caso a condição seja falsa, a execução do programa passa para o comando seguinte à chave de fechamento do bloco de comandos.

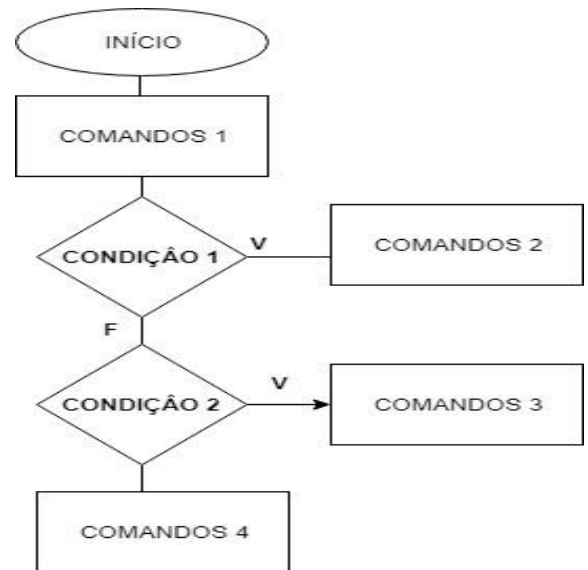
Exemplo: O exemplo identifica qual o maior entre dois números digitados.

```
1 package condicional;
2 public class Condicional {
3     public static void main(String[] args) {
4         int num1 = 10, num2 = 5;
5
6         if(num1>num2){
7             System.out.println(num1+" é maior que "+ num2);
8         }
9         if(num2>num1){
10            System.out.println(num2+" é maior que "+ num1);
11        }
12        if(num1==num2){
13            System.out.println(num1+" é igual à "+ num2);
14        }
15    }
16 }
```

Note que no exemplo anterior 3 comparações são realizadas, sendo que uma delas é desnecessária, como evitá-la? - **A resposta é utilizar a estrutura condicional composta.**

Estrutura Condicional Composta

```
//Bibliotecas
public class condicionalSimples{
    public static void main (String [ ] args )
    {
        if(condicao1) {
            bloco de comandos 1;
        }
        else if(condicao2) {
            bloco de comandos2;
        }
        else{
            bloco de comandos3;
        }
    }
}
```



- Na estrutura condicional composta existe um **bloco de comandos** a ser realizado caso a **condição** testada seja falsa.
- No exemplo acima, se a **condição1** for verdadeira, o **bloco de comandos 1** é executado; **senão** se a **condição2** é verificada, caso a mesma seja verdadeira, executa-se o **bloco de comandos 2**; **senão**, o **bloco de comandos 3** é executado.
- **Obs.:** O comando **else** deve ser sempre precedido de um **if**.

Exemplo: Identifica maior entre 2 números digitados utilizando estrutura condicional composta.

```
1 package condicional;
2 public class Condicional {
3     public static void main(String[] args) {
4         int num1 = 10, num2 = 5;
5
6         if(num1>num2){
7             System.out.println(num1+" é maior que "+ num2);
8         }
9         else if(num2>num1){
10            System.out.println(num2+" é maior que "+ num1);
11        }
12        else{
13            System.out.println(num1+" é igual à "+ num2);
14        }
15    }
16 }
```



Como as **condições das estruturas condicionais** podem ser compostas por expressões aritméticas, operadores relacionais e operadores lógicos, vamos defini-los a seguir.

Operadores relacionais:

Operador	Exemplo	Comentário
<code>==</code>	<code>x == y</code>	O conteúdo de X é igual ao conteúdo de Y
<code>!=</code>	<code>x != y</code>	O conteúdo de X é diferente do conteúdo de Y
<code><=</code>	<code>x <= y</code>	O conteúdo de X é menor ou igual ao conteúdo de Y
<code>>=</code>	<code>x >= y</code>	O conteúdo de X é maior ou igual ao conteúdo de Y
<code><</code>	<code>x < y</code>	O conteúdo de X é menor que o conteúdo de Y
<code>></code>	<code>x > y</code>	O conteúdo de X é maior que o conteúdo de Y

Operadores lógicos:

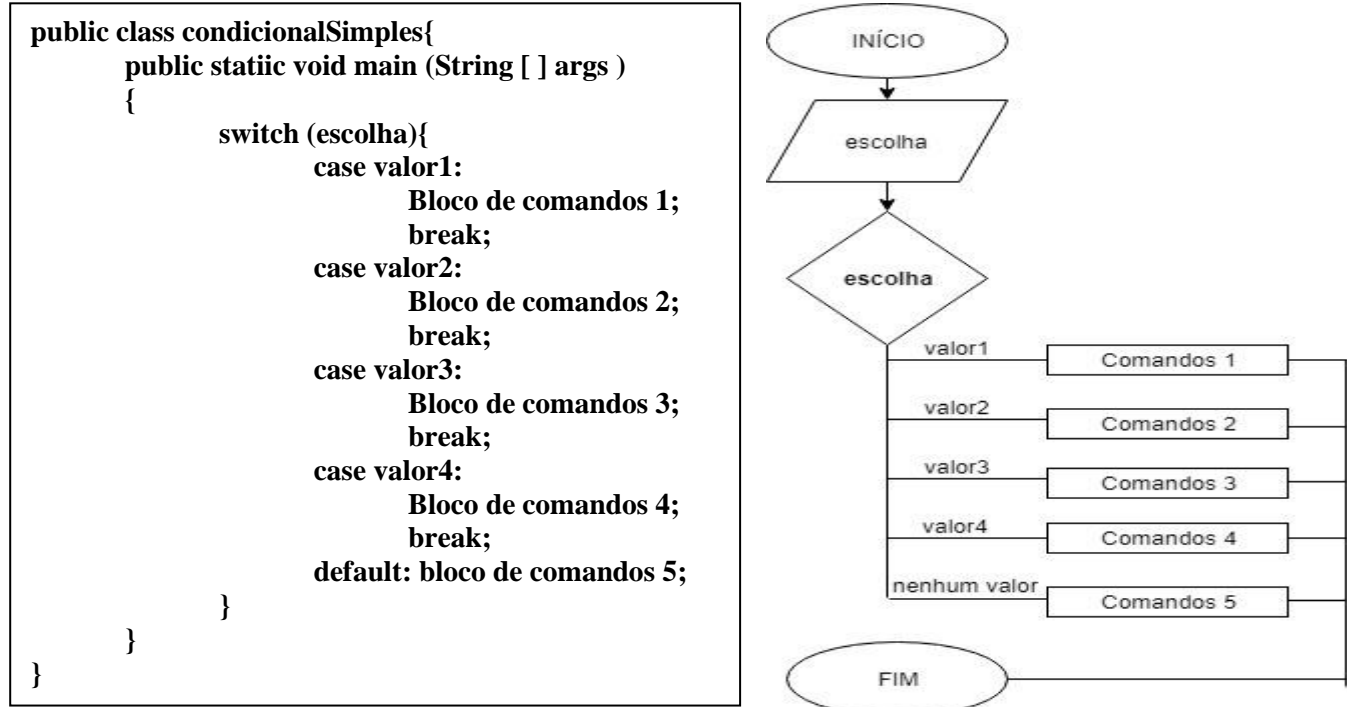
Operador	Exemplo	Comentário
<code>&&</code>	<code>if (x >= 3 && x < 7)</code>	Significa que a condição verificada será <code>X >= 3 E X < 7</code> , ou seja, a condição será verdadeira se X estiver entre 3 e 7
<code> </code>	<code>if (x >= 3 x >= 7)</code>	Significa que a condição verificada será <code>X >= 3 OU X <= 7</code> , ou seja, tanto se X for maior que 3 como se X for maior que 7 a condição será verdadeira
<code>!</code>	<code>!x</code>	A <code>!</code> (NÃO) é a <i>negação</i> , ou seja, muda um valor lógico de verdadeiro para falso e de falso para verdadeiro - Resulta 1 (verdadeiro) somente se x for falsa. - Resulta 0 (falso) somente se x for verdadeiro.

Tabela verdade:

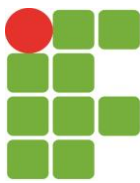
Tabela E (<code>&&</code>)	Tabela OU (<code> </code>)	Tabela NÃO (<code>!</code>)
<code>V e V = V</code>	<code>V ou V = V</code>	<code>Não V = F</code>
<code>V e F = F</code>	<code>V ou F = V</code>	<code>Não F = V</code>
<code>F e V = F</code>	<code>F ou V = V</code>	
<code>F e F = F</code>	<code>F ou F = F</code>	

Estrutura switch-case

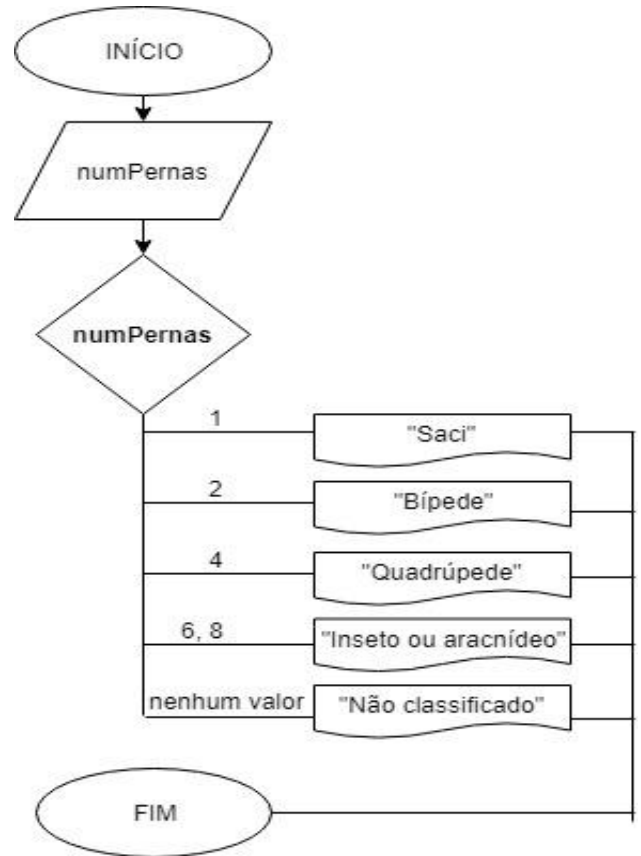
O comando **switch** é uma estrutura condicional de decisão múltipla que testa o valor de uma expressão e desvia o fluxo de execução do programa de acordo com o resultado do teste.



- Em situações mutuamente exclusivas, isto é, se uma situação for executada, as demais **não serão** executadas, um comando seletivo é indicado (estrutura CASE).
- O comando **switch(variável)** avalia o valor da variável para decidir qual case será executado.
- Cada case está associado a UM possível valor da variável, que deve ser obrigatoriamente, do tipo **char, int, byte, short ou String**.
- O comando **break;** deve ser utilizado para impedir a execução dos comandos definidos nos **cases subsequentes**.
- O comando **default:** é executado quando o valor da variável **não** coincidir com nenhum dos especificados nos *cases*.



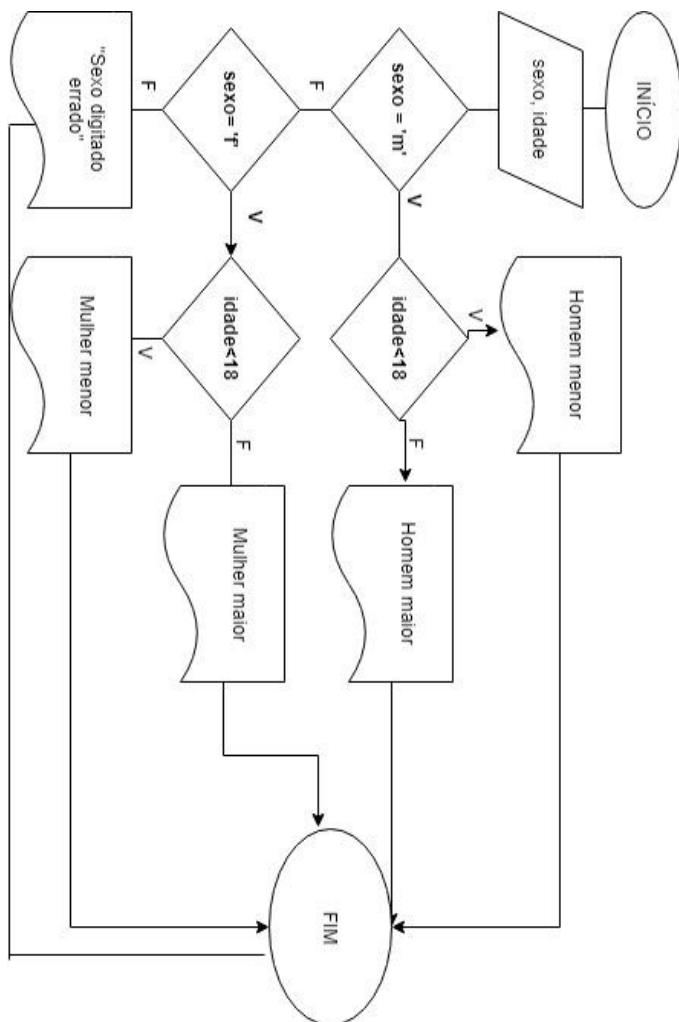
```
package condicional;  
public class Condicional {  
    public static void main(String[] args) {  
        int numPernas = 2;  
        switch(numPernas) {  
            case 1:  
                System.out.println("Saci");  
                break;  
            case 2:  
                System.out.println("Bípede");  
                break;  
            case 4:  
                System.out.println("Quadrúpede");  
                break;  
            case 6:  
            case 8:  
                System.out.println("Inseto ou aracnídeo");  
                break;  
            default:  
                System.out.println("Não classificado");  
        }  
    }  
}
```



Estrutura condicional aninhada

A estrutura condicional aninhada é utilizada em situações onde necessita-se a verificação de condições sucessivas, ou seja, uma ação será executada somente no caso de um conjunto anterior de ações seja satisfeito. A estrutura representa nada mais do que o encadeamento de estruturas de condicionais. Então é importante ficar claro que qualquer comando pode fazer parte de um conjunto de instruções de um **if** ou **if-else**, inclusive outros comandos **if** ou **if-else**.

O fluxograma e o código JAVA a seguir apresentam um exemplo que verifica, dados o sexo e idade de uma pessoa, se ela é homem maior de idade ou homem menor de idade; mulher maior de idade ou mulher menor de idade. Note que da forma como se modelou a resolução do problema, a condicional de verificar a idade é **aninhada** à condicional que verifica o sexo.



```

float idade;
char sexo;
Scanner teclado = new Scanner(System.in);
System.out.println("Digite o sexo:");
sexo = (char) teclado.next().charAt(0);
sexo = Character.toLowerCase(sexo);
System.out.println("Digite a idade:");
idade = teclado.nextFloat();
if(sexo=='m'){
    if(idade<18){
        System.out.println("Homem menor");
    }
    else{
        System.out.println("Homem maior");
    }
}
else if(sexo=='f'){
    if(idade<18){
        System.out.println("Mulher menor");
    }
    else{
        System.out.println("Mulher maior");
    }
}
else{
    System.out.println("Sexo digitado errado!");
}
  
```

Exemplos

1 - Digite e compile os códigos dos exemplos abaixo colocando comentários nas partes que você considera importante.

Exemplo1: Identifica intervalos

```
1 package condicional;
2 import java.util.Scanner;
3 public class Condicional {
4     public static void main(String[] args) {
5         float num1;
6         Scanner teclado = new Scanner(System.in);
7         num1 = teclado.nextFloat();
8         if(num1<0){
9             System.out.println(num1+" é negativo!");
10        }
11        else if(num1>=0 && num1<=7){
12            System.out.println(num1+" está no intervalo de 0 à 7");
13        }
14        else{
15            System.out.println(num1+" é maior que 7");
16        }
17    }
18 }
```

Exemplo 2: Identifica valores.

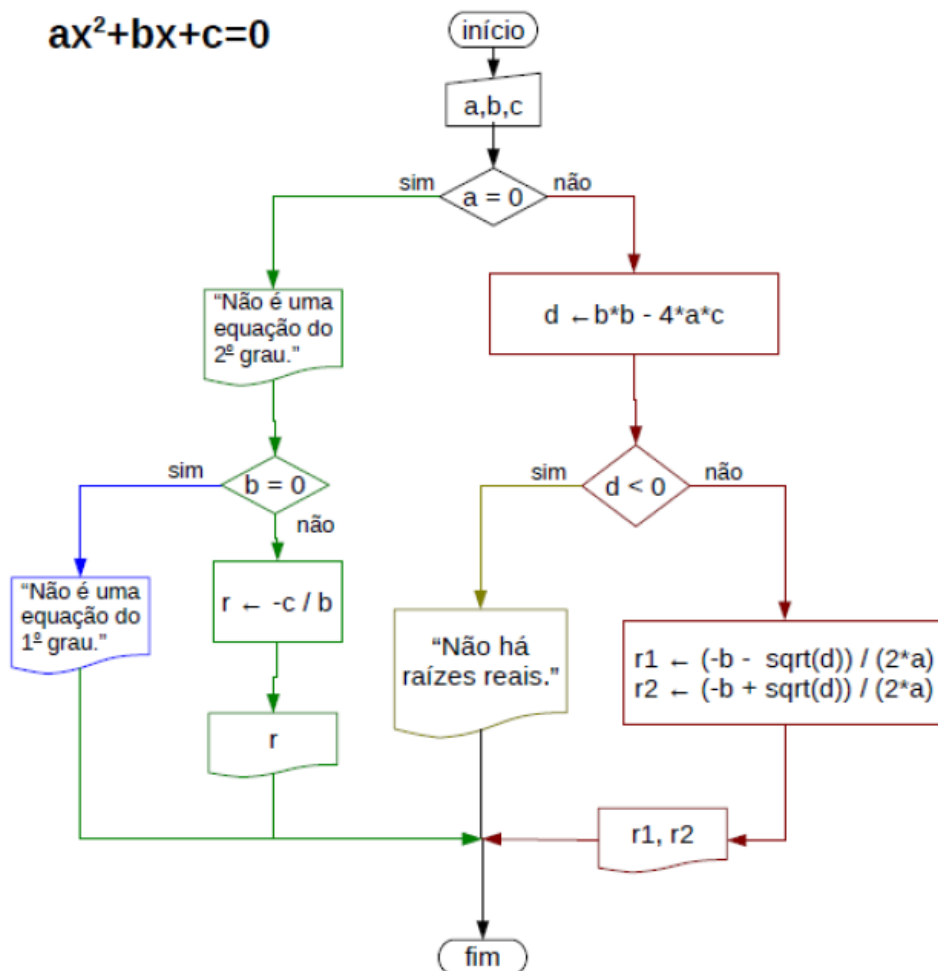
```
1 package condicional;
2 import java.util.Scanner;
3 public class Condicional {
4     public static void main(String[] args) {
5         int num1;
6         Scanner teclado = new Scanner(System.in);
7         num1 = teclado.nextInt();
8         switch(num1){
9             case 0:
10                System.out.println("Zero!");
11                break;
12             case 1:
13                System.out.println("Um!");
14                break;
15             case 2:
16                System.out.println("Dois!");
17                break;
18             case 3:
19                System.out.println("Três!");
20                break;
21             default:
22                System.out.println("Valor diferente de zero, um, dois e três!");
23            }
24        }
25    }
```


Exemplo 3: Escolha entre opções

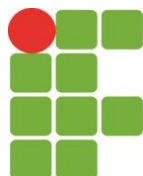
```
1 package condicional;
2 import java.util.Scanner;
3 public class Condicional {
4     public static void main(String[] args) {
5         char opcao;
6         Scanner teclado = new Scanner(System.in);
7         opcao = teclado.nextLine().charAt(0);
8         switch(opcao) {
9             case 'a':
10                 System.out.println("Opção \"a\" escolhida!");
11                 break;
12             case 'b':
13                 System.out.println("Opção \"b\" escolhida");
14                 break;
15             case 'c':
16                 System.out.println("Opção \"c\" escolhida");
17                 break;
18             case 'd':
19                 System.out.println("Opção \"d\" escolhida");
20                 break;
21             default:
22                 System.out.println("Opção inválida!");
23         }
24     }
25 }
```


Exercícios

1. Faça um programa que identifique se um número digitado pelo usuário é positivo, negativo ou zero.
2. Faça um algoritmo que dado um número digitado pelo usuário mostre se o mesmo é par ou ímpar.
3. Crie um programa que solicite a entrada de dois números, em seguida imprima na tela o quadrado do menor e a raiz quadrada do maior número, caso os números sejam iguais mostre uma mensagem informando que os números são iguais. Lembre-se de verificar as condições para um número possuir raiz quadrada real.
4. Refaça o exercício referente à equação do segundo grau do roteiro anterior considerando todas as condições possíveis, conforme fluxograma abaixo.



5. O custo de um carro novo ao consumidor é a soma do custo de fábrica, a porcentagem do distribuidor e os impostos. O valor pago pela distribuição e dos impostos é calculado com base no custo de fábrica. As porcentagens estão apresentadas na tabela abaixo. Faça um programa que receba o custo de fábrica de um carro e mostre o custo ao consumidor



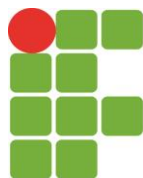
Custo de Fábrica	% Distribuidor	% Impostos
Até R\$ 28000.00	5	Isento
Entre R\$28000.01 e R\$45000.00	10	15
Acima de R\$45000.00	15	20

6. Faça um programa que simule uma calculadora simples. Para isso, o usuário deverá entrar com dois números quaisquer e a operação que deseja realizar: '+' soma; '-' subtração; '*' multiplicação e '/' divisão. Lembre-se das condições para realizar a divisão. Utilize a estrutura **switch/case** e apresente um menu de opções para o usuário.
7. Faça um programa que solicite ao usuário que digite um caracter e em seguida imprima se o caracter digitado é ou não uma vogal.
8. Uma empresa decide dar um aumento de 30% aos funcionários com salário inferior à R\$700,00 e 10% aos funcionários com salário superior ou igual à R\$700,00. Faça um algoritmo que calcule o reajuste e mostre o novo salário.
9. Na cidade de Sabará, para transferências de veículos, a SABATRAN cobra uma taxa de 1% para carros fabricados antes de 1990 e uma taxa de 1.5% para os fabricados de 1990 em diante. Essa taxa está incidindo sobre o valor de tabela do carro. Implemente um algoritmo que lê o ano de fabricação e o preço do carro e a seguir calcula e imprime imposto a ser pago.
10. Criar um algoritmo que a partir da idade e peso do paciente calcule a dosagem de determinado medicamento e **imprima a receita informando quantas gotas do medicamento** o paciente deve tomar por dose. Considere que o medicamento em questão possui 500 mg por ml, e que cada ml corresponde a 20 gotas.
 - Adultos ou adolescentes desde 12 anos, inclusive, se tiverem peso igual ou acima de 60 quilos devem tomar 1000 mg; com peso abaixo de 60 quilos devem tomar 875 mg.
 - Para crianças e adolescentes abaixo de 12 anos a dosagem é calculada pelo peso corpóreo conforme a tabela a seguir:

Peso	Dosagem
5 kg a 9 kg	125 mg
9.1 kg a 16 kg	250 mg
16.1 kg a 24 kg	375 mg
24.1 kg a 30 kg	500 mg
Acima de 30 kg	750 mg

11. Crie um algoritmo que receba o valor de x , calcule e imprima o valor de $f(x)$.

$$f(x) = \begin{cases} 1, & \text{se } x \leq 1 \\ 2, & \text{se } 1 < x \leq 2 \\ x^2, & \text{se } 2 < x \leq 3 \\ x^3, & \text{se } x > 3 \end{cases}$$



12. Faça um algoritmo que receba a idade de um nadador e mostre a categoria do mesmo. Faça os testes necessários em relação a idade inválida.

Categoria	Operação
Infantil	5 a 7 (inclusive)
Juvenil	8 a 10 (inclusive)
Adolescente	11 a 15 (inclusive)
Adulto	16 a 40 (inclusive)
Sênior	Acima de 40 (inclusive)

13. O IMC (Índice de Massa Corporal) é um critério da Organização Mundial de Saúde para dar uma indicação sobre a condição de peso de uma pessoa adulta. A fórmula é $IMC = PESO / (ALTURA^2)$. Elabore um algoritmo que leia o peso e a altura de um adulto e mostre sua condição.

IMC em adultos	Condição
abaixo de 18.5	abaixo do peso
entre 18.5 e 25	peso normal
entre 25 e 30	acima do peso
acima de 30	Obeso

14. Elabore um programa que calcule e exiba a tensão S de uma barra cilíndrica de diâmetro D submetida a uma carga Q . Os valores de D e Q devem ser fornecidos pelo usuário do programa via teclado. Utilize a fórmula:

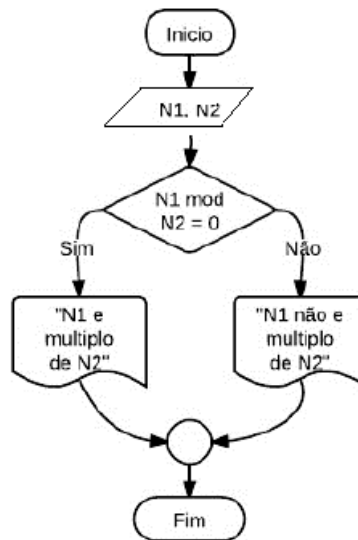
$$S = \frac{4 * Q}{\pi * D^2} * n$$

onde, $n = 2$ para $D > 100$; $n = 6$ para $D < 50$ e; $n = 4$ para outros valores de D .

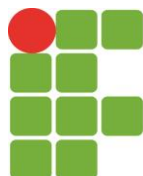
15. Faça um algoritmo que receba duas datas (dia, mês e ano) e determine a maior entre elas. Você deverá receber as datas no formato de uma String e fazer as devidas comparações. **Pesquisem sobre as classes Date e Calendar.**
16. Faça um programa que receba de entrada a data de votação do ano atual e a data de nascimento de uma pessoa. Utilizando as informações pesquisadas na questão anterior, verifique se essa pessoa não pode votar, é obrigada a votar ou o voto é facultativo, segundo as regras brasileiras.

Idade	Voto
Menor de 16	Não vota
16 à 18 e acima de 70	Facultativo
Entre 18 e 70	Obrigatório

17. Faça um programa que receba de entrada três valores. Verificar se esses valores podem formar um triângulo. Caso não, informar que não forma um triângulo; caso sim, imprima que tipo de triângulo pode ser formado (escaleno, isósceles ou equilátero).
18. Implemente um programa que receba um número inteiro de 3 dígitos de entrada. Após isso, verifique se o mesmo é palíndromo, ou seja, é o mesmo número lido da direita para esquerda ou vice-versa. Ex.: 121, 353, etc. **Não é permitido a utilização de funções prontas do JAVA, tal como o método reverse.**
19. Implemente um algoritmo que receba de entrada um valor inteiro com 4 dígitos e verifique se o mesmo pertence à família dos binários. Para isto, “desmembre” o número digitado verifique se foram digitados somente bits. Caso o número digitado seja binário, seu algoritmo deve o converter para decimal e mostrar o resultado.
20. Elabore um programa que realize o mesmo procedimento do fluxograma abaixo. A função **mod** utilizada no fluxograma representa o resto da divisão inteira.



21. Implemente um programa que receba um número inteiro positivo representando um ano qualquer e imprima se o ano é bissexto ou não. Um ano bissexto é definido quando: $(ano \% 4 = 0 \text{ e } (ano \% 100 \neq 0 \text{ ou } ano \% 400 == 0))$.
22. Faça um algoritmo que receba três números e mostre-os em ordem crescente e decrescente.
23. O número de inscrição no CPF é composto de onze dígitos decimais, sendo os oito primeiros aleatoriamente designados no momento da inscrição. Já o nono (antepenúltimo) dígito indica a região fiscal responsável pela inscrição (MG é a região 6, portanto em todos CPF's emitidos em MG o nono dígito é 6). Por fim, o décimo e o décimo primeiro são **dígitos verificadores calculados de acordo com um algoritmo definido pela Receita Federal** e publicamente conhecido. Referências abaixo:
 - a. <http://www.gerardocumentos.com.br/?pg=entenda-a-formula-do-cpf>
 - b. <https://bityli.com/CI4gC>



Assim, implemente um algoritmo que receba de entrada 11 dígitos de um CPF qualquer (**uma única variável inteira**) e verifique se o valor inserido é ou não um CPF válido. Utilize as referências citadas anteriormente para realizar os cálculos para validação.

24. Faça um programa que receba de entrada o código de um produto e a quantidade comprada do mesmo. Calcule e mostre:

- O preço unitário do produto, segundo Tabela I
- Preço total da nota
- O valor do desconto, segundo Tabela II, aplicado sobre o valor total da nota
- Preço final ao consumidor após desconto
- Não se esqueça de verificar a existência do produto

Código	Valor unitário
1 a 10	R\$10,00
11 a 20	R\$20,00
21 a 30	R\$30,00
31 a 40	R\$40,00

Total da nota	% Desconto
Até R\$ 250	5%
Entre R\$250 e R\$500	10%
Acima de R\$500	15%

25. Faça um algoritmo que receba de entrada a altura e o sexo de uma pessoa ('M' - Masculino ou 'F' - Feminino), construa um algoritmo que calcule o peso ideal da pessoa de acordo com as regras:

- Homens: $(72.7 * \text{altura}) - 58$

- Mulher: $(62,1 * \text{altura}) - 44.7$

26. Faça um algoritmo que receba o salário base e o tempo de serviço de um funcionário. Calcule e mostre:

(i) O imposto, conforme tabela:

SALÁRIO BASE	% SOBRE O SALÁRIO BASE
< R\$ 200,00	Isento
Entre R\$ 200,00 (inclusive) e R\$ 450,00 (inclusive)	3%
Entre R\$ 450,00 e R\$ 700,00	8%
\geq R\$ 700,00	12%

(ii) A gratificação, conforme tabela:

SALÁRIO BASE	TEMPO DE SERVIÇO	GRATIFICAÇÃO
Superior a R\$ 500,00	Até 3 anos	20
	Mais de 3 anos	30
Até R\$ 500,00	Até 3 anos	23
	Entre 3 e 6 anos	35
	De 6 anos para cima	33

(iii) O salário líquido, ou seja, salário base menos imposto mais gratificação

(iv) A categoria que está na tabela a seguir:

SALÁRIO LÍQUIDO	CLASSIFICAÇÃO
Até R\$ 350,00	A
Entre R\$ 350,00 e R\$ 600,00	B
De R\$ 600,00 para cima	C