

Prodabel - Inclusão Digital

Tecnologia

<Introdução a Lógica de Programação com Javascript/>

1 - Apresentação



2 - A Lógica de Programa...



3 - Variáveis



4 - Tipos de dados



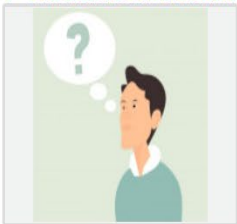
5 - Entrada e Saída de da...



6 - Operadores



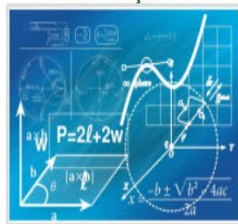
7 - Estruturas Condicionais



8 - Estruturas de Repetição



9 - Funções



10 - Considerações Finais



11 - Pesquisa de Satisfação



12 - Certificado



Sumário

Apresentação do Curso.....	4
1.0 Introdução a Lógica de Programação	5
1.1 O pensamento Lógico	5
1.2 O que é Lógica de Programação?	7
1.3 O que é Algoritmo?	8
1.4 O Apresentação do Visual Studio.....	10
1.5 A computação e a programação	10
Qual linguagem vou utilizar no Curso?	11
2.0 Entrada e Saída de Dados	11
2.1 Dados e Informação	11
2.2 Dispositivos de Entrada e Saída	12
2.3 Entrada e Saída de Dados: Simbologia	12
3.0 Tipos de Dados e Variáveis	14
3.1 O que são variáveis?	15
3.2 Tipos de Dados.....	16
4.0 Operadores Aritméticos I.....	17
4.1 Operações básicas.....	17
4.2 Operadores Aritméticos (Conceitos e prática)	18
5.0 Operadores Aritméticos II.....	18
6.0 Desvios Condicionais I.....	19
6.1 Conceitos: Tarefa e Condição	20
6.2 Comando IF (SE)	20
6.3 Comando ELSE (SE NÃO)	21
6.4 Comando ELSE IF (SE NÃO SE).....	21
6.5 Comando Switch	21
7.0 Operadores Relacionais	22
7.1 Operadores Relacionais – Símbolos.....	23
8.0 Estruturas de Repetição.....	23
8.1 O Comando While (Enquanto).....	25
8.2 Comando Do While (Faça Enquanto).....	25
8.3 Comando For.....	25
8.4 Comando Break e Continue	26
8.5 Comentários.....	26
9.0 Operadores Lógicos	27

9.1 Premissas (Proposição)	27
10.0 Desvios Condicionais II.....	28
10.1 Conceitos: Escolha, Switch/case	28
11.0 Vetores.....	28
12.0 Matrizes	29
13.0 Funções	29
14.0 Teste de mesa e Bibliotecas.....	30
15.0 Arquivos	30

Apresentação do Curso



Antes de começar a falar deste assunto de programação e dessas palavras que agora parecem um pouco complicadas. Gostaria de falar diretamente com você meu Aluno. Você é muito bem vindo em nosso curso e o intuito aqui é que você aprenda e comece sua carreira como programador. O nosso curso não vai ensinar tudo a você, esse mundo da programação é gigantesco e aqui nós não vamos esgotar o assunto, na verdade nem chegamos perto disso!

O intuito aqui é que você seja apresentado a técnicas e assuntos que vão ajudar você nessa viagem, isso eu considero apenas uma iniciação, mas isso não quer dizer que você não vai aprender muita coisa, muito pelo contrário, você vai aprender muita coisa interessante. Então, convido você a mergulhar nesse tema comigo e aprender uma profissão bem interessante. Para começar assista ao vídeo abaixo onde falo um pouco sobre o curso. Vamos lá?

Carga Horária do módulo:

36 horas

Atenção:

Nas aulas deste módulo, você terá a oportunidade de acompanhar a teoria junto com a prática. Os vídeos irão te ensinar os conceitos de lógica e você também verá como aplicar o que aprendeu na prática.

- Separe um caderno e uma caneta para que você possa fazer suas anotações durante o vídeo.

- Sempre que precisar pausar o vídeo e voltar em algum conceito, fique à vontade.

- Para certificação final você deverá cumprir, em cada módulo, no mínimo 60% de frequência e participação no curso e 60% de aproveitamento do conteúdo, que será avaliado pela produção dos trabalhos.

Se organize para ter um bom aprendizado.

**DIRETORIA DE INCLUSÃO DIGITAL - PRODABEL
2020**

1.0 Introdução a Lógica de Programação

Bons estudos neste novo módulo.

Os conteúdos deste módulo irão abordar sobre algoritmos e introdução a programação.

O que são algoritmos e introdução a programação?

A partir de agora, vamos aprender a transformar a vida real em sistemas. Vamos seguir instruções, passo a passo, na solução de um problema proposto através de comandos definidos e que funcionam com padrão.

Algoritmos não dizem respeito somente sobre computadores e nem exclusivamente sobre linguagem de programação.

OBS: Os conteúdos serão apresentados sequencialmente para que facilite o entendimento e a prática de construção de um raciocínio lógico. Assim, vamos apresentando os temas utilizando vídeos que apresentam conceitos e exercícios que vão organizando o pensamento e a criação de um raciocínio lógico. Dedique seu tempo em ver e compreender a proposta de cada vídeo. Ao final do módulo, você vai concluir que estará preparado, com aprendizados conceituais e prático estará e pronta para entender os conteúdos do próximo módulo. Vamos lá.

1.1 O pensamento Lógico

Começando...

Preste atenção na sua rotina do dia a dia e verá que você utiliza algoritmos diariamente em diversas ocasiões, por exemplo, para fazer um bolo. Abaixo, temos uma receita completa e ela servirá de referência para iniciar o aprendizado de como pensar e programar.

RECEITA

Ingredientes:

Bolo:

3 ovos

1 xícara de óleo de soja

2 cenouras bem grandes ou 3 se forem médias

2 xícaras de farinha de trigo

2 xícaras de açúcar

1 colher (sopa) de fermento em pó.

Cobertura:

4 colheres cheias de açúcar

2 colheres cheias de chocolate em pó

1 colher de manteiga

Um pouquinho de leite, apenas para desmanchar

Modo de preparo:

Massa:

Bata a cenoura, os ovos e o óleo no liquidificador.

Em uma vasilha à parte, misture os outros ingredientes e coloque em um tabuleiro untado.

Asse em forno médio, pré-aquecido, por 30 a 40 minutos.

Cobertura:

Coloque todos os ingredientes em uma panela e leve ao fogo, sempre mexendo, até ferver um pouco.

Faça furos no bolo para o chocolate escorrer pelo meio do bolo.

Fonte: <https://www.tudogostoso.com.br/receita/48426-bolo-de-cenoura-fofinho-com-cobertura-de-chocolate.html> em 25/08/2020

Algoritmo enumerado

Agora, vamos pensar na sequência de comandos que são necessários para executar essa receita.

1 – Pegue os ingredientes com a mão que são: 3 copos de farinha de trigo, 2 copos de açúcar, 1 colher (sobremesa) de fermento em pó, 3 cenouras grandes, 1 copo de óleo e 4 ovos. (Lembrando que os copos são em média de 300ml).

2 – Pegue primeiro a faca e descasque a cenoura e corte-a.

3 – Pegue o liquidificador.

4 – Coloque no liquidificador as cenouras (descascadas e picadas), o óleo e os ovos (Lembrando de quebrá-los).

5 – Depois junte todos os ingredientes, bata mais um pouco e reserve.

6 – Ligue o botão do forno liberando o gás para ele ser ligado, no caso isso serve para ir aquecendo em temperatura de 180 a 210 graus.

7 – Pegue uma forma retangular ou redonda, pegue a manteiga na geladeira junto com um papel toalha.

8 – Abra a tampa da manteiga, pegue o papel já dobrado e passe suavemente na manteiga.

9 – Depois passe o papel com manteiga em toda forma de bolo (menos do lado de fora).

10 – Coloque a massa pronta na forma de bolo.

11 – Coloque a forma no forno pré-aquecido e deixe por uns 40 minutos.

12 – Passado os 40 minutos pegue um garfo e coloque no bolo, se não sair massa no garfo retire o bolo do forno com um guardanapo, se sair deixe mais um tempo no forno lembrando de sempre furá-lo para ver se já está bom.

13 – Passe uma faca envolta do bolo ainda dentro da forma, pegue um recipiente que caiba o bolo, e vire-o nele (lembrando de pegar a forma ainda com um guardanapo pois estará quente).

14 – Pegue uma faca e corte o bolo.

Cobertura:

1 – Pegue o chocolate em pó, uma panela, uma colher, uma tesoura, manteiga e leite condensado.

2 – Ligue o fogo, girando o botão do fogão e apertando o botão do fogo, para o próprio ser ligado (na temperatura que quiser), pegue a panela circular e coloque no fogo.

3 – Em seguida pegue a colher com a manteiga, pegando duas colheres rasas. Despeje essa manteiga da colher na panela.

4 – Pegue a tesoura e abra o leite condensado, despeje na panela.

5 – Pegue a colher e pegue em média de 4 a 6 colheradas de chocolate em pó, e também coloque na panela.

6 – Comece a mexer sem parar com a colher, até começar a ter uma consistência mais homogênea (a ficar mais durinho ou incorporado).

7 – Então, desligue o fogo não se esquecendo de voltar o botão ao modo inicial.

8 – Pegue a panela com um guardanapo e despeje a cobertura no bolo.

9 – Pegue um pedaço de bolo em um prato, junto a um garfo e delicie-se.

1.2 O que é Lógica de Programação?

O que é lógica?

Uso de raciocínio em alguma atividade que você vai realizar.

Por exemplo: Traçar um caminho para ir em um determinado lugar, você precisará usar a lógica (raciocínio) para não dar volta demais e não se perder.

O que é Lógica de Programação?

Olhe que interessante:



...lógica é a 'ciência das regras do pensamento'.

Esse trecho de texto foi retirado do link:

<https://educacao.uol.com.br/disciplinas/filosofia/logica---introducao-uma-porta-ao-mundo-da-filosofia-e-da-ciencia.htm?cmpid=copiaecola>

Se a palavra lógica pode ser considerada a ciência das regras do pensamento e a programação pode ser considerada a tarefa de criar programas de computador, então a lógica de programação seria a técnica de juntar as regras do nosso pensamento para atingirmos o objetivo de criar um código que vai programar o computador para realizar uma tarefa.

Assim podemos também definir que a lógica de programação é a elaboração das sequências de ações que o computador deve tomar para atingir um determinado objetivo. Esse processo vai envolver o uso de algumas estruturas condicionais e de repetição que vamos estudar mais à frente em nosso curso.

AÇÕES + LÓGICA = BOA SOLUÇÃO

Por que vamos estudar Lógica?

É a base para nosso estudo de programação. Será o jeito de conseguir se organizar e apresentar suas ideias diante de uma solução a ser pensada e dada para uma necessidade ou problema. Isso acaba facilitando a sua vida, quando tiver que analisar ideias de outras pessoas também.

Existe lógica no dia a dia das pessoas? Pense em várias situações que acontecem com todos nós...

a) A gaveta está fechada

A agenda está na gaveta

Preciso primeiro abrir a gaveta, para depois pegar a agenda.

b) Ana é mais velha do que João

João é mais velho do que Pedro

Portanto, Ana é mais velha do que Pedro.

1.3 O que é Algoritmo?

São passos (chamados de comandos ou instruções) que você pode usar para conseguir resolver um problema.

As instruções do seu algoritmo podem ser claras e objetivas, ou seja, você não precisa dar voltas para resolver um problema simples.

Problemas? Que tipo de problemas?

No nosso dia a dia, temos diversos problemas, como:

Uma dúvida ou questão

Algo difícil de explicar

Questões de matemática

- Problema: lâmpada queimada que deve ser trocada, atividade do dia a dia.

Solução:

Pegue uma escada;
Posicione a escada embaixo da lâmpada;
Busque uma lâmpada nova;
Suba na escada;
Retire a lâmpada;
Coloque a lâmpada nova;
Acenda o interruptor.

- Problemas: Como se vestir pela manhã?**Solução:**

Vestir a calça;
Calçar as meias;
Vestir a camisa.

- Algoritmo Recepcionista de Cinema

Início

- Solicitar ao cliente o bilhete do filme
- Conferir a data e o horário do filme no bilhete.
Se data/hora atual maior que data/hora do filme + 30 minutos

Então

- Informar ao cliente que o tempo limite para entrada foi excedido.
- Não permitir a entrada.
Senão Se data/hora atual menor que data/hora do filme - 30 minutos

Então

- Informar ao cliente que a sala do filme ainda não foi liberada para entrada.
- Não permitir a entrada.

Senão

- Permitir a entrada.
- Indicar ao cliente onde fica a sala do filme.

Fim-Se

Fim

Esse exemplo acima é um algoritmo do passo a passo de como uma recepcionista de cinema deve trabalhar. Se eu estivesse programando um computador bastava traduzir esse passo a passo para uma linguagem de programação que eu conseguiria construir um programa de computador.

1.4 O Apresentação do Visual Studio

Link para VS Code: <https://code.visualstudio.com/download>



Vídeo Como instalar o VS Code - (Parte 1)

Interpretador de Código

Para que nossos códigos funcionem, precisamos também instalar um interpretador de código. Assista ao vídeo abaixo sobre o que é um interpretador de Código.

Link para Node JS: <https://nodejs.org/en/>

1.5 A computação e a programação

O que é computação?

A computação pode ser definida como a busca de uma solução para um problema a partir de entradas e têm seus resultados depois de trabalhada através de um algoritmo.

O que é programar?

Programar ou programação de computadores diz respeito ao processo de escrever códigos que criam um software, também chamado de programa de computador.

Com a programação podemos criar aplicativos de smartphone (Celular) ou sites.

Assim, para programar, vamos precisar utilizar de uma Linguagem de programação.

Em nosso curso, vamos estudar um pouco sobre os conceitos de lógica de programação utilizando uma a linguagem de programação chamada Javascript.

Fique tranquilo, nas próximas aulas vamos ensinar sobre Lógica de Programação, Linguagem de programação e sobre o que seria a Javascript.

Por que programar?

Aprender a programar é ter nas mãos o "poder" de melhorar ou facilitar a vida das pessoas. Isso acontece por meio da criação de softwares e aplicativos que possam resolver problemas reais da sociedade.

O que é Linguagem de Programação?



A linguagem de programação pode ser considerada uma escrita que nós programadores utilizamos e é entendida pelo computador para especificar as ações que queremos que ele execute. Ou seja, estamos dizendo ao computador o que ele deve fazer. Existem diversos tipos de linguagens de programação, diversas formas de utilizar uma linguagem de programação e diversos paradigmas (modelos) de programação, todos com o intuito de criar programas de computador para as milhares de tarefas que queremos que o computador faça para nós.

Qual linguagem vou utilizar no Curso?

Em nosso curso, vamos aprender, utilizando a linguagem de programação denominada Javascript.

2.0 Entrada e Saída de Dados

Agora, vamos pensar como será nosso processo de organização do raciocínio lógico. Voltando à nossa receita de bolo da aula anterior. Pense que neste momento o bolo já está? Na forma? No forno? À mesa? Na geladeira? Na barriga? O que conseguimos reparar nestas perguntas? O bolo entra ou sai do forno? O bolo está na barriga? Como?

Vamos associar essa fase do bolo com os conteúdos dessa unidade. Iremos aprender sobre dispositivos/meios de entrada e saída de informações.

Este conteúdo liga você futuramente a:

- **Javascript** - Mostrando valores na tela - Prompt - Console
- **SQL** - Comandos de seleção e amostra SHOW, SELECT
- **PHP e HTML** - Comandos de exibição e entrada - ECHO - INPUT - AJAX - BUTTON – SCRIPTS

2.1 Dados e Informação

O que são Dados?

É um documento, uma informação ou um testemunho que permite chegar ao conhecimento de alguma situação. O dado não possui significado e não é possível compreender algo somente com ele, porque ele não tem sentido sozinho.

Você não pode tirar nenhuma conclusão sem ter dados a sua disposição.

Exemplo: Dados dos participantes do curso de Programação Web: Geralmente, quando pessoas fazem a inscrição para um curso, eles preenchem o formulário de inscrição. Este formulário contém dados do aluno como nome, e-mail, endereço, idade etc.

O que é informação?

É quem organiza os dados. Com ela é possível entender o significado e compreender melhor os dados.

É o que nos permite chegar a um conhecimento.

Exemplo: Preparar um Relatório de participantes do curso de Programação Web que tenham idade acima de 25 anos.

2.2 Dispositivos de Entrada e Saída

Chamamos de dispositivos de entrada e saída os responsáveis por extrair informação de um sistema de computador. Observe a figura:



Fonte: PUCRS/FACIN - Introdução à Informática

2.3 Entrada e Saída de Dados: Simbologia

Entrada e Saída de Dados







Entrada: Dados - Exemplo - (Escreva dois números na tela)

Processamento: Operações - Exemplo: (Os dois números são somados)

Saída: Informação - Exemplo: Apresentação do resultado da soma dos dois números digitados na entrada)

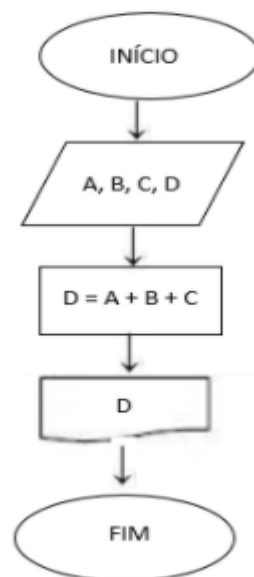
Simbologia: Fluxograma

Vamos aprender a utilizar símbolos gráficos predefinidos, eles nos ajudam a organizar o pensamento para a solução de um problema. Veja os símbolos que são mais utilizados na imagem abaixo:

Símbolo	Descrição
	Utilizado para indicar o início e o fim do algoritmo.
	Utilizado para representar a entrada de dados.
	Utilizado para representar a saída de dados
	Utilizado para indicar os cálculos e atribuição de valores.
	Utilizado para indicar a tomada de decisão e os possíveis desvios.
	Utilizado para indicar o fluxo de dados, conectando os blocos existentes.

Fonte Própria

Exemplo: Somar três números



Fonte: Própria

3.0 Tipos de Dados e Variáveis

Como ficou bom o nosso bolo, hummmmm?

Os amigos e amigas ficaram sabendo dessa receita especial e, agora, teremos de fazer um bolo um pouco maior!

Vamos relembrar a receita!

RECEITA

INGREDIENTES

- **Bolo:**
- 3 ovos
- 1 xícara de óleo de soja
- 2 cenouras bem grandes ou 3 se forem médias
- 2 xícaras de farinha de trigo
- 2 xícaras de açúcar
- 1 colher (sopa) de fermento em pó.
- **Cobertura:**
- 4 colheres cheias de açúcar
- 2 colheres cheias de chocolate em pó
- 1 colher de manteiga
- Um pouquinho de leite, apenas para desmanchar

MODO DE PREPARO

1. **Bolo:**
2. Bata a cenoura, os ovos e o óleo no liquidificador.
3. Em uma vasilha à parte, misture os outros ingredientes e coloque em um tabuleiro untado.
4. Asse em forno médio, pré-aquecido, por 30 a 40 minutos.
5. **Cobertura:**
6. Coloque todos os ingredientes em uma panela e leve ao fogo, sempre mexendo, até ferver um pouco.
7. Faça furos no bolo para o chocolate escorrer pelo meio do bolo.

Fonte: <https://www.tudogostoso.com.br/receita/48426-bolo-de-cenoura-fofinho-com-cobertura-de-chocolate.html> em 25/08/2020

O bolo, da primeira receita que fizemos, foi suficiente para 6 pessoas.

Agora temos 24 pessoas! Como faremos? Podemos usar a mesma quantidade de ovos ou de farinha? Vamos lá aqueça-se! Monte a receita para 24 pessoas no seu caderno de anotações.

Pronto?

- A. Agora vamos às comparações:
- B. Qual era a quantidade de farinha?
- C. E agora? Precisamos de quanto de farinha?
- D. Antes, qual era o tamanho da forma?

- E. E agora? Qual o tamanho da forma usar? Ou quantas formas você usou?
F. Então podemos dizer que as quantidades variaram, certo?
G. Quanto tempo o bolo ficou no forno?

Agora, é um bom momento para conversar sobre as variáveis que aconteceram na preparação para esse bolo maior! Enumere tudo que variou. Quais os nomes do que variou?

- 1 –
- 2 –
- 3 –
- 4 –
- 5 –
- 6 –
- 7 –
- 8 –
- 9 –
- 10 –

Este conteúdo liga você futuramente a:

- **JavaScript** - Variáveis, seleção de ações, cálculos, vetores;
- **SQL** - Armazenamento, tipos de dados armazenados, vetores;
- **PHP e HTML** - Variáveis, seleção de ações, cálculos, vetores.

3.1 O que são variáveis?

Variáveis

Agora, pense em um local que guarda um conteúdo. Observe o local que você está neste exato momento, com certeza, você está perto de muitas variáveis.

Exemplo 1: Imagine uma variável, como uma gaveta "etiquetada" em um armário. Onde cada gaveta é uma variável e podemos guardar várias coisas dentro delas e também separar por camisas, calças e bermudas, ou seja, o tipo de roupa. Dentro do programa, criamos a variável e todas elas têm um tipo, como foi apresentado no vídeo.



Fonte: <https://dicasdeprogramacao.com.br/>

Consolide esse conceito sobre variável!

Uma variável possui três características (atributos): um nome, um tipo de dado e a informação por ela guardada.

Um nome de variável deve começar com uma letra;

Um nome de variável não deve ter nenhum símbolo especial exceto o sublinhar “_”.

Veja os exemplos abaixo:

SALARIO = correto

1ANO = errado

A CASA = errado (contém o caractere espaço em branco)

ASA/COBRA = errado (contém o caractere “/”)

ASA_COBRA = correto

_DESCONTO = errado (não começou com uma letra)

3.2 Tipos de Dados

A nossa atenção agora será para identificar os tipos de dados que são utilizados em algumas linguagens de programação.

1. Numérico: Inteiro e Real

Inteiro: Qualquer número que pode ser contado, números que não tem casas decimais.
Exemplo: 1, 2, 27, -9...

Real: Números com casas decimais - Exemplo: Valor do aluguel R\$600,00

2. Literal - Carácter e Cadeia

Carácter: é todo símbolo que pode ser introduzido pelo teclado do computador. Os principais são Letras, Números e Símbolos.

Cadeia: cadeia de caracteres - Exemplo: Seu nome

3. Lógico

É uma situação ou outra. Ou seja 8 ou 80. Exemplo: Rose é alta? Para esta pergunta só tem dois tipos de resposta: SIM ou NÃO.

Rose foi à praia! Esta frase pode ser VERDADEIRA ou FALSA.

4.0 Operadores Aritméticos I

Utilizando a receita do bolo, avalie a forma que você trabalhou para aumentar o tamanho do bolo?

Você já tem uma receita de bolo para 6 pessoas! Você com um algoritmo de conhecimento, ou seja, um algoritmo de vida conseguiu dobrar a receita, não foi? Mas como você fez isto? Agora, um desafio: E se você precisasse fazer um bolo para 252 pessoas?

Vamos fazer um algoritmo para qualquer quantidade de pessoas.

Pense como faremos isso.

Falaremos agora sobre Operadores Aritméticos.

Este conteúdo liga você futuramente a:

- **JavaScript** - Juntar conteúdos, fazer operações matemáticas, tomar decisões, converter números.
- **SQL** - Juntar conteúdos, fazer operações matemáticas, tomar decisões, disparar gatilhos
- **PHP e HTML** - Juntar conteúdos, fazer operações matemáticas, tomar decisões, converter números.

4.1 Operações básicas

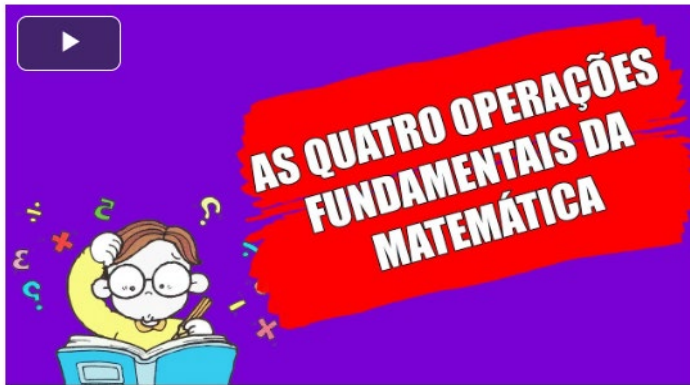
Definições sobre:

Operações básicas (Soma, Subtração, Multiplicação e Divisão)

Operador + (Concatenação) – Juntas dois valores

Na matemática utilizamos as operações básicas que vimos no vídeo (soma, subtração, multiplicação e divisão). Fique atento aos símbolos utilizados.

Muito importante, agora, voltarmos a rever as quatro operações básicas de uma forma simples, mas que serão muito úteis no aprendizado da lógica de programação.



Fonte: <Canal do Youtube PortalZUN>

Acesse: <https://www.youtube.com/watch?v=RheXFO8fGTU>

4.2 Operadores Aritméticos (Conceitos e prática)

Operadores Aritméticos: são contas matemáticas feitas com tipo numérico (um dos tipos de dados que vimos na aula passada). São aquelas operações básicas já aprendidas na escola como a soma, subtração, multiplicação e divisão.

Operador	Descrição
+	Adição
-	Subtração
*	Multiplicação
/	Divisão
%	Módulo

Concatenação: podemos colocar juntos dois pensamentos diferentes, duas ideias.

Ideia 1: Eu vou à praia!

Ideia 2: Vou tomar sorvete

Concatenação: Eu vou à praia, vou tomar sorvete.

5.0 Operadores Aritméticos II

Continuamos com o aprendizado sobre os Operadores Aritméticos.

Este conteúdo liga você futuramente a:

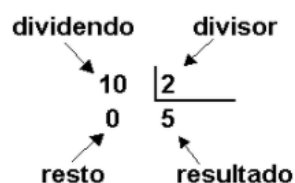
- **JavaScript** - Juntar conteúdos, fazer operações matemáticas, tomar decisões, converter números

- **SQL** - Juntar conteúdos, fazer operações matemáticas, tomar decisões, disparar gatilhos
- **PHP e HTML** - Juntar conteúdos, fazer operações matemáticas, tomar decisões, converter números

Operações básicas

Operador	Descrição
+	Adição
-	Subtração
*	Multiplicação
/	Divisão
%	Módulo

Operador de Módulo: A operação módulo encontra o resto da divisão de um número por outro.



Fonte: <https://www.estudamos.com.br/divisao/>

6.0 Desvios Condicionais I

O que são Estruturas Condicionais?

As Estruturas Condicionais ou Instruções Condicionais são utilizadas para executar determinada ação, se uma condição previamente definida for atendida.

Por exemplo:

Se o interruptor de acender a lâmpada estiver ligado
a lâmpada deve acender.

Se não
a lâmpada deve ficar apagada.

Por mais simples que possa parecer o exemplo acima, demonstra como uma das estruturas condicionais funciona.

Serão apresentadas nos próximos tópicos.

6.1 Conceitos: Tarefa e Condição

Tarefa: qualquer trabalho que você tem para fazer, por exemplo: os exercícios aqui do curso de Programação Web; arrumar casa; pagar uma conta;

Condição: é o importante para a realização, ou não, de uma tarefa. Exemplo: vou à praia se não chover. A minha condição **é se não chover**, ou seja, se chover eu não vou à praia.

Você se lembra do exercício da Calculadora da última aula? Nele pedia para entrar com dois números inteiros e o usuário escolhia qual das operações básicas ele queria fazer. Ou seja, soma, subtração, multiplicação ou divisão.

Nesta aula, iremos fazer uma para pensar em:

no caso um desses números tiverem que ser maior que o outro, ou menor, ou maior igual, menor igual.

Vamos lá! O que fazer?

6.2 Comando IF (SE)

A instrução **IF** executa um código definido por você, se a condição que você definiu for atendida.

Exemplo:

```
let nome == "Lucas";  
if(nome == "Lucas") {  
  console.log ("Esse é o nome do Professor");  
}
```

Nesse exemplo, a porção de código “Esse é o nome do Professor” só vai ser executada se a variável nome possuir o valor Lucas. Se ela não contiver esse valor, o código vai seguir pelas próximas linhas sem executar essa instrução: “Esse é o nome do Professor”;

Observações:

-Entende-se por bloco de código todo o código entre chaves {}.

-As chaves são opcionais quando o bloco de código tiver apenas uma linha

Pode parecer um pouco confuso, mas na medida em que vamos construindo nossos programas, você verá que é simples.

6.3 Comando ELSE (SE NÃO)

A instrução IF e ELSE funciona assim: primeiro, a condição estabelecida no comando IF é testada. Se ela for atendida o bloco de código do IF é executado e o bloco de código do ELSE é ignorado. Se a condição que está no IF não for atendida, então, o bloco de código do IF não é executado e o bloco de código que está no ELSE será executado.

Exemplo:

```
let hora = 15;
if(hora < 18){
  console.log("Tenha um ótimo dia.");
} else {
  console.log("Tenha uma ótima noite");
}
```

6.4 Comando ELSE IF (SE NÃO SE)

A instrução if, else if, else vai primeiramente testar o IF. Se a condição do IF for atendida, o bloco de código do IF vai ser executado e os blocos de códigos do ELSE IF e do ELSE vão ser ignorados. Se ela não for atendida, ele vai testar o ELSE IF e se o ELSE IF for atendido o bloco de código do ELSE IF é executado e os outros blocos de códigos são ignorados. Se a condição do ELSE IF não for atendida, então, o bloco de código do ELSE é que vai ser executado.

Múltiplas condicionais if ... else podem ser utilizadas quando necessário.

```
if (hora < 10) {
  console.log ("Bom dia!");
} else if (hora < 18) {
  console.log ("Boa tarde!");
} else {
  console.log ("Boa noite!");
}
```

6.5 Comando Switch

A instrução switch é utilizada para selecionar apenas um dos blocos de código a serem executados, dentre todos os que você precisa definir.

Switch (opções) {

Case 1:

```
  console.log ("Você escolheu a primeira opção.");
}
```

```

    Break;
Case 2:
    console.log ("Você escolheu a segunda opção.");
    Break;
Case 3:
    console.log ("Você escolheu a terceira opção.");
    Break;
Default:
    console.log ("Opção inválida.");
}

```

Ela funciona da seguinte forma: primeiro, temos uma expressão (geralmente uma variável), no nosso caso a variável opções, que é avaliada uma vez.

O valor que a variável opções recebe é, então, comparado com os valores para cada caso (case) na estrutura do código. Se esse valor for correspondente a algum valor definido em caso (case), o bloco de código associado a esse caso (case) vai ser executado.

O Break é utilizado para evitar que o código seja executado no próximo caso, faz com que apenas o caso que foi escolhido seja executado e nada mais que venham depois dele.

Mais à frente, será feito mais abordagem sobre a instrução break (pausa).

A instrução default é simples. Ela está ali para ser executada, caso as opções que pré-determinamos, ou seja, 1, 2 ou 3 não sejam escolhidas pelo usuário. Então, se o usuário digitar: 4, 5, 10 ou uma letra (ou qualquer coisa que não seja 1,2 ou 3) é a opção default que será executada.

Este conteúdo liga você futuramente a:

- **JavaScript** - Tomadas de decisão, desvios de programa, controle de fluxo, condicional IF, ELSE, ELSE IF
- **SQL** - Tomadas de decisão, desvios de programa, controle de fluxo, condicional IF, ELSE, ELSE IF, THEN, JOINS
- **PHP e HTML** - Tomadas de decisão, desvios de programa, controle de fluxo, condicional IF, ELSE, ELSE IF

7.0 Operadores Relacionais

Definições sobre:

- Operador Maior que
- Operador Menor que
- Operador Maior ou igual
- Operador Menor ou igual

Este conteúdo liga você futuramente a:

- **JavaScript** - comparações, tomadas de decisão, fluxo de programa
- **SQL** - comparações, tomadas de decisão, fluxo de programa, classificação de dados, exibição de dados
- **PHP e HTML** - Comparações, tomadas de decisão, fluxo de programa, classificação de dados, exibição de dados

7.1 Operadores Relacionais – Símbolos

Operadores Relacionais (de comparação)

Operador	Descrição	Exemplo	Resultado
>	Maior que	5 < 10.0	true
<	Menor que		
>=	Maior ou igual	42 >= "hello"	false
<=	Menor ou igual		

8.0 Estruturas de Repetição

Estruturas de repetição ou Laços de repetição oferecem uma forma de uma ação ser executada repetidas vezes.

Às vezes, é necessário, que parte de um código seja repetida diversas vezes, para que uma parte do seu programa funcione da forma esperada.



Por exemplo: Estou construindo um jogo de computador onde quero que enquanto o usuário do jogo esteja apertando tal botão o personagem do jogo antepara frente sem parar.

Existem várias formas diferentes de utilizar as estruturas de repetição.

Todas elas são bem parecidas mas oferecem formas diferentes de determinar quando a repetição vai começar ou terminar.

Em cada situação você pode utilizar uma estrutura de repetição determinada para resolver a lógica do seu programa da melhor forma possível.

Este conteúdo liga você futuramente a:

- **JavaScript** - fluxo finito/infinito de um programa, WHILE, DO WHILE
- **SQL** - gatilhos, WHILE, DO WHILE, THEN
- **PHP e HTML** - Fluxo finito/infinito, WHILE, DO WHILE

8.1 O Comando While (Enquanto)

A Estrutura de repetição (Loop) While (Enquanto) vai executar um bloco de código, enquanto uma condição pré-estabelecida estiver sendo atendida.

Por exemplo: se você quer que todos os números de 0 a 9 sejam impressos na tela, enquanto o valor de determinada variável for menor que 10.

```
a = 0;

while(a < 10) {
  console.log(a);
  A++;
}
```

8.2 Comando Do While (Faça Enquanto)

O Comando Do... While (Faça ... enquanto), faz um loop em um bloco de código uma vez antes de passar pelo While e, em seguida, repete o loop enquanto condição especificada por você para parar o Loop não for atendida.

A grande diferença de utilizar Do... While, ao invés de utilizar apenas While, é que o Loop será executado ao menos uma vez. No caso da utilização apenas do comando While a condição de parada pode ser atendida antes de entrar no Loop e ele não ser executado nem ao menos uma vez.

```
a = 1;

do {
  console.log(a);
  a++;
} while (a < 10);
```

8.3 Comando For

O Comando For (Para) é usado quando você sabe, com antecedência, quantas vezes você quer repetir determinado código.

```
for (a = 0; a <= 10; a++) {
  console.log(a);
}
```


8.4 Comando Break e Continue

Como havíamos dito no tópico sobre a instrução Switch, o comando Break serve para interromper um bloco de código e ignorar os outros blocos de código e também serve para sair de um Loop, veja a seguir.

Este exemplo sai do loop quando a variável a for igual a 8:

```
for (a = 0; a < 10; a++) {  
  if (a == 8) {  
    break;  
  }  
  console.log(a);  
}
```

O comando Continue (Continuar) interrompe uma iteração (no loop), se uma condição especificada for atendida e continua com a próxima iteração no loop.

Este exemplo pula o valor de 8:

```
for (a = 0; a < 10; a++) {  
  if (a == 8) {  
    continue;  
  }  
  console.log(a);  
}  
?>
```

8.5 Comentários

//comentários até ao fim da linha

/*

Linhas de comentários

Linhas de comentários

*/

9.0 Operadores Lógicos

Definições sobre:

- Premissas (verdadeiro/falso)
- Operação Lógica **e** (conjunção)
- Operação Lógica **ou** (disjunção)
- Operação Lógica **não** (negação)

Este conteúdo liga você futuramente a:

- **JavaScript** - E, OU, NÃO, Condições lógicas (Booleanas) True, False.
- **SQL** - E, OU, NÃO, Condições lógicas (Booleanas) True, False.
- **PHP** - E, OU, NÃO, Condições lógicas (Booleanas) True, False.

9.1 Premissas (Proposição)

Premissa significa a **proposição**, o conteúdo, as informações essenciais que servem de base para um raciocínio, para um estudo que levará a uma conclusão.

Exemplo: Então, se eu afirmar “Belo Horizonte é maior que Ouro Preto”, estarei diante de uma **proposição** cujo **valor lógico** é verdadeiro.

Quando falamos em valor lógico fica claro que estamos nos referindo a um dos dois possíveis valores: verdadeiro (V) ou falso (F).

E se alguém disser: “Feliz Natal”, será que isso é uma proposição verdadeira ou falsa?

Nenhuma, pois não se trata de uma sentença para a qual se possa atribuir um valor lógico.

Ou seja, eu não posso falar que a sentença "Feliz Natal" é verdadeiro ou falso.

Concluimos, pois, que:

- Sentenças exclamativas: “Caramba!” ; “Feliz aniversário!”
- Sentenças interrogativas: “como é o seu nome?” ; “o jogo foi de quanto?”
- Sentenças imperativas: “Estude mais.” ; “Leia aquele livro”.
- Sentenças declarativas: “Belo Horizonte é maior que Ouro Preto”,

Vamos trabalhar aqui com as sentenças declarativas:

Normalmente, as proposições são representadas por letras minúsculas (p, q, r, s, etc).

São outros exemplos de **proposições**:

p: Maria é médica.

q: $10 > 8$

r: João foi ao cinema ontem à noite

Proposições podem ser ditas simples ou compostas.

Serão proposições **simples** aquelas que vêm sozinhas, desacompanhadas de outras proposições.

- Toda mulher é mortal
- O novo atleta é japonês

Se duas (ou mais) proposições vêm conectadas entre si, formando uma só sentença, estaremos diante de uma proposição **composta**.

Maria é médica **e** Pedro é dentista.

Meire vai ao clube **ou** Paulo vai ao centro.

Ou Luís é paulista, **ou** é carioca.

Se chover amanhã de manhã, **então** não irei à praia.

Comprarei uma casa **se e somente se** eu ganhar na loteria.

Conectivos Lógicos são expressões que servem para unir duas ou mais proposições.

10.0 Desvios Condicionais II

Definições sobre:

- Comando **escolha-caso**

Este conteúdo liga você futuramente a:

- **JavaScript** - SWITCH/CASE/BREAK
- **SQL** - SWITCH/CASE/BREAK
- **PHP** - SWITCH/CASE/BREAK
- **HTML/CSS** – Seletores

10.1 Conceitos: Escolha, Switch/case

Escolha: quando você vai em uma pizzeria tem um cardápio e nele os sabores das pizzas. O valor da pizza se altera de acordo com o sabor, por exemplo: Se você escolher a pizza no sabor à moda ela custa R\$30,00, caso escolha o sabor calabresa, ela custa a R\$35,00, caso escolha o sabor catupiry, ela custa R\$40,00.

O comando escolha-caso, nos permite fazer essas escolhas dentro do programa. No próximo Módulo (Java Script), você irá aprender a usar o mesmo comando escolha-caso mais em JavaScript o comando será conhecido como Switch-Case.

O **switch/case** é uma estrutura condicional que está presente em várias linguagens de programação. Vai estar presente, também, no módulo de PHP. Anote no seu caderno.

11.0 Vetores

Definições sobre:

- Vetor
- Posição
- Declaração de vetor
- Leitura de um vetor

Este conteúdo liga você futuramente a:

- **Javascript** - ARRAYS,OBJETOS
- **SQL** - Armazenamento de dados, conceito básico de sql
- **PHP** - ARRAYS,OBJETOS,CLASSES
- **HTML/CSS** - ESTILOS CLASSES

Vetor: é uma variável que armazena várias variáveis do mesmo tipo. Exemplo: você precisa criar um algoritmo que lê nomes e idades dos 27 participantes da sua festa de aniversário. Pense em quantas variáveis você teria que criar, o código ficaria enorme, por isso, para estes problemas contamos com os vetores. Neles conseguimos armazenar muitas informações do mesmo tipo, exemplo: quando você vai ao cinema tem vários lugares (posições), é como se fosse nosso vetor cinema com 50 lugares. Vetor cinema[50]. Cada posição desta é de uma pessoa que compra o ingresso para ver o filme e pode ser usada por outras pessoas, se uma pessoa muda de lugar ou sai da sala de cinema e vai para um outro vetor (sala do cinema).

Conseguimos acessar os lugares (posições) deste vetor cinema. Eu comprei a posição vetor cinema[2].

12.0 Matrizes

Definições sobre:

- Matriz
- Declaração de matriz
- Leitura de matriz

Este conteúdo liga você futuramente a:

- **JavaScript** - ARRAYS,OBJETOS
- **SQL** - DATABASE
- **PHP** - ARRAYS,OBJETOS,CLASSES

Matriz: É um vetor de vetores.

13.0 Funções

Definições sobre:

- Procedimentos (Funções)

Este conteúdo liga você futuramente a:

- **Javascript** - Function
- **SQL** - Function

- **PHP** – Function

Funções: são instruções (comandos) que executam alguma tarefa e que têm um nome. Divide um problema grande em problemas menores que serão resolvidos um a um.

14.0 Teste de mesa e Bibliotecas

Definições sobre:

- Teste de mesa
- Bibliotecas

Este conteúdo liga você futuramente a:

- **Javascript** - Libs, JQUERY, ANGULAR
- **PHP** - Bibliotecas de Gráficos, Cálculos, Decisões
- **HTML** - Folhas de estilos, Bootstrap, Fonts Awesome

Teste de mesa: antes de você testar seu programa (você pode testar) chamamos de teste de mesa, ou seja, você vai pegar um caderno e vai desenhar como seu programa vai responder aos seus comandos.

Bibliotecas: são funções especiais prontas que já são fornecidas pelo Portugal Studio.

15.0 Arquivos

Definições sobre:

- Arquivos

Chegamos ao fim do nosso módulo de Introdução à Lógica de Programação. Dentro desta estrutura, que foi apresentada no decorrer deste módulo, o Portugal conta com uma pasta onde é colocado todos os seus programas construídos. Toda vez que você for utilizar o Portugal você pode salvar seus programas nesta pasta.

REFERENCIAL BIBLIOGRÁFICO:

Vídeos - Canal do Youtube HM Programming -
<https://www.youtube.com/channel/UCsXyoSkpyZB741fkBKeEovQ>

Rede e-Tec Brasil - <http://portal.mec.gov.br/rede-e-tec-brasil>

Instituto de Educação, Ciência e Tecnologia do Rio Grande do Norte -
<https://portal.ifrn.edu.br/>

