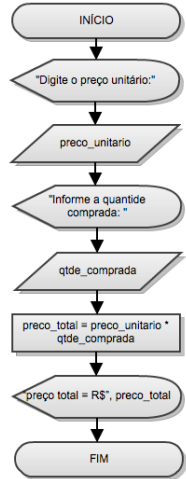



ATIVIDADES

01. Faça um algoritmo para ler um preço unitário, uma quantidade comprada e calcular e exibir o preço total a pagar.

Pseudocódigo	Fluxograma
<pre> inicio real: precoUnit, precoTot; inteiro: qtd; escreva("Digite o preço: "); leia(precoUnit); escreva("Digite a quantidade comprada:"); leia(qtd); precoTot <- precoUnit * qtd; escreva("preço total = R\$", precoTot); fim. </pre>	 <pre> graph TD INICIO([INÍCIO]) --> Input1[/Digite o preço unitário:/] Input1 --> Data1[/preco_unitario/] Data1 --> Input2[/Informe a quantidade comprada:/] Input2 --> Data2[/qtde_comprada/] Data2 --> Process[preco_total = preco_unitario * qtde_comprada] Process --> Output[/preço total = R\$, preco_total/] Output --> FIM([FIM]) </pre>

02. Faça um algoritmo que leia o nome do usuário e exiba uma frase de saudação, desejando um bom dia de trabalho para o mesmo. A resposta deve ter o formato deste exemplo: "Olá Joao, tenha um ótimo dia de trabalho!".

Pseudocódigo	Fluxograma
<pre> // Mostrar frase de saudação inicio caracter: usuario; escreva("Digite seu nome: "); leia(usuario); escreva("Olá ", usuario, " tenha um otimo dia de trabalho!"); fim </pre>	 <pre> graph TD INICIO([INÍCIO]) --> Input[/Digite seu nome: /] Input --> Data[/usuário/] Data --> Output[/Olá , usuário, " tenha um otimo dia de trabalho!/] Output --> FIM([FIM]) </pre>

03. Faça um algoritmo que leia dois números inteiros A e B e exiba o resultado da subtração de A por B. A resposta deve ter o seguinte formato de saída: "6 - 2 = 4" (sendo 6 e 2 exemplos de valores fornecidos pelo usuário).

Pseudocódigo	Fluxograma
<pre>// Cálculo subtração entre dois números inteiros inicio inteiro: A, B, subtracao; escreva("Digite dois numeros inteiros: "); leia(A, B); subtracao <- A - B; escreva(A, " - ", B, " = ", subtracao); fim</pre>	<pre> graph TD INICIO([INICIO]) --> PROMPT1[/Digite dois numeros inteiros:/] PROMPT1 --> LEIA[/X, Y/] LEIA --> PROCESS[subtração = A - B] PROCESS --> PROMPT2[/A, " - ", B, " = ", subtração/] PROMPT2 --> FIM([FIM]) </pre>

04. Faça um algoritmo que leia o nome do usuário e o ano de seu nascimento e exiba quantos anos ele terá ao final do ano 2021 (considere que o usuário tenha feito aniversário no ano corrente). A resposta deve ter o formato deste exemplo: "Rodrigues, em 2021 você estará com 45 anos".

Pseudocódigo	Fluxograma
<pre>// Cálculo a idade em 2020 inicio caracter: usuario; inteiro: idade, idade_futura; escreva("Digite seu nome: "); leia(usuario); escreva("Digite o ano de seu nascimento: "); leia(ano); idade_futura <- 2021 - ano; escreva(usuario, " em 2020 você estará com ", idade_futura, " anos"); fim</pre>	<pre> graph TD INICIO([INICIO]) --> PROMPT1[/Digite seu nome:/] PROMPT1 --> LEIA1[/usuario/] LEIA1 --> PROMPT2[/Digite sua idade:/] PROMPT2 --> LEIA2[/idade/] LEIA2 --> PROCESS[idade_futura = 2020 - idade] PROCESS --> PROMPT3[/usuario, " em 2020 você estará com ", idade_futura, " anos/] PROMPT3 --> FIM([FIM]) </pre>

05. Faça um algoritmo que leia um número qualquer e exiba o seu cubo.

Pseudocódigo	Fluxograma
<pre>// Cálculo do cubo de um número inicio real: numero; escreva("Digite um numero: "); leia(numero); escreva("Cubo do número ", numero, " é ", numero^3); fim</pre>	<pre> graph TD INICIO([INICIO]) --> Prompt1[/Digite um número: /] Prompt1 --> Input[/número/] Input --> Process[cubo = numero^3] Process --> Prompt2[/Cubo do número ", numero, " é ", cubo/] Prompt2 --> FIM([FIM]) </pre>

06. Uma empresa vai dar um aumento de salário ao seu funcionário. Faça um algoritmo para ler o salário do funcionário e o percentual de reajuste. Calcular e escrever o valor do novo salário.

Pseudocódigo	Fluxograma
<pre>// Cálculo do salário reajustado de um funcionário inicio real: salario, reajuste, novo_salario; escreva("Digite o salario do funcionario: R\$ "); leia(salario); escreva("Digite o percentual de reajuste:"); leia(reajuste); novo_salario <- salario + salario*reajuste/100; escreva("Salário reajustado: R\$ ", novo_salario); fim</pre>	<pre> graph TD INICIO([INICIO]) --> Prompt1[/Digite o salario do funcionario: R\$ /] Prompt1 --> Input1[/salário/] Input1 --> Prompt2[/Digite o percentual de reajuste:/] Prompt2 --> Input2[/reajuste/] Input2 --> Process[novo_salário = salario + salario*reajuste/100] Process --> Prompt3[/Salário reajustado: R\$ ", novo_salário/] Prompt3 --> FIM([FIM]) </pre>

07. Faça um algoritmo que leia o número total de prestações de um consórcio, o total de prestações pagas e o valor atual da prestação. O algoritmo deve calcular e apresentar o saldo devedor atual e a quantidade de prestações sem pagar.

Pseudocódigo	Fluxograma
<pre> // Cálculo do saldo devedor atual e a quantidade de prestações a pagar em um consórcio inteiro real: valor_prestacao, saldo_devedor; inteiro: total_prestacoes, pg_prestacoes, apg_prestacoes; //pg_prestacoes igual a prestações pagas ; apg_prestacoes igual a prestações a pagar escreva("Digite o total de prestações do consórcio: "); leia(total_prestacoes); escreva("Digite o total de prestações pagas: "); leia(pg_prestacoes); escreva("Digite o valor atual da prestação: R\$"); leia(valor_prestacao); apg_prestacoes <- total_prestacoes - pg_prestacoes; saldo_devedor <- apg_prestacoes * valor_prestacao; escreva("Saldo devedor atual: R\$ ", saldo_devedor); escreva("Prestações a pagar: R\$ ", apg_prestacoes); fim </pre>	<pre> graph TD INICIO([INICIO]) --> Input1[/Digite o total de prestações do consórcio: /] Input1 --> Process1[/total_prestações /] Process1 --> Input2[/Digite o total de prestações pagas: /] Input2 --> Process2[/pg_prestações /] Process2 --> Input3[/Digite o valor atual da prestação: R\$/] Input3 --> Process3[/valor_prestação /] Process3 --> Process4[apg_prestações = total_prestações - pg_prestações] Process4 --> Process5[saldo_devedor = apg_prestações * valor_prestação] Process5 --> Output1[/Saldo devedor atual: R\$ /] Output1 --> Output2[/Prestações a pagar: R\$ /] Output2 --> FIM([FIM]) </pre>

08. Fornecidas as seguintes entradas: quilometragem inicial, quilometragem final, litros consumidos, preço do litro de combustível. Faça um algoritmo que exiba o seguinte relatório: distância percorrida, valor total gasto e o consumo do carro (quantos quilômetros percorre com 1 litro de combustível).

Pseudocódigo	Fluxograma
<pre> // Cálculo do saldo devedor atual e a quantidade de prestações a pagar em um consórcio início real: preco_litro, valor_gasto, consumo_carro; inteiro: km_inicial, km_final, litros_consumidos, distancia_percorrida; //km igual a quilometragem escreva("Entre com a quilometragem inicial: "); leia(km_inicial); escreva("Entre com a quilometragem final: "); leia(km_final); escreva("Digite a quantidade de litros consumidos: "); leia(litros_consumidos); escreva("Entre com o preço do combustível: R\$"); leia(preco_litro); distancia_percorrida <- km_final - km_inicial; valor_gasto <- preco_litro * litros_consumidos; consumo_carro <- distancia_percorrida / litros_consumidos; escreva("RELATORIO DE VIAGEM"); escreva("Distância percorrida: ", distancia_percorrida, "km"); escreva("Valor total gasto na viagem: R\$", valor_gasto); escreva("Consumo do carro nesta viagem: ", consumo_carro, "km/l"); fim </pre>	<pre> graph TD INICIO([INICIO]) --> Entrada1[/Entre com a quilometragem inicial: /] Entrada1 --> km_inicial[/km_inicial/] km_inicial --> Entrada2[/Entre com a quilometragem final: /] Entrada2 --> km_final[/km_final/] km_final --> Entrada3[/Entre com o preço do combustível: R\$/] Entrada3 --> preco_litro[/preço_litro/] preco_litro --> CalculaDistancia[distância_percorrida = km_final - km_inicial] CalculaDistancia --> CalculaValorGasto[valor_gasto = preco_litro * litros_consumidos] CalculaValorGasto --> CalculaConsumo[consumo_carro = distância_percorrida / litros_consumidos] CalculaConsumo --> Relatorio[/RELATÓRIO DE VIAGEM/] Relatorio --> Saída1[/Distância percorrida: , distância_percorrida, "km"/] Saída1 --> Saída2[/Valor total gasto na viagem: R\$, valor_gasto/] Saída2 --> Saída3[/Consumo do carro nesta viagem: , consumo_carro, "km/l"/] Saída3 --> FIM([FIM]) </pre>

09. Um motorista de táxi deseja calcular o rendimento de seu carro na praça. Sabendo-se que o preço do combustível é de R\$2.97, faça um algoritmo para ler: a marcação do odômetro (Km) no início do dia, a marcação (Km) no final do dia, a quantidade de litros de combustível gasto e o valor total (R\$) recebido dos passageiros. Calcular e escrever: a média do consumo em km/l e o lucro (líquido) do dia.

Pseudocódigo	Fluxograma
<pre> // Cálculo do saldo devedor atual e a quantidade de prestações a pagar em um consórcio inicio real: valor_recebido, lucro_liquido, consumo_carro; inteiro: km_inicial, km_final, litros_consumidos; //km igual a quilometragem escreva("Entre com a quilometragem inicial: "); leia(km_inicial); escreva ("Entre com a quilometragem final: "); leia(km_final); escreva ("Digite a quantidade de litros consumidos: "); leia(litros_consumidos); escreva ("Entre com o valor recebido dos passageiros: R\$"); leia(valor_recebido); consumo_carro <- (km_final - km_inicial) / litros_consumidos; lucro_liquido <- valor_recebido - 2.68 * litros_consumidos; escreva ("TAXI DO WILL"); escreva ("Media de consumo do carro: ", consumo_carro, "km/l"); escreva ("Lucro líquido do dia: R\$ ", lucro_liquido); fim </pre>	<pre> graph TD INICIO([INÍCIO]) --> EntradaKmInicial[/Entre com a quilometragem inicial: /] EntradaKmInicial --> SaídaKmInicial[/km_inicial/] SaídaKmInicial --> EntradaKmFinal[/Entre com a quilometragem final: /] EntradaKmFinal --> SaídaKmFinal[/km_final/] SaídaKmFinal --> EntradaLitrosConsumidos[/Digite a quantidade de litros consumidos: /] EntradaLitrosConsumidos --> SaídaLitrosConsumidos[/litros_consumidos/] SaídaLitrosConsumidos --> EntradaValorRecebido[/Entre com o valor recebido dos passageiros: R\$/] EntradaValorRecebido --> SaídaValorRecebido[/valor_recebido/] SaídaValorRecebido --> ProcessamentoConsumo[consumo_carro = (km_final - km_inicial) / litros_consumidos] ProcessamentoConsumo --> ProcessamentoLucro[lucro_liquido = valor_recebido - 2.68 * litros_consumidos] ProcessamentoLucro --> SaidaTudo[/TAXI DO WILL Media de consumo do carro: , consumo_carro, km/l Lucro líquido do dia: R\$, lucro_liquido/] SaidaTudo --> FIM([FIM]) </pre>

**Todos os algoritmos deverão ser feitos pelas representações em Pseudocódigo e Fluxograma.*

Bom divertimento!