AVALIAÇÃO (LÓGICA DE PROGRAMAÇÃO)

1. **Você está criando um programa simples em Visual G para registrar informações de um usuário. O programa precisa armazenar o nome do usuário, sua idade e seu saldo bancário.**
   1. **Quais tipos de dados você usaria para armazenar o nome do usuário, a idade e o saldo bancário?**inteiro: idade

real: saldoBancario

caractere: user **Justifique sua escolha.**

O nome do usuário é uma sequência de caracteres.

A idade do usuário é um número inteiro, já que não faz sentido ter uma fração de idade.

O saldo bancário é um valor numérico que pode incluir frações (centavos).

* 1. **Escreva um código simples em Visual G para declarar e inicializar variáveis para armazenar o nome do usuário, a idade e o saldo bancário. Inclua também uma instrução para exibir essas informações na tela.**

Algoritmo "semnome"

Var

idade: inteiro

saldoBancario: real

user: caractere

Inicio

idade <- 18

saldoBancario <- 1.500,00

user <- "ev7@"

escreva("seu nome é:", user)

escreva("sua idade é:", idade)

escreva("seu saldo é de:", saldoBancario)

Fimalgoritmo

1. **Você está criando um programa em Visual G para calcular o valor total de uma compra em um supermercado. O programa deve ler o preço de um item e a quantidade comprada e, em seguida, calcular o total da compra e o valor com desconto de 10%.**
   1. **Quais operadores aritméticos você usaria para calcular o valor total da compra e o valor com desconto? Descreva o uso de cada operador no contexto desta tarefa.**

Multiplicação, Divisão e Subtração

Multiplicação (\*):

Uso: Para calcular o valor total da compra, você multiplicaria o preço do item pela quantidade comprada.

Exemplo: total <- preco \* quantidade

Divisão (/):

Uso: Para calcular o valor do desconto, você pode dividir o valor total por 10 para encontrar 10% do total.

Exemplo: desconto <- total / 10

Subtração (-):

Uso: Para calcular o valor final com desconto, você subtrairia o valor do desconto do valor total da compra.

Exemplo: total\_com\_desconto <- total – desconto

* 1. **Escreva um código simples em Visual G que leia o preço do item e a quantidade comprada, calcule o valor total da compra, aplique um desconto de 10% e exiba o valor final da compra.**

Algoritmo "semnome"

Var

preco: real

quantidade: inteiro

total: real

desconto: real

valorComDesconto: real

Inicio

escreva("Digite o preço do item: ")

leia(preco)

escreva("Digite a quantidade comprada: ")

leia(quantidade)

total <- preco \* quantidade

desconto <- total \* 0.10

valorComDesconto <- total - desconto

escreva("Valor total da compra: R$ ", total)

escreva("Valor com desconto de 10%: R$ ", valorComDesconto)

Fimalgoritmo

1. **Você está desenvolvendo um programa em Visual G para verificar se um aluno pode participar de um curso específico. Para participar, o aluno deve atender aos seguintes critérios:**
2. **O aluno deve ter pelo menos 18 anos.**
3. **O aluno deve ter um desempenho mínimo de 7.0 em uma prova de qualificação.**

O programa deve ler a idade do aluno e a nota da prova de qualificação e então verificar se o aluno atende a ambos os critérios para participar do curso.

* 1. **Quais operadores lógicos você usaria para verificar se o aluno atende a ambos os critérios? Descreva o uso de cada operador no contexto desta tarefa.** Se, senao   
       
     Se verificou idade e nota, senao informa se atende ou não aos critérios.
  2. **Escreva um código simples em Visual G que leia a idade do aluno e a nota da prova de qualificação, e verifique se o aluno pode participar do curso. O programa deve exibir uma mensagem indicando se o aluno pode ou não participar.**

Algoritmo "semnome"

Var

idade: inteiro

nota: real

Inicio

escreva("Digite a sua idade: ")

leia(idade)

escreva("Digite a sua nota: ")

leia(nota)

se idade < 18 entao

escreva("Você não atende aos critérios do curso.")

senao

se nota < 7.0 entao

escreva("Você não atende aos critérios do curso.")

senao

escreva("Você atende aos critérios do curso.")

fimse

fimse

Fimalgoritmo

1. **Você está criando um programa em Visual G para somar dois números fornecidos pelo usuário e exibir o resultado.**
   1. **Descreva os passos do algoritmo para somar dois números e exibir o resultado.**

1º Declaramos as váriaveis.

2º Solicitamos os números.

3º Somamos os números.

4º Mostramos o resultado.

* 1. **Escreva um código simples em Visual G que leia dois números do usuário, calcule a soma e exiba o resultado.**

Algoritmo "semnome"

Var

numeroUm: real

numeroDois: real

soma: real

Inicio

escreva("Digite o primeiro número: ")

leia(numeroUm)

escreva("Digite o segundo número: ")

leia(numeroDois)

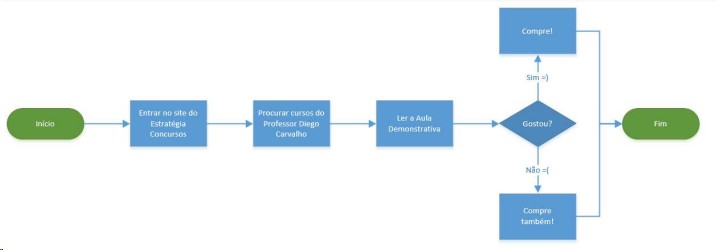
soma <- numeroUm + numeroDois

escreva(" A soma dos dois número é:",soma)

Fimalgoritmo

* 1. **O que é algoritmo?**

É uma sequência finita de instruções bem definidas que, quando seguidas passo a passo, leva à solução de um problema específico,ou seja, é um conjunto de regras e passos lógicos que devem ser seguidos para realizar uma tarefa.



“O fluxograma é uma das formas de se representarem as instruções de um programa, utilizando-se de algunscomandos genéricos ou primitivos para a interpretação do algoritmo.” (Certo ou Errado) - (Justifique sua resposta)

1. **Relacione os itens**
2. **Constantes**
3. **Variáveis**
4. **Atribuições**

(3) É o processo de associar um valor.

(2) São espaços de armazenamento na memória do computador que podem conter diferentes valores durante a execução de um programa.

(1) São valores fixos que não podem ser alterados durante a execução do programa.

1. **Você está desenvolvendo um programa em Visual G para verificar se um número inserido pelo usuário está dentro de um intervalo específico. O intervalo é de 10 a 20, inclusivo. O programa deve ler um número e verificar se ele está dentro desse intervalo.**
   1. **Quais operadores relacionais você usaria para verificar se o número está dentro do intervalo de 10 a 20?** Operadores: >, < e =
   2. **Escreva um código simples em Visual G que leia um número do usuário e verifique se ele está dentro do intervalo de 10 a 20. O programa deve exibir uma mensagem indicando se o número está dentro do intervalo ou não.**

Algoritmo "semnome"

Var

numero: inteiro

Inicio

escreva("Digite um número: ")

leia(numero)

se (numero >=10) e (numero <=20) entao

escreva(" O número está dentro do intervalo.")

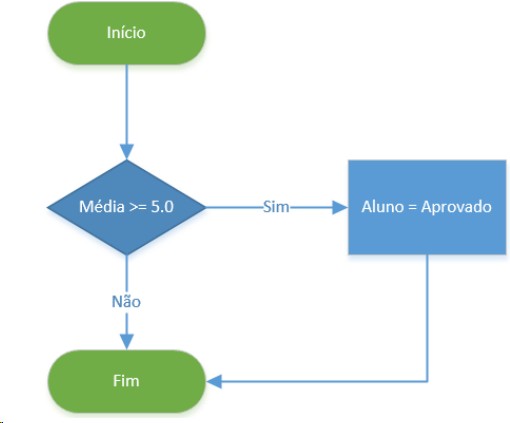
senao

escreva("O número não está dentro do intervalo.")

fimse

Fimalgoritmo

1. **De acordo com fluxograma abaixo responda às seguintes perguntas:**



Qual a estrutura que temos? Estrutura Condicional (Seleção)

O que o fluxograma está avaliando? A média do aluno.

Como é chamada a estrutura de decisão (seleção) quando ela possui somente o bloco "Se Então"? Estrutura de decisão simples.

1. **Em relação às funções e procedimentos, o que seria uma rotina, sub-rotina e co-rotina? Cite as vantagens e desvantagens. (Exemplifique)**

Rotina**:** Uma rotina é um bloco de código que executa uma tarefa específica dentro de um programa. Pode ser chamada várias vezes ao longo do programa.

Vantagens: Consistência, Produtividade, Qualidade, Reutilização de código, Facilidade de aprendizado, Melhoria da colaboração.  
Desvantagens: Rigidez, Resistência a mudanças, Sobrecarga de padrões, Perda de foco, Adaptação a novas tecnologias.

Ex: algoritmo "RotinaExemplo"

inicio

escreval("Esta é uma rotina em Visualg.")

fimprocedimento

Sub-rotina: Uma sub-rotina é uma rotina que é chamada por outra rotina. Ela é usada para dividir um programa em partes menores e mais gerenciáveis. Exemplos incluem funções e procedimentos em várias linguagens de programação.

Vantagens: Reusabilidade, legibilidade e modularidade. Código pode ser reutilizado, fácil leitura, permite dividir o programa em módulos menores e independentes.

Desvantagens: Sobrecarga de chamada e complexidade

Ex:

algoritmo "calcularSoma"

var

inteiro: x , y

resultado: inteiro

inicio

resultado <- x + y

retorne resultado

fimalgoritmo

Co-Rotina: Uma co-rotina é semelhante a uma sub-rotina, mas permite a suspensão e retomada da execução em pontos específicos. Isso é útil para tarefas que precisam ser pausadas e retomadas, como a geração de sequências ou a manipulação de fluxos de dados.

Vantagens: Eficiencia e flexibilidade: Execução de tarefas eficientes. Pode ser pausada e retomada.

Desvantagens: Complexidade e Depuração: Pode ser dificil, o entedimento. A depuração pode ser complicada, devida a natureza não-linear.

Ex: algoritmo "CoRotina"

var

inteiro : a, b, temp

procedimento inicializaFibonacci()

inicio

a := 0

b := 1

fimprocedimento

procedimento proximoFibonacci()

inicio

temp := a

a := b

b := temp + b

escreval(a)

fimprocedimento

inicio

inicializaFibonacci()

// Chama a co-rotina simulada várias vezes

proximoFibonacci() // 1

proximoFibonacci() // 1

proximoFibonacci() // 2

proximoFibonacci() // 3

proximoFibonacci() // 5

fimalgoritmo

1. **Você está criando um programa em Visual G para armazenar e manipular as notas de 5 alunos em uma turma. O programa deve ler as notas dos alunos, calcular a média das notas e exibir as notas e a média.**
   1. **Descreva como você usaria um vetor para armazenar as notas dos 5 alunos e calcular a média.** Declarando: (notas: vetor[1..5]) em real, pois se tratar de notas. Após isso, é o utilizado o “para”, para que seja colocado os dados, dentro do vetor.
   2. **Escreva um código simples em Visual G que declare um vetor para armazenar as notas dos alunos, leia as notas, calcule a média e exiba as notas e a média.**

algoritmo "NotasAlunos"

var

notas: vetor[1..5] de real

soma, media: real

i: inteiro

inicio

// Inicializa a soma das notas

soma <- 0

// Aqui recebe as notas

para i de 1 ate 5 faca

escreva("Digite a nota do aluno ", i, ": ")

leia(notas[i])

soma <- soma + notas[i]

fimpara

// Calculo da media

media <- soma / 5

// Aqui exibe as notas

escreval("Notas dos alunos:")

para i de 1 ate 5 faca

escreval("Aluno ", i, ": ", notas[i])

fimpara

escreval("Média das notas: ", media)

fimalgoritmo

1. **Você está criando um programa em Visual G para somar dois números. O programa deve ter uma função que receba dois números como parâmetros e retorne a soma desses números.**
2. **Como você passaria dois números para a função que calcula a soma?**somaLocal <- a + b
3. **Escreva um código simples em Visual G que declare uma função para calcular a soma, passe dois números como parâmetros e exiba a soma.**

Algoritmo "Soma dos Números com Parâmetro"

Var

num1, num2, resultado: inteiro

Funcao soma(a, b: inteiro): inteiro

Var

somaLocal: inteiro

Inicio

somaLocal <- a + b

Retorne somaLocal

FimFuncao

Inicio

// Recebe os dados

Escreva("Digite o primeiro número: ")

Leia(num1)

Escreva("Digite o segundo número: ")

Leia(num2)

// Chama a função e declara no resultado

resultado <- soma(num1, num2)

// Resultado

EscrevaL("A soma de ", num1, " e ", num2, " é igual a ", resultado)

fimalgoritmo