



ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

ALAN ROCHA DA SILVA JÚNIOR

RA: 3493297801

PROFESSOR: WESLEY VIANA PEREIRA

TUTORA: LUANA GOMES DE SOUZA

2ºSEMESTRE

RELATÓRIO DE AULA PRÁTICA:

ALGORITMOS E PROGRAMAÇÃO ESTRUTURADA

SUMÁRIO

•INTRODUÇÃO.....	1
•DESENVOLVIMENTO.....	2
•CONCLUSÃO.....	3

Relatório de Aula Prática Algoritmos e Programação Estruturada

INTRODUÇÃO:

O seguinte relatório de Algoritmos e Programação Estruturada, do Curso Superior de Tecnólogo em Análise e Desenvolvimento de Sistemas que foi realizado por mim, Alan Rocha da Silva Júnior no Polo da Unopar da cidade de Itabira MG, atividade proposta pelo professor Wesley Viana Pereira. Atividade desenvolvida no programa VisualG.

Neste relatório, será apresentado o processo de desenvolvimento de um programa para calcular a idade e potenciação.

O Visualg é um ambiente de desenvolvimento integrado (IDE) para programação em Portugol, uma linguagem pseudocódigo baseada em algoritmos que facilita o aprendizado de programação. O Visualg oferece uma interface gráfica simples e intuitiva que permite aos iniciantes em programação compreenderem os conceitos fundamentais sem se preocupar com detalhes complexos de uma linguagem de programação completa.

Aqui está uma introdução básica sobre como começar a usar o Visualg:

Após baixar e descompactar o software VisualG, a ferramenta foi executada e foram criados dois algoritmos: A verificação de idade e cálculo matemático de potenciação.

1. Instalação: Primeiro, você precisa fazer o download e instalar o Visualg em seu computador. Ele é um software gratuito e pode ser facilmente encontrado online.

2. Abrindo o Visualg: Após a instalação, inicie o Visualg. Você será recebido com uma tela de início vazia, pronta para começar a escrever seu código.

3. Interface do Visualg: A interface do Visualg é dividida em diferentes seções. A área principal é onde você escreverá seus algoritmos. Há também uma barra de ferramentas com botões para salvar, abrir e executar seus algoritmos.

4. Escrevendo Algoritmos: No Visualg, você escreverá seu código usando uma linguagem semelhante ao pseudocódigo. Por exemplo, para exibir uma mensagem na tela, você usaria o comando `escreval("Olá, mundo!")`. Para ler um valor, usaria `leia(valor)`.

5. Executando Algoritmos: Após escrever seu código, você pode clicar no botão "Executar" ou pressionar F9 para ver a saída do seu algoritmo. O Visualg mostrará os resultados na janela de saída.

6. Depuração: Seu algoritmo pode conter erros. O Visualg pode ajudar a identificar problemas. Se houver algum erro, ele indicará a linha onde ocorreu e fornecerá algumas dicas para resolvê-lo.

7. Exemplos e Recursos: O Visualg geralmente vem com exemplos pré-programados que você pode explorar para entender melhor como os algoritmos são escritos. Também existem muitos tutoriais e recursos online que podem ajudá-lo a aprender mais sobre programação usando o Visualg.

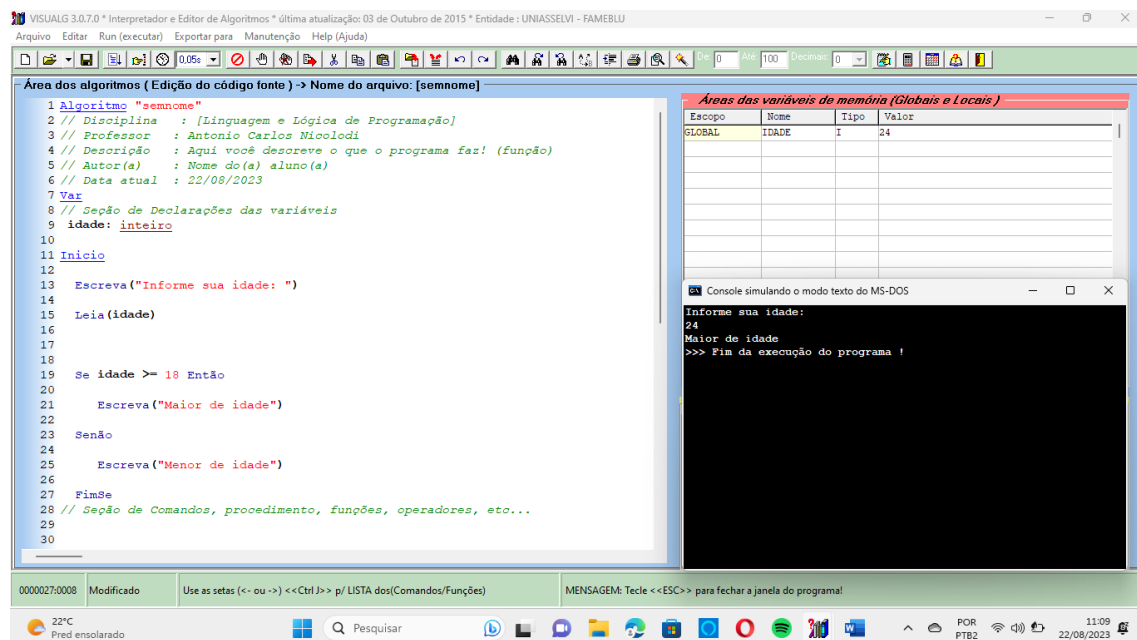
Lembre-se de que o Visualg é uma ferramenta educacional destinada a ajudar iniciantes a aprenderem os conceitos básicos de programação. À medida que você ganha mais experiência, pode ser uma boa ideia começar a explorar linguagens de programação reais, como Python, Java, C++ e outras, para expandir suas habilidades de programação.

Certifique-se de consultar a documentação do Visualg e explorar exemplos para obter uma compreensão mais profunda de como usar essa ferramenta para aprender programação.

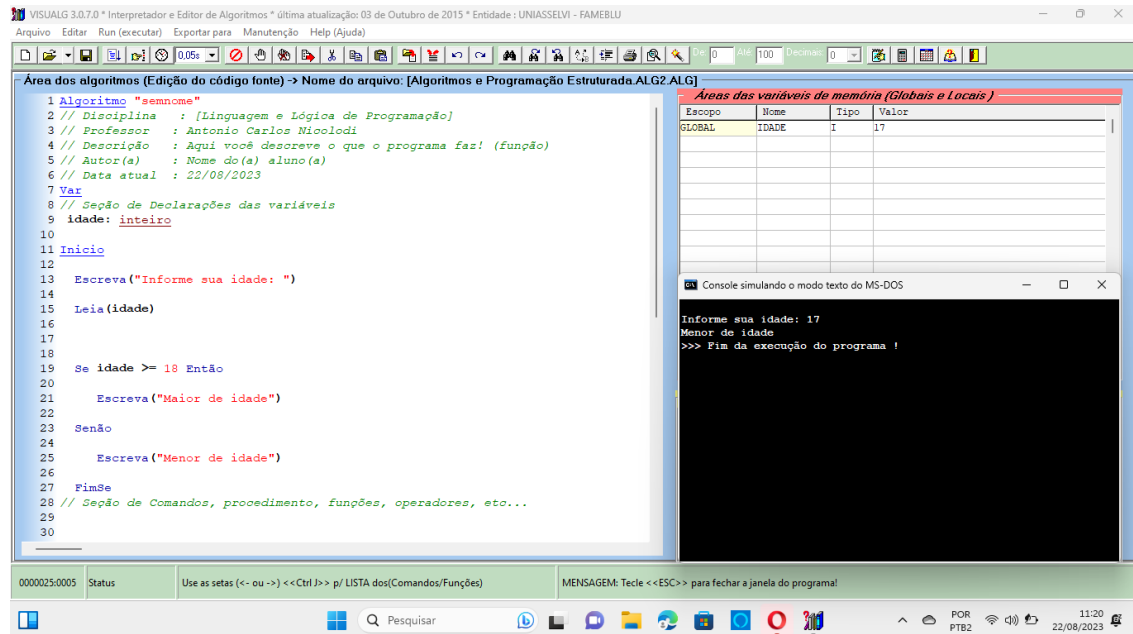
DESENVOLVIMENTO:

Após elaborar o algoritmo de verificação de idade, abaixo evidência visual e o algoritmo em si para testes. Tela apresentando execução do algoritmo:

PRINT ALGORITMO MAIOR DE IDADE



PRINT ALGORITMO MENOR DE IDADE



Código utilizado para execução no VisualG:

1. Algoritmo

Algoritmo "idade"

Var

idade: inteiro

Inicio

escreval("Digite sua idade: ")

leia(idade)

se(idade >= 18) entao

escreval("Você é maior de idade!")

senao

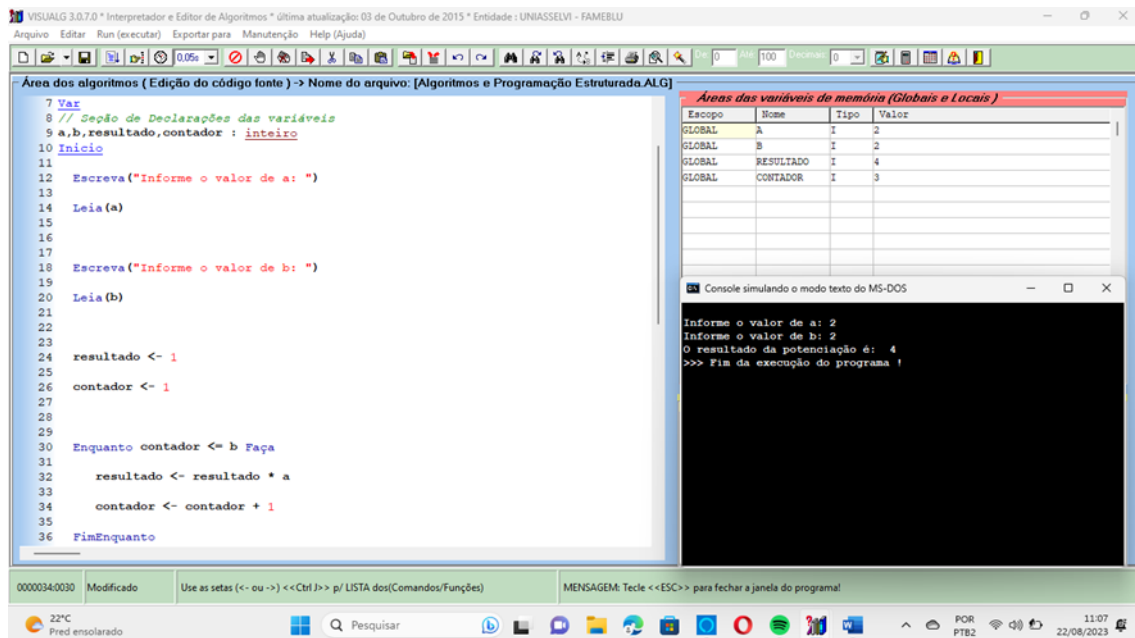
escreval("Você é menor de idade!")

fimse

Fimalgoritmo

Após elaborar o algoritmo de verificação de idade, abaixo evidência visual e o algoritmo em si para testes. Tela apresentando execução do algoritmo:

PRINT POTENCIAÇÃO



Código utilizado para execução no VisualG:

Algoritmo "Potenciação"

Var

a : Inteiro

b : Inteiro

potencia : Inteiro

contador : Inteiro

Inicio

escreval("Por favor, informe o número")

leia(a)

escreval("Por favor, informe o número do expoente")

leia(b)

potência <- 1

contador <- 1

Para contador de 1 até b faça

potência <- potencia * a

fimpara

escreval("Resultado: ", potencia)

Fimalgoritmo

CONCLUSÃO:

Após esta atividade, concluí que o pseudocódigo pode ser um grande aliado para criar algoritmos, resolver problemas e facilitar o entendimento de condicionais e cálculos.