



UNIVERSIDADE DO VALE DO TAQUARI – UNIVATES  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENSINO DE CIÊNCIAS EXATAS –  
MESTRADO

## VISUALG COMO RECURSO DE SIMULAÇÃO NO ENSINO DE ALGORITMOS

### *VISUALG AS A SIMULATION RESOURCE IN THE TEACHING OF ALGORITHMS*

Carlos Eduardo Gomes da Costa<sup>1</sup>, Andreia A. Guimarães Strohschoen<sup>2</sup>, Márcia  
Jussara Hepp Rehfeldt<sup>3</sup>

<sup>1</sup>Mestrando em Ensino de Ciências Exatas – Univates  
[carlos.costa1@universo.univates.br](mailto:carlos.costa1@universo.univates.br)

<sup>2</sup>Dr.<sup>a</sup> em Ciências – Univates – [aaquim@univates.br](mailto:aaquim@univates.br)

<sup>3</sup>Dr.<sup>a</sup> em Informática na Educação – Univates – [mrehfeld@univates.br](mailto:mrehfeld@univates.br)

#### **Finalidade**

Este produto educacional apresenta uma proposta pedagógica para o ensino de algoritmos, com o apoio do *software* VisuAlg. Esta proposta pode ser aplicada tanto com turmas de alunos do Ensino Médio, quanto do Ensino Superior, em qualquer disciplina cuja ementa contemple o ensino de algoritmos. O material contém estratégias de abordagem do conteúdo e traz exercícios propostos para serem explorados, todos utilizando o *software* VisuAlg.

#### **Contextualização**

Este produto educacional originou-se de uma prática de intervenção pedagógica realizada para a dissertação de Mestrado, do Programa de Pós-Graduação em Ensino de Ciências Exatas (PPGECE) da Universidade do Vale do Taquari (Univates). A intervenção foi realizada no Instituto Federal de Mato Grosso (IFMT) /*Campus* Avançado Sinop, na cidade de Sinop (MT), com alunos do 1º Ano do Curso Técnico em Eletromecânica Integrado ao Ensino Médio. Os alunos possuíam pouco ou nenhum conhecimento relativo a algoritmo.

A intervenção iniciou com a participação de 29 alunos e finalizou com a participação de 16 alunos. A redução dos alunos ocorreu devido às adaptações que a escola teve que fazer por causa da pandemia do Covid-19 que ocorreu no ano de 2020, ano de realização da intervenção. Os alunos da turma tinham idade entre 14 e 17 anos, de ambos os sexos, todos estavam em seu primeiro ano na instituição, e ingressaram a partir do processo seletivo que ocorre anualmente, sendo oriundos de escolas públicas e particulares de Sinop e de outras cidades do entorno.

Esta proposta teve origem na percepção do primeiro autor deste produto educacional em relação ao fato de vários alunos apresentarem dificuldade de desenvolver raciocínio lógico, para resolver problemas nas disciplinas técnicas, na instituição em que ele leciona. O professor ministra a disciplina Informática Aplicada e trabalha o conteúdo de algoritmos, que é a base para várias outras disciplinas do curso por desenvolver o raciocínio lógico. Segundo Gomes (2018), raciocínio lógico é a capacidade de analisar dados e formar uma relação entre o essencial e o geral. Além disso, ajuda na concentração, aperfeiçoa a capacidade de interpretação e desenvolve a capacidade de resolver problemas. Conforme Garlet (2016), o aprimoramento do pensamento computacional pode melhorar o rendimento em outras disciplinas.

Estudos relacionados ao ensino do pensamento computacional, como o de Boucinha (2017), relacionam o pensamento computacional com o desenvolvimento do raciocínio lógico e permitem identificar que estratégias pedagógicas precisam ser repensadas para a criação de propostas pedagógicas, que trabalhem junto o pensamento computacional com o uso de tecnologias digitais. Ademais, estudos de Maia (2012) e Inácio (2016) afirmam que a inserção de tecnologias digitais dentro da escola é uma realidade e não tem como não integrar esses recursos à prática docente.

A motivação para o desenvolvimento deste produto educacional surgiu da leitura da pesquisa de Borba (2019, p. 4), que conclui dizendo que o uso do VisuAlg faz que "a motivação dos estudantes aumenta, pois pode-se comprovar que os códigos criados estão funcionando, além da facilidade de poder encontrar os erros", da leitura de outros trabalhos que igualmente recomendam o VisuAlg para o ensino de algoritmo, e também por considerar a sociedade atual que vive cotidianamente

integrada com a tecnologia e por saber que essa tecnologia fornece equipamentos eletrônicos e digitais com capacidade de agregar ou facilitar o conhecimento dentro da sala de aula.

A intenção foi criar uma proposta pedagógica com o uso do *software* VisuAlg como recurso de simulação para o desenvolvimento do conteúdo de algoritmos. O produto educacional propõe aulas de uma forma mais prática, abordando o ensino de algoritmos por meio de situações-problema que provocam determinados cenários para estudos. Esses cenários são trabalhados junto com o desenvolvimento do pensamento computacional que, segundo Boucinha (2017), colabora para o aprimoramento do raciocínio lógico, necessário para a construção de algoritmos.

Existem várias técnicas de ensino que podem ser utilizadas, uma é aplicar tecnologias digitais para expor o assunto e, depois, explicar o conteúdo para que, então, possa ser compreendido de forma mais fácil e clara. Este produto educacional vai ao encontro das ideias de Siemens (2004), quando afirma que é um desafio conseguir ativar o conhecimento conhecido, até o ponto ideal para sua aplicação para qualquer técnica, e de Mota (2009), quando diz que o processo de atribuir valor é visto como uma característica intrínseca para incorporação do conhecimento.

A associação do conteúdo que o professor apresenta com o conhecimento que o aluno tem, em alguns casos, pode não ser fácil de ser percebida pelo aluno, porque cada pessoa tem seus significados com relação a uma situação ou conhecimento. Segundo Masini (2008), quando um professor cria um cenário para tentar passar alguns significados, nem todos os alunos conseguem assimilá-los, porque tiveram experiências diferentes e criaram significados diferentes. Para Inácio (2016, p. 27):

[...] a formação de um aluno crítico e criativo depende justamente da bagagem de informação adquirida e do domínio dos conhecimentos consolidados. [...] acredita-se que quanto, mais exercícios aplicados aos alunos, melhor é a formação de conhecimento destes.

Segundo Maia (2012), trabalhos que proponham aquisição de conhecimento e elaboração de respostas devem ser explorados pelo professor; é necessário que o aluno tenha situações para serem concluídas e com possibilidade de serem modificadas; que seja trabalhado o raciocínio com situações-problema para proporcionar o desenvolvimento cognitivo do aluno, que aprenderá com seus erros.

Assim, aprendem sem perceber que estão aprendendo e que estão adquirindo conhecimento para resolver problemas semelhantes. Esse modo de agir vai ao encontro ao proposto pelo Ministério da Educação nos Parâmetros Curriculares Nacionais, que recomendam ser:

[...] necessário desenvolver habilidades que permitam **pôr à prova os resultados, testar seus efeitos, comparar diferentes caminhos, para obter a solução**. Nessa forma de trabalho, o valor da resposta correta cede lugar ao valor do processo de resolução. O fato de o aluno ser estimulado a questionar sua própria resposta, a questionar o problema, a transformar um dado problema numa fonte de novos problemas, evidencia uma concepção de ensino e aprendizagem não pela mera reprodução de conhecimentos, mas pela via da ação refletida que constrói conhecimentos (BRASIL, 1997, p. 33, grifo do autor).

Segundo Althaus (2016), trabalhar com problemas auxilia na formação de cidadãos e permite-lhes criar seu próprio jeito de encontrar solução, elaborando e desenvolvendo suas estratégias, o que os torna autônomos e responsáveis em relação a seu conhecimento. Em concordância com o epistemólogo Jean Piaget, Inácio (2016, p. 17) afirma que “à medida que as crianças se desenvolvem intelectualmente, criam esquemas mentais para a resolução de problemas das situações cotidianas”. Então, o trabalho com situações-problema desafia os alunos a encontrarem uma solução, apesar de não existir apenas uma solução, pois os alunos podem resolver diferentemente o mesmo problema, um do outro, e todas soluções estarem corretas; o importante é o caminho de planejamento que o aluno faz para encontrar a solução.

No Brasil, existe a Base Nacional Comum Curricular (BNCC), cujo objetivo é garantir um patamar comum de conhecimento para todos os estudantes e que busca assegurar o desenvolvimento de dez competências gerais. A BNCC define competência “como a mobilização de conhecimentos [...], habilidades [...], atitudes e valores para resolver demandas complexas da vida cotidiana, do pleno exercício da cidadania e do mundo do trabalho” (BRASIL, 2019, p. 8). Em várias competências, é citada a utilização de experimentos por meio de simulação, utilização de *softwares* para possibilitar confrontar dados, realizar projeções e análise de novas ideias, teste de situações, validação de viabilidade e riscos de determinados cenários serem utilizados nos trabalhos (BRASIL, 2019).

Na disciplina Informática Aplicada, utilizada para a realização das intervenções, os alunos aprendem a criar algoritmos e precisam desenvolver seu raciocínio lógico para resolver os problemas que lhes são propostos. Portanto, no trabalho de forma experimental, com situações-problema que permitam os alunos

testarem suas ideias usando um simulador, não é possível não pensar na utilização de recursos tecnológicos e digitais, como vários autores e a própria BNCC defendem.

Este produto educacional propõe o ensino de algoritmos, com aulas expositivas, utilizando o *software* VisuAlg para demonstrações durante as explicações, trazendo também exercícios de fixação, em forma de situações-problema, proporcionando cenários variados, com dados de validação para testar e validar os algoritmos, e simulando no VisuAlg.

### **Objetivo**

Apresentar uma proposta pedagógica, utilizando o *software* VisuAlg como recurso de simulação, com o intuito de contribuir para o ensino de algoritmos para estudantes do ensino técnico ou superior.

### **Detalhamento**

Esta investigação ocorreu no primeiro semestre do ano de 2020, teve início com a participação de 29 alunos e finalizou com a participação de 16 alunos do 1º Ano do Curso Técnico em Eletromecânica Integrado ao Ensino Médio do IFMT / *Campus* Avançado de Sinop. Na sequência, apresentamos as atividades planejadas deste trabalho, divididas em 13 encontros. A partir do terceiro, os encontros devem ocorrer em uma sala onde os alunos tenham acesso ao uso de computadores com o *software* VisuAlg instalado.

No primeiro encontro, antes de começar a falar sobre o que é um algoritmo e seus conceitos, pergunte aos alunos se eles sabem preparar algum tipo de alimento e, em seguida, após definir qual alimento será usado, pergunte por onde se começa seu preparo. Questione-os até algum aluno falar que se inicia a partir da receita, então pergunte o que a receita fornece. Com suporte nessas respostas, detalhando o que a receita fornece, faça uma conclusão junto com eles de que toda essa informação da receita é um passo a passo para preparar o alimento e, ao final, ter um alimento pronto para ser servido.

Após entenderem que a receita é um passo a passo, escolha um alimento que a maioria conheça como preparar e pergunte quais os passos necessários para o preparo. Anote no quadro a sequência de passos que os alunos disserem ser necessários para a preparação. Ao iniciar os passos que os alunos citaram, mostre a

importância da sequência correta e também a falta de mais informações, ou seja, de passos que ficaram faltando entre um e outro, que possam impedir o preparo correto, caso não se conheça como preparar aquele alimento. Essa dinâmica tem o propósito de fazê-los perceber a importância de ser colocado o máximo de informações no passo a passo, e de entenderem a necessidade de os passos estarem claros e na sequência correta.

Em seguida, questione o que eles acham ser um algoritmo, considerando o que foi discutido até aquele momento. Após expressarem o que julgam ser um algoritmo, então comece a expor o conceito do que é um algoritmo, fazendo analogia entre os conceitos e o passo a passo da receita que foi anotada no quadro.

Ao concluir a explicação dos conceitos de algoritmo, aplique como desafio o Exercício 01 do Quadro 1. Os alunos devem responder baseados no que entenderam até aquele momento, defina um tempo para responderem. Ao término do tempo, solicite que, de forma voluntária, leiam suas respostas. Após alguns alunos lerem, faça a correção e, em seguida, aplique o Exercício 02 do Quadro 1 com a mesma dinâmica.

#### Quadro 1 – Exercícios propostos para o primeiro encontro

<b>Encontro 01</b>
<i>Exercício 01</i> - Escreva um algoritmo para trocar uma lâmpada com soquete rosqueável que está queimada.
<i>Exercício 02</i> - Escreva um algoritmo para trocar pneu furado de um carro que você esteja dirigindo.

Fonte: do autor (2020).

Após a leitura das respostas de cada exercício, antes de iniciar a correção, faça observações acerca das respostas lidas sobre a sequência dos passos, da falta de informações entre um passo e outro, da omissão de informações importantes e de outras observações que julgar necessárias para que o algoritmo possa ficar o mais completo possível.

Inicie o segundo encontro, lembrando os exercícios do encontro anterior e relacionando com os conceitos de algoritmos que foram apresentados. Após revisar tudo que foi estudado, mostre a resolução de um dos exercícios de sua preferência. A resolução deve conter os passos necessários do ponto de vista do professor (Figura 1), com o máximo de passos possíveis, respeitando a sequência das ações, e explique a necessidade de cada passo adicionado.

Figura 1 – Exemplo de resolução de um algoritmo

- ▶ Início
- ▶ Verifica se o interruptor está desligado;
- ▶ Procura uma lâmpada nova;
- ▶ Pega uma escada;
- ▶ Leva a escada até o local;
- ▶ Posiciona a escada;
- ▶ Sobe os degraus;
- ▶ Para na altura apropriada;
- ▶ Retira a lâmpada queimada rosqueando;
- ▶ Coloca a lâmpada nova rosqueando;
- ▶ Desce da escada;
- ▶ Acione o interruptor para testar;
- ▶ Joga a lâmpada queimada no lixo;
- ▶ Guarda a escada;
- ▶ Fim

Fonte: do autor (2020).

Em seguida, apresente as três formas de se construírem algoritmos: textual, fluxograma e pseudocódigo. Então, explique que os exercícios realizados até aquele momento foram no formato textual. Para expor os outros formatos de algoritmos, recomendamos valer-se do mesmo cenário para criação dos algoritmos nos três formatos. Sugerimos utilizar o cenário de cálculo de nota do bimestre, porque é um cenário que os alunos já sabem os passos necessários, e pode ser mais fácil conseguirem associar o que fazem na prática com os passos necessários que o algoritmo deve contemplar.

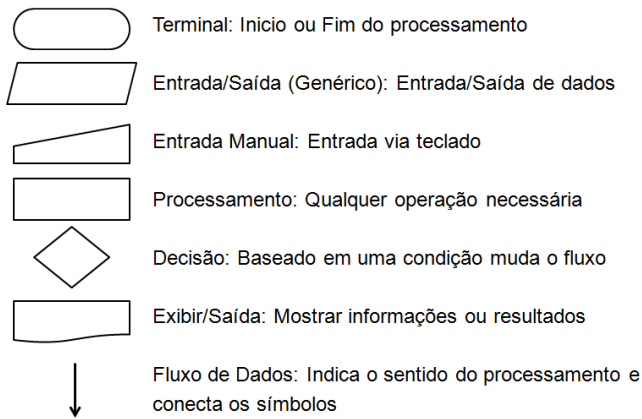
Durante a explicação do tipo fluxograma, quando apresentar o exemplo, faça associação dos passos em fluxograma com o exemplo mostrado no formato textual. Após demonstrar os exemplos no formato textual e fluxograma, explique o significado de cada uma das principais figuras que podem ser utilizadas na construção de algoritmos no formato de fluxograma (Figura 2).

Ao encerrar a explicação, aplique o Exercício 01 do Quadro 3, defina um tempo para responderem. Encerrado o tempo, faça a correção perguntando aos alunos o que eles responderam e anote no quadro. Durante a correção, quando um aluno ler a resposta, questione os demais se responderam da mesma forma ou diferente. Caso algum aluno tenha respondido diferente, solicite que leia sua resposta e questione os outros sobre qual das respostas é a correta e por qual motivo. Ao final, corrija e explique os erros que aparecerem.

Figura 2 – Figuras para construção de algoritmo do tipo fluxograma

## Algoritmos

### ▶ Fluxograma – Símbolos

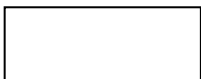
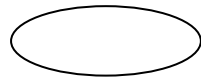
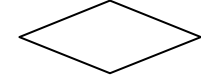
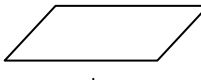




Fonte: do autor (2020).

### Quadro 3 – Exercícios propostos para o segundo encontro

#### Encontro 02

*Exercício 01* – Escreva nas figuras abaixo quando ela deve ser usada na construção de um fluxograma.

	_____
	_____
	_____
	_____
	_____
	_____

*Exercício 02* – Faça um fluxograma que leia 4 notas, calcule e mostre a média.

*Exercício 03* – Faça um fluxograma que leia 2 notas e calcule a média. Se a média for maior que 7,0 escreva aprovado; se for menor, escreva reprovado.

Fonte: do autor (2020).



Encerrada a correção do primeiro exercício, aplique o Exercício 02 do Quadro 3, defina um tempo para responderem. Encerrado o tempo, corrija o exercício perguntando como responderam cada passo do algoritmo, questione os demais alunos se todos concordam ou não com a resposta lida; caso alguma resposta esteja errada e ninguém questione, então explique o erro e como seria o correto. Finalizada a correção, aplique o Exercício 03 do Quadro 3, defina um tempo para responderem.

Inicie o terceiro encontro, revisando os tipos de algoritmos estudados e suas principais diferenças; após, explique o tipo de algoritmo em formato de pseudocódigo, apresentando os principais comandos (Figura 3). Explique quando e por que deve ser utilizado cada comando.

Figura 3 – Comandos de algoritmo para o tipo pseudocódigo

- ▶ Comandos
  - ▶ Início
    - ▶ Indica o início dos comandos.
  - ▶ Escreva
    - ▶ Mostra uma mensagem ao usuário.
  - ▶ Leia
    - ▶ Lê dados de entrada.
  - ▶ Se
    - ▶ Assinala se há uma condição.
  - ▶ Fim
    - ▶ Indica o fim dos comandos.

Fonte: do autor (2020).

Após explicar os comandos, mostre um exemplo de algoritmo no formato de pseudocódigo. Utilize o mesmo cenário usado para explicar o tipo textual e fluxograma, para os alunos poderem comparar o mesmo problema escrito em formato textual (Figura 4), fluxograma (Figura 5) e pseudocódigo (Figura 6); assim, podem fazer comparações e analogias.

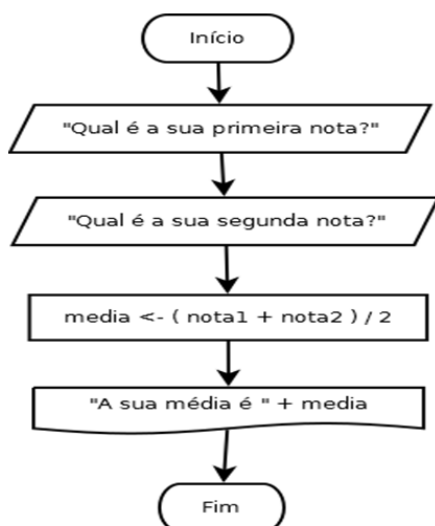
Abra o *software* VisuAlg (Figura 7) e apresente-o aos alunos (Apêndice A), explique sua finalidade e por que será utilizado. Então, recupere o fluxograma de correção do Exercício 02 do Quadro 3 e transcreva-o para o formato pseudocódigo diretamente no VisuAlg. Durante a escrita, simule o algoritmo a começar da escrita de cada comando e demonstre o comportamento dos comandos, utilizando valores aleatórios.

Figura 4 – Exemplo de Algoritmo Textual

- ▶ Início
- ▶ Qual é sua primeira nota?
- ▶ Qual é sua segunda nota?
- ▶ Some a primeira nota com a segunda nota e divida por 2.
- ▶ Mostre o resultado encontrado com os cálculos.
- ▶ Fim

Fonte: do autor (2020).

Figura 5 – Exemplo de Algoritmo Fluxograma



Fonte: do autor (2020).

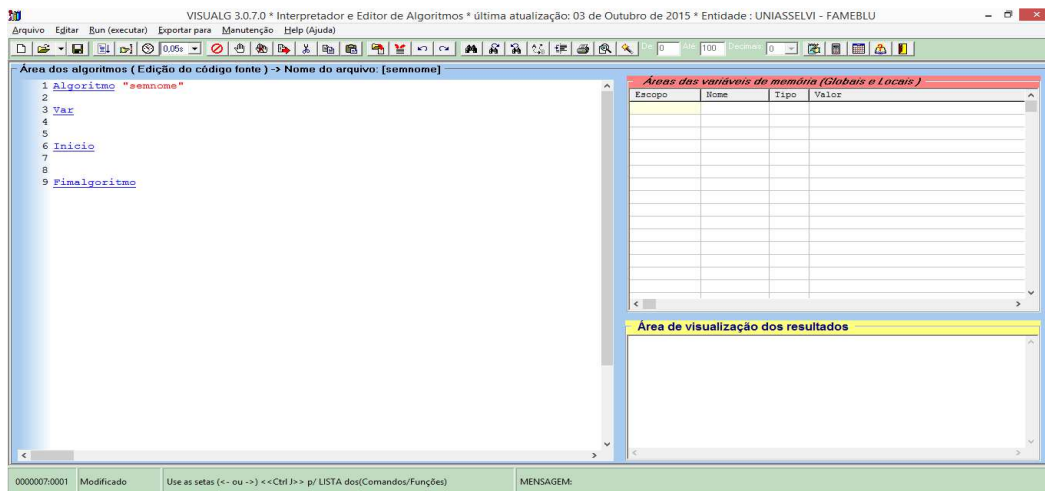
Figura 6 – Exemplo de Algoritmo Pseudocódigo

```

1 Algoritmo "media"
2
3 Var
4 nota1, nota2 : inteiro
5 media : real
6 Inicio
7
8 escreva("Qual é a sua primeira nota?")
9 leia ( nota1 )
10
11 escreva("Qual é a sua segunda nota?")
12 leia ( nota2 )
13
14 media <- (nota1 + nota2) / 2
15
16 escreva("Sua média é ", media)
17
18 Fimalgoritmo
  
```

Fonte: do autor (2020).

Figura 7 – Tela do *software* VisuAlg



Fonte: do autor (2020).

Durante a resolução do exercício, explique como o *software* pode ser utilizado (Apêndice A), os comandos que podem ser utilizados, como pode ajudar na escrita, como testar e simular os algoritmos. Também explique erros comuns que possam ocorrer durante a escrita e como corrigi-los.

Após transcrever o exercício e explicar o uso do *software*, faça a correção do Exercício 03 do Quadro 3; primeiro, no formato de fluxograma e, depois, em pseudocódigo diretamente no VisuAlg. Durante a correção, utilize apenas condicional simples. Aplique a mesma dinâmica já empregada nos exercícios anteriores, perguntando aos alunos como eles responderam, anotando no quadro as respostas e questionando os outros se concordam ou não, até chegar ao final da resolução. Durante a escrita, no formato pseudocódigo, também siga a mesma dinâmica, simulando cada comando escrito no algoritmo para demonstrar seu comportamento.

Terminada a correção, aplique o Exercício 01 do Quadro 4, defina um tempo para responderem. Encerrado o tempo, corrija o exercício utilizando a mesma dinâmica já relatada nas correções anteriores. Aplique o Exercício 02 do Quadro 4, defina um tempo para responderem. Encerrado o tempo, corrija o exercício aplicando a mesma dinâmica das correções anteriores. Solicite que respondam primeiro em fluxograma e, posteriormente, em pseudocódigo, escrevendo diretamente no *software* VisuAlg.

#### Quadro 4 – Exercícios propostos para o terceiro encontro

<b>Encontro 03</b>	
<i>Exercício 01</i> – Escreva nos comandos abaixo quando devem ser usados na construção de um pseudocódigo.	
Escreva	_____
Início	_____
Leia	_____
Fim	_____
Se	_____
<i>Exercício 02</i> – Faça o fluxograma e, depois de concluir faça o pseudocódigo que pergunte o ano de nascimento da pessoa; então, deve calcular a idade e mostrar qual a idade encontrada.	
Validação:	
Caso 01	Caso 02
Entrada - 1980	Entrada - 1999
Saída - 40	Saída - 21

Fonte: do autor (2020).

No quarto encontro, aplique os exercícios do Quadro 5, informe que devem fazer primeiramente o fluxograma e, posteriormente, o pseudocódigo no VisuAlg; corrija os exercícios, utilizando a mesma dinâmica já relatada nas correções anteriores.

#### Quadro 5 – Exercícios propostos para o quarto encontro

<b>Encontro 04</b>	
<i>Exercício 01</i> – Faça um algoritmo para ler a base e a altura de um triângulo. Em seguida, escreva sua área. Obs.: $\text{Área} = (\text{Base} * \text{Altura}) / 2$	
Validação:	
Caso 01	Caso 02
Entrada – base 10 e altura 5	Entrada – base 4 e altura 11
Saída - A área do triângulo é 25	Saída - A área do triângulo é 22
<i>Exercício 02</i> – Uma empresa paga 12 reais por hora trabalhada e 40 reais por filho aos funcionários. Faça um algoritmo para ler o número de horas trabalhadas e número de filhos de um funcionário, após deve calcular e mostrar o salário.	
Validação:	
Caso 01	Caso 02
Entrada - 1000 horas e 0 filhos	Entrada - 650 horas e 3 filhos
Saída - O salário é 12000	Saída - O salário é 7920
<i>Exercício 03</i> – Faça um algoritmo para ler a distância da escola até a casa do aluno e qual o tempo que ele gasta para chegar a sua casa; então, calcule e mostre a velocidade média do aluno para percorrer esse trajeto. Obs.: $V_m = \text{distancia}/\text{tempo}$	

Validação:

Caso 01

Entrada - 1000 e 5

Saída - A velocidade média é 200 metros por minuto

Caso 02

Entrada - 4500 e 10

Saída - A velocidade média é 450 metros por minuto

Fonte: do autor (2020).

No quinto encontro, explique o que é declaração de variável, como declarar uma variável e quais tipos existentes. Utilize o VisuAlg para demonstrar como os valores são armazenados em cada tipo de variável e mostre a diferença de valores que cada tipo permite armazenar. Em seguida, explique o que é uma expressão aritmética, quais caracteres podem ser utilizados, seu significado e qual operação ele representa. Simule, no VisuAlg, operações aritméticas para demonstrar o comportamento e resultado de todos caracteres que podem ser usados.

Explique o que é uma expressão lógica, como deve ser montada, quais operadores lógicos de comparação podem ser utilizados, quando deve ser empregado cada operador de comparação e qual resultado é retornado em uma expressão lógica. Em seguida, apresente os operadores lógicos para união de expressões lógicas, quando devem ser utilizados e qual valor será retornado. Simule, no VisuAlg, expressões lógicas para demonstrar o comportamento e resultado de todos operadores lógicos que podem ser úteis valendo-se de uma ou mais de uma expressão. Então, aplique a lista de exercício do Quadro 6.

#### Quadro 6 – Exercícios propostos para o quinto encontro

##### Encontro 05

*Exercício 01* – Dadas as alternativas abaixo, responda o valor das operações e expressões:

- a)  $(50+10)/8$  \_\_\_\_\_
- b)  $25\%10$  \_\_\_\_\_
- c)  $50/5$  \_\_\_\_\_
- d)  $(10-(5*5)+90)$  \_\_\_\_\_
- e)  $2^5$  \_\_\_\_\_
- f)  $(80 \geq 50)$  \_\_\_\_\_
- g)  $(30 \leq 30)$  \_\_\_\_\_
- h)  $(10 \neq 5)$  \_\_\_\_\_
- i)  $((5 < 5) \text{ ou } (10 > 10))$  \_\_\_\_\_
- j)  $((10 < 50) \text{ e } (3 > 1))$  \_\_\_\_\_
- k)  $((11,1 > 11,0) \text{ ou } (9,9 > 9,9))$  \_\_\_\_\_

- l) (verdadeiro e verdadeiro) \_\_\_\_\_
- m) (verdadeiro e falso) \_\_\_\_\_
- n) (falso ou verdadeiro) \_\_\_\_\_
- o) (falso ou falso) \_\_\_\_\_
- p) (verdadeiro ^ verdadeiro) \_\_\_\_\_
- q) (verdadeiro ^ falso) \_\_\_\_\_
- r) (falso ^ falso) \_\_\_\_\_

*Exercício 02* – Uma empresa precisa controlar a câmara fria de forma automática, e alertas de atenção precisam ser acionados quando a temperatura ficar muito baixa, abaixo de 10, ou muito alta, acima de 20, e também quando a diferença de temperatura da leitura atual para a anterior for maior que 4. Dado o algoritmo abaixo, complete os comandos e operadores para que o algoritmo funcione corretamente para o problema dado.

```

Algoritmo "Câmara Fria"
Var
//Entrada
tempAnterior, tempAtual : inteiro
//Auxiliar
tempDiferenca : inteiro
Inicio
  escreva("Qual temperatura lida anteriormente?")
  leia(tempAnterior)
  escreva("Qual temperatura atual?")
  leia(tempAtual)
  tempDiferenca <- tempAnterior _____ tempAtual
  se ( tempDiferenca _____ 4 ) _____ ( tempDiferenca _____ -4 ) entao
    escreva("Diferença de temperatura muito alta. Verificar sistema!")
  fimse
  se ( temAtual _____ 20 ) entao
    escreva ("Temperatura muito alta, iniciar resfriamento!")
  fimse
  se ( temAtual _____ 10 ) entao
    escreva ("Temperatura muito baixa, desligando resfriamento!")
  fimse
Fimalgoritmo

```

*Exercício 03* – Escreva um algoritmo que deve ler apenas um caractere informado pelo usuário, se for informado o caractere X deve apresentar a mensagem que ganhou, para qualquer outro caractere informado deve mostrar a mensagem que foi informado caractere inválido.

*Exercício 04* – Escreva um algoritmo que leia o nome da disciplina, 2 notas e então calcule a média. Nas avaliações, poderão ser informados números fracionados. No final, deve ser mostrado o nome da disciplina e a média encontrada.

Validação:

Caso 01

Entrada - Matemática / 6,2 / 8,4

Saída - A média em Matemática é 7,3

Caso 2

Entrada - Português / 1,8 / 9,0

Saída - A média em Português é 5,4

Fonte: do autor (2020).

No sexto encontro, destine o tempo para os alunos terminarem de resolver os exercícios aplicados no último encontro, para que possam tirar as dúvidas do conteúdo e dos exercícios.

Faça a correção de todos os exercícios referentes ao Quadro 6, no sétimo encontro. Corrija os exercícios, utilizando a mesma dinâmica já relatada nas correções anteriores. Durante a correção, para cada comando escrito no VisuAlg, teste o algoritmo para verificar se o comportamento esperado é o que acontece. Durante as correções, apresente situações de possíveis erros que possam acontecer durante sua construção, mostre porque acontece o erro, como identificá-lo e como corrigi-lo.

No oitavo encontro, explique o conteúdo de estrutura de decisão simples, utilize um exemplo de algoritmo em formato de fluxograma e, depois, o mesmo exemplo em formato de pseudocódigo. Faça uso de um exemplo com apenas uma condicional simples e, depois, outro exemplo com duas condicionais simples. Recomendamos a utilização do cenário de verificar a média do aluno para informar se foi aprovado, porque esse cenário poderá ser evoluído durante a explicação do restante do conteúdo, e os alunos conhecem bem como ele ocorre.

Com o exemplo de algoritmo escrito no VisuAlg, simule utilizando vários valores e demonstre passo a passo o fluxo que o algoritmo percorre para cada valor informado; mostre, de forma visual, a execução do algoritmo com uma e, depois, com duas estruturas de decisão simples.

Ao concluir as demonstrações de decisão simples, explique a estrutura de decisão composta, utilize um exemplo em formato de fluxograma e, depois, o mesmo exemplo no formato de pseudocódigo. Aproveite os exemplos de fluxograma e pseudocódigo que foram aplicados na explicação da estrutura de decisão simples, modifique-os e demonstre utilizando estrutura de decisão composta. Durante a alteração, faça comparações, mostre as diferenças. Quando o exemplo em pseudocódigo tiver alterado no VisuAlg, simule-o com vários valores e demonstre o passo a passo que o algoritmo percorre para cada valor, mostrando de forma visual a execução do algoritmo com mais de uma estrutura de decisão simples, junto com uma estrutura de decisão composta. Então, explique e aplique os exercícios do Quadro 7.

#### Quadro 7 – Exercícios propostos para o oitavo encontro

<b>Encontro 08</b>
<i>Exercício 01</i> – Faça um algoritmo que leia a idade e informe se a pessoa é uma criança para a idade até 12 anos, mostre adolescente para a idade até 18 anos, mostre adulto para a idade até 60 anos e mostre idoso para a idade maior que 60 anos.

*Exercício 02* – Faça um questionário que contenha 5 perguntas quaisquer para responder apenas verdadeiro ou falso. Após o preenchimento de todo o questionário, se tiver sido respondido 3 ou mais questões como verdadeiro, deve mostrar uma mensagem, dizendo que ela passou para a próxima fase.

Validação:

Caso 01

Entrada - verdadeiro / verdadeiro / falso / falso / falso

Saída -

Caso 02

Entrada - verdadeiro / verdadeiro / falso / verdadeiro / falso

Saída - Passou para próxima fase

Caso 03

Entrada - falso / verdadeiro / verdadeiro / verdadeiro / verdadeiro

Saída - Passou para próxima fase

*Exercício 03* – Faça um algoritmo que identifique qual tipo de multa será emitida, deve ser informada a velocidade do veículo, a velocidade permitida e considere as informações abaixo para definir o tipo de multa que será aplicada:

- Até 10% acima da velocidade permitida, multa leve.

- Até 20% acima da velocidade permitida, multa grave.

- Acima de 20%, multa gravíssima.

Validação:

Caso 01

Entrada - 55 / 60

Saída -

Caso 03

Entrada - 120 / 100

Saída - Multa grave

Caso 02

Entrada - 65 / 60

Saída - Multa leve

Caso 04

Entrada - 160 / 110

Saída - Multa gravíssima

Fonte: do autor (2020).

Destine o tempo do nono encontro para os alunos terminarem de resolver os exercícios aplicados no último encontro, para tirarem as dúvidas do conteúdo e dos exercícios.

Faça a correção de todos os exercícios referentes ao Quadro 7, no décimo encontro. Corrija os exercícios, servindo-se da mesma dinâmica já relatada nas correções anteriores. Empregue os valores de validação dos exercícios para testar e validar o algoritmo, simule no VisuAlg para comprovar que o algoritmo ficou escrito corretamente.

No décimo primeiro encontro, explique a estrutura de decisão dupla. Após explicar o conceito, recupere o fluxograma e pseudocódigo de exemplo utilizado para explicar condição simples e composta, modifique-o e mostre a diferença entre



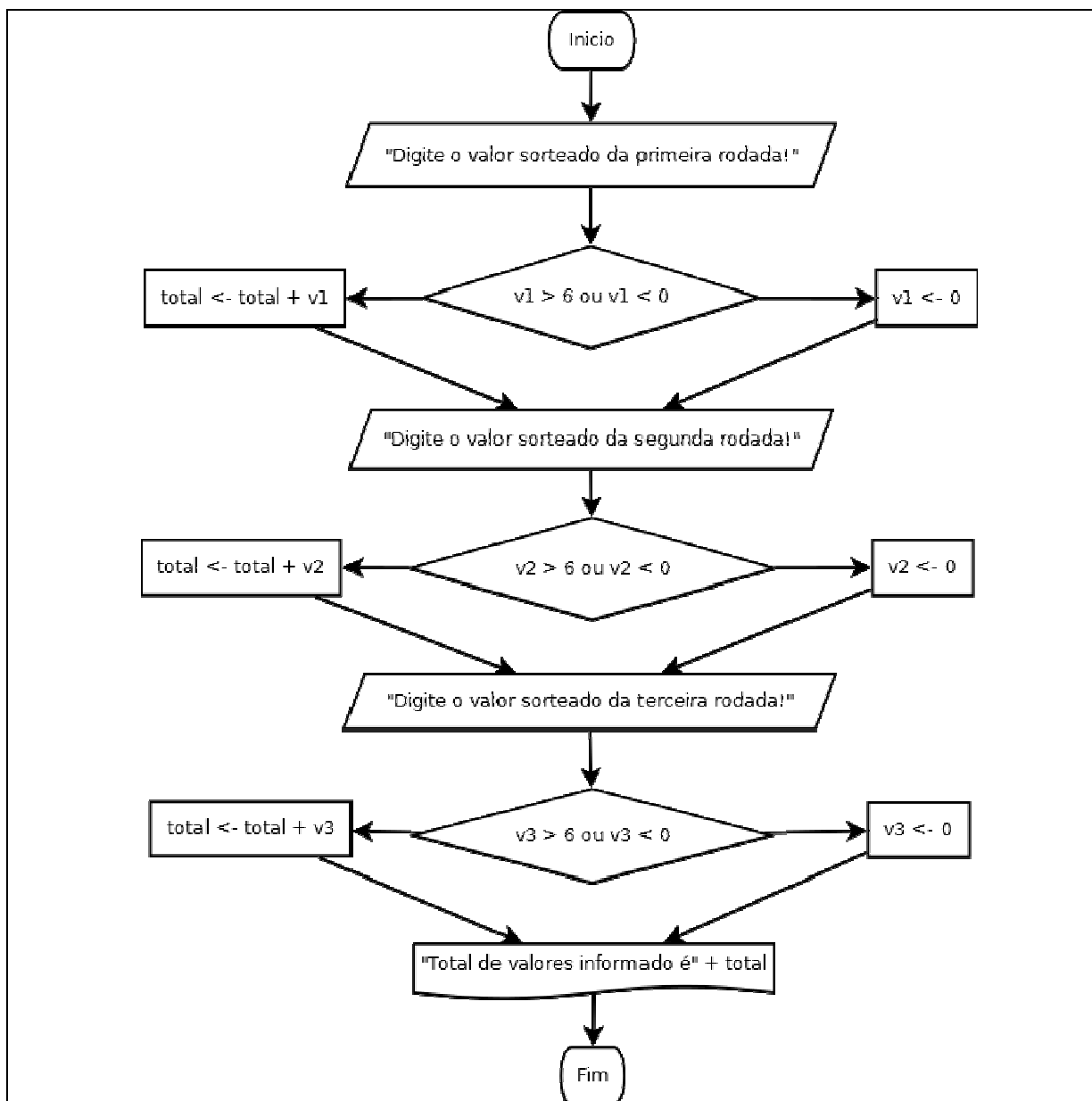
condição simples para condição dupla. Primeiro altere o fluxograma e, depois, o pseudocódigo; após alterar o pseudocódigo, simule-o no VisuAlg, demonstrando a diferença de usar simples ou dupla. Utilize vários valores para poder apontar todos os possíveis comportamentos do algoritmo e demonstrar passo a passo o que acontece, durante a execução, para cada valor informado.

Encerrada a explicação de decisão dupla, explique a estrutura de decisão encadeada. A partir do fluxograma e pseudocódigo aplicado na explanação anterior, de condição dupla, modifique-o para explicar a diferença de condição dupla ou simples para condição encadeada. Após alterar o pseudocódigo, simule-o no VisuAlg e demonstre a diferença em usar condição simples, dupla ou encadeada. Na simulação, opere vários valores para conseguir mostrar todos os possíveis comportamentos do algoritmo e demonstrar, passo a passo, o que acontece durante sua execução.

Ao final de cada explicação, utilize o VisuAlg para apresentar situações que possam ocasionar erros durante a escrita dos algoritmos, simule os erros, mostre como identificá-los e como corrigi-los. Então, elucide e aplique os exercícios do Quadro 8.

#### Quadro 8 – Primeiro exercício proposto para o décimo primeiro encontro

<b>Encontro 11</b>		
<i>Exercício 01</i> – Considerado um jogo onde o jogador precisa jogar um dado a cada rodada e informar o valor sorteado, o jogo termina depois de 3 rodadas. Se o valor informado for maior que 6 ou menor que 1, deve ser considerado o valor 0. No final, deve mostrar a soma de todos os valores informados, conforme fluxograma abaixo. Agora, reescreva em pseudocódigo.		
Validação:		
Caso 01	Caso 02	Caso 03
Entrada - 1 / 6 / 3	Entrada - 5 / 1 / 7	Entrada - 6 / 1 / -1
Saída - 10	Saída - 6	Saída - 7



*Exercício 02* – Uma empresa precisa controlar o gerador de forma automática, ele precisa ser acionado quando faltar energia e desligar quando a energia voltar. A leitura para verificar se há energia ou não será respondida como verdadeiro ou falso. Dado o algoritmo abaixo, complete os comandos e operadores para que o algoritmo funcione corretamente para o problema dado.

Algoritmo "controlar gerador"

Var

//Entrada

energia : logico

Inicio

  escreva("Tem energia?")

  leia(energia)

  \_\_\_\_\_ ( energia ) \_\_\_\_\_

  escreva("Ligando gerador!!!")

  \_\_\_\_\_ escreva ("Desligando gerador!!!")

Fimalgoritmo

*Exercício 03* – Faça um algoritmo que solicite a idade da pessoa. Para idades acima e igual 21, deve mostrar a mensagem de acesso liberado e, para idades menores deve mostrar acesso não liberado.

Validação:

Caso 01

Entrada - 10

Saída - acesso não liberado

Caso 02

Entrada - 21

Saída - acesso liberado

Caso 03

Entrada - 30

Saída - acesso liberado.

*Exercício 04* – Faça um pseudocódigo que leia um valor de 0 a 100 e mostre a mensagem conforme regras abaixo:

- Se valor menor ou igual 20, mostre "muito baixo";
- Se valor maior ou igual 21 e menor ou igual 50, mostre "valor próximo, mas é pouco";
- Se valor maior ou igual 51 e menor ou igual 70, mostre "excelente, valor correto";
- Se valor maior ou igual 71, mostre "valor muito alto".

Validação:

Caso 01

Entrada - 10

Saída - Muito baixo

Caso 02

Entrada - 50

Saída - Valor próximo, mas é pouco

Caso 03

Entrada - 51

Saída - Excelente, valor correto

Caso 04

Entrada - 90

Saída - Valor muito alto

*Exercício 05* – Dados três valores A, B e C, verifique se os valores informados podem ser os comprimentos dos lados de um triângulo. Se forem, deve mostrar o tipo de triângulo: equilátero ( todos os lados iguais), isósceles (somente dois lados iguais) ou escaleno (todos os lados diferentes). Se os valores não formarem um triângulo, deve mostrar mensagem que não compõe um triângulo.

Validação:

Caso 01

Entrada - 10 / 14 / 5

Saída - Triângulo Escaleno

Caso 02

Entrada - 20 / 20 / 10

Saída - Triângulo Isóscele

Caso 03

Entrada - 15 / 15 / 15

Saída - Triângulo Equilátero

Caso 04

Entrada - 10 / 10 / 0

Saída - Não compõe um triângulo

*Exercício 06* – Para um aluno ser aprovado de ano, precisa alcançar uma média anual igual ou superior a 6,0 e não ter mais do que 25% de faltas na disciplina. Faça um algoritmo que pergunte ao aluno qual sua nota em cada bimestre, considere que o ano tem 4 bimestres. Deve perguntar também qual a nota da prova final, qual a quantidade de faltas ele teve, qual a quantidade de aulas da disciplina e deve mostrar as mensagens conforme regras abaixo:

- Se a média anual das notas dos bimestres for inferior a 6,0, deve mostrar a mensagem "Ficou de prova final"; caso contrário, deve mostrar "Aprovado sem prova final";
- Se a média anual das notas dos bimestres for inferior a 6,0, deve ser somado a média anual com a nota da prova final e dividir por 2. Se o resultado encontrado for menor que 6,0, deve mostrar a mensagem "Reprovado após prova final"; caso contrário, deve mostrar "Aprovado após prova final".

- Se tiver tido mais de 25% de faltas, deve mostrar a mensagem "Reprovado por falta"; caso contrário, deve mostrar "Não reprovado por falta".

Validação:

Caso 1

Entrada - Notas: 8.0 7.0 5.0 6.0

Nota Prova Final: 0.0

Quantidade Falta: 2

Quantidade de Aulas: 80

Saída - Aprovado sem prova final

Não reprovado por falta

Caso 2

Entrada - Notas: 1.0 9.0 3.0 2.0

Nota Prova Final: 5.0

Quantidade Falta: 40

Quantidade de Aulas: 80

Saída - Ficou de prova final

Reprovado após prova final

Reprovado por falta

Caso 3

Entrada - Notas: 5.0 4.0 6.0 7.0

Nota Prova Final: 9.0

Quantidade Falta: 5

Quantidade de Aulas: 60

Saída - Ficou de prova final

Aprovado após prova final

Não reprovado por falta

Fonte: do autor (2020).

Destine o tempo do décimo segundo encontro para os alunos terminarem de resolver os exercícios aplicados no último encontro, para tirarem dúvidas do conteúdo e dos exercícios.

Faça a correção de todos os exercícios referentes ao Quadro 8, no décimo terceiro encontro. Corrija os exercícios, utilizando a mesma dinâmica já relatada nas correções anteriores. Utilize os valores de validação dos exercícios para testar e validar o algoritmo; simule no VisuAlg para comprovar se o algoritmo ficou escrito corretamente.

O Apêndice B contém uma lista de questões que podem ser aplicadas ao final dos encontros sugeridos neste produto educacional para realizar uma verificação do conhecimento adquirido pelos alunos sobre o conteúdo de algoritmo após ter aplicado esta proposta pedagógica.

## Resultados obtidos

Percebemos que a utilização do *software* VisuAlg contribui para o ensino de algoritmos, melhora o desenvolvimento do raciocínio lógico e do pensamento computacional. Aplicá-lo para a resolução dos exercícios trouxe autonomia aos alunos, motivando-os e provocando desafios entre eles para conseguirem concluir as atividades.

Os resultados obtidos com a aplicação deste trabalho mostraram que a utilização do *software* VisuAlg para simular exemplos de algoritmos, durante as explicações dos conceitos e comandos, foi satisfatória. Por meio da simulação de exemplos que abordaram cenários provavelmente conhecidos pelos alunos, os conceitos deixaram de ser algo não imaginável, e os alunos puderam ver na prática o que acontece quando se cria um algoritmo; isso fez que prestassem mais atenção às aulas.

Podemos perceber também que a utilização de tecnologias e ferramentas computacionais no ensino, desde que integradas ao planejamento, pode facilitar o ensino, pois permite que as aulas sejam mais dinâmicas e interativas. O uso desperta nos alunos maior interesse em participar e, de alguma forma, serve como estímulo para concluir as atividades que são propostas, e isso se torna um desafio para os alunos. Esse desafio traz sua atenção para a aula, estimula-os a conseguir completar as atividades e proporciona interação entre eles para troca de informações e conhecimentos, em busca de sempre aprender mais.

Para finalizar as reflexões deste trabalho, destacamos o quanto a utilização do *software* VisuAlg pode trazer bons resultados no ensino de algoritmos fazendo uso de cenários provocativos, onde os alunos possam investigar, testar e validar o algoritmo escrito.

## Apêndice A

### Software VisuAlg

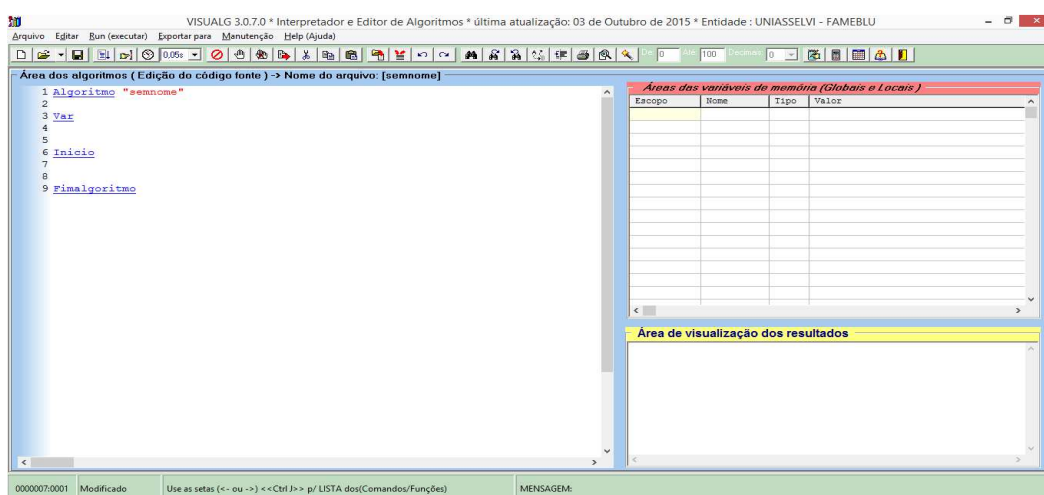
É um programa de fácil utilização para escrever algoritmos no formato de pseudocódigo. Foi criada com propósito didático para alunos escreverem algoritmos utilizando a língua portuguesa. É uma ferramenta que pode ser utilizada para apoio a aprendizagem e permite simular os algoritmos criados (SOUZA, 2013).

Algoritmos podem ser escritos em português utilizando as regras de escrita do tipo pseudocódigo, o VisuAlg permite criar, editar, executar, testar, simular, visualizar valores das variáveis criadas em tempo de execução, o que facilita o entendimento para os alunos (GOMES, 2018).

O *software* permite simular e testar para identificar erros no algoritmo, executando-o passo a passo, e analisar os dados gerados durante e ao fim da execução (BORBA, 2019).

O VisuAlg é gratuito e pode ser feito o *download* no site de repositório de aplicativos (<https://sourceforge.net/projects/visualg30/>). Sua tela é dividida em três áreas principais (Figura 8).

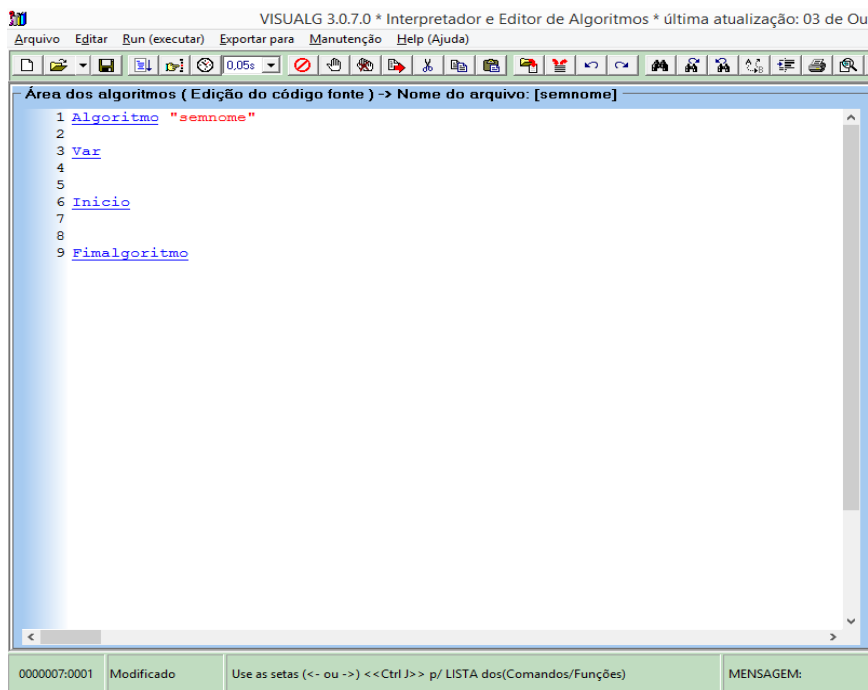
Figura 8 – Tela do *software* VisuAlg



Fonte: do autor (2020).

A área a esquerda da tela (Figura 9) é onde o algoritmo deve ser escrito no formato de pseudocódigo, nesta área todas as variáveis devem ser declaradas logo após a linha que esta escrita de azul “Var” e todos os passos do algoritmo devem ser escritos entre as linhas que estão escrito de azul “Inicio” e “Fimalgoritmo”.

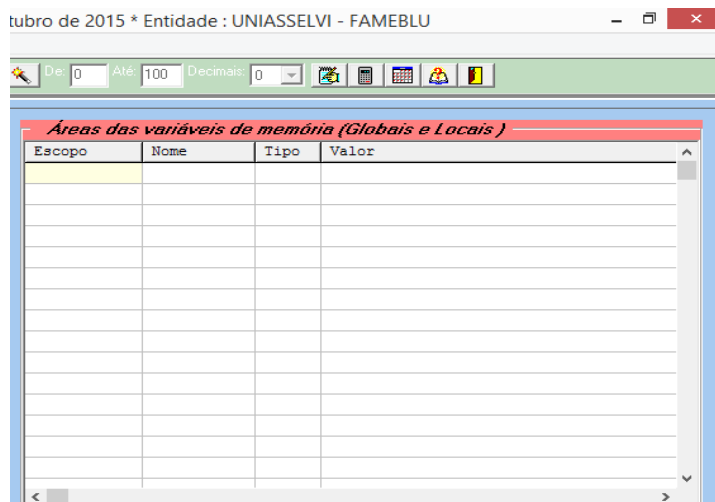
Figura 9 – Área do software VisuAlg para escrever os passos do algoritmo



Fonte: do autor (2020).

A área a direita superior da tela (Figura 10) mostra todas as variáveis declaradas no algoritmo quando o mesmo está em execução e depois que é encerrado, além do nome e tipo da variável, também é mostrado o valor dela em tempo de execução, permitindo acompanhar se os valores estão sendo armazenados corretamente.

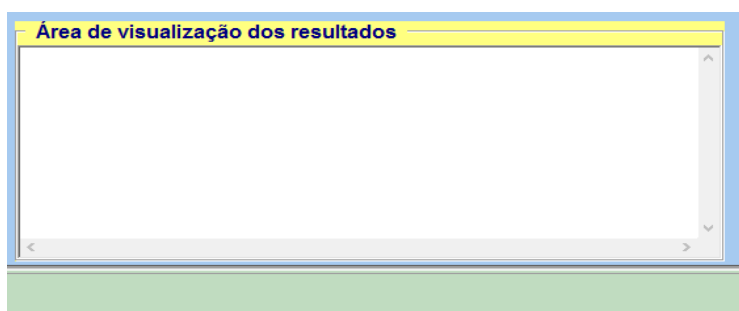
Figura 10 – Área do software VisuAlg para visualizar as variáveis do algoritmo



Fonte: do autor (2020).

A área direita inferior da tela (Figura 11) mostra o passo a passo de execução do algoritmo.

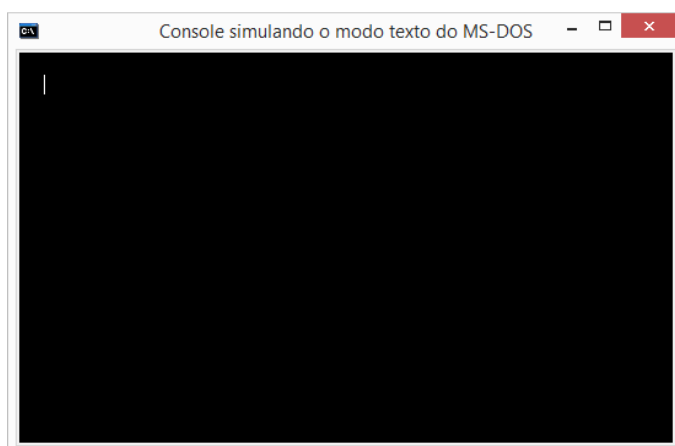
Figura 11 – Área do *software* VisuAlg para visualizar o passo a passo executado



Fonte: do autor (2020).

Quando o algoritmo está sendo executado, uma janela preta (Figura 12) é criada para que os valores possam ser inseridos, quando for necessário.

Figura 12 – Janela de execução do algoritmo no *software* VisuAlg



Fonte: do autor (2020).

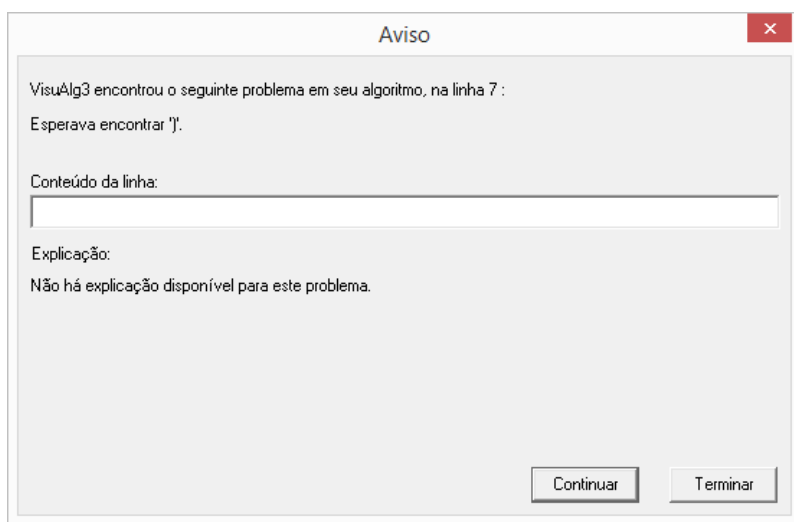
Para executar e simular um algoritmo pressione selecione no menu "Run -> Rodar o Algoritmo" ou pressione a tecla F9 que serve como atalho para a função, então será aberta a janela de execução conforme Figura 12, nesta será apresentado as informações escritas no algoritmo.

Se o algoritmo contiver algum erro de escrita uma janela (Figura 13) será apresentada quando a linha com o erro for executada, nesta janela será mostrado o texto da linha que gerou o erro de execução e um texto informando a causa do erro



que ajudará a analisar o algoritmo para corrigir. As mensagens informativas podem mostrar a falta de caracteres necessários, mas também erros podem ocorrer por escrever caracteres a mais do necessário. Então sempre é necessário conferir a linha que o erro foi gerado e a linha anterior, caso não encontro erros nestas duas linhas, então deve revisar todo o algoritmo.

Figura 13 – Janela de erro de execução do algoritmo no *software* VisuAlg



Fonte: do autor (2020).

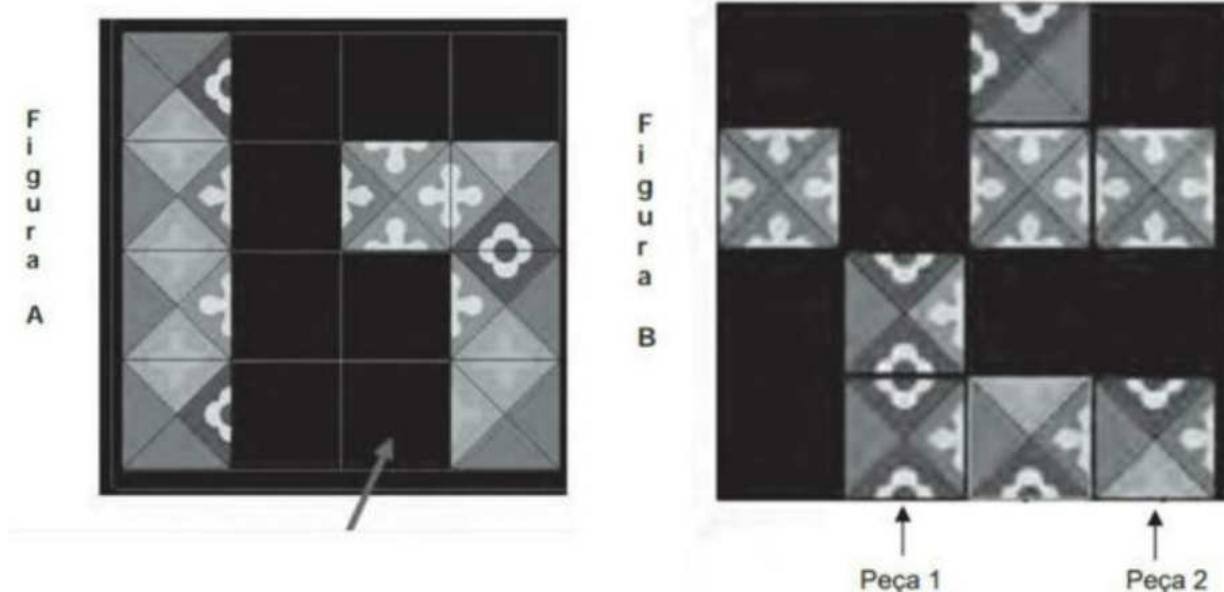
Muitos erros são causados por abrir um parêntese ou chave e não fechar, ou então fechar dois e ter aberto apenas um. Outra situação é escrever comandos faltando caráter ou com caráter a mais.

Quando o erro for difícil de ser encontrado, isso ocorre muito quando o erro não interrompe a execução, mas o resultado é gerado errado. O VisuAlg tem a função de executar o algoritmo passo a passo, que pode ser executado através do menu “Run -> Rodar passo a passo” ou pressionando a tecla F8 que serve como atalho para essa função. Quando executado no modo passo a passo, o VisuAlg só passa para a próxima linha do algoritmo quando a tecla F8 for pressionada, sendo necessário ficar pressionando ela até o algoritmo ser finalizado. Durante o processo de execução passo a passo, pressionando a tecla F8, a linha que estiver sendo executada será marcada de azul, podendo acompanhar os caminhos executados e verificar em tempo de execução os valores calculados para encontrar o erro.

## Apêndice B

### Questionário de Verificação

01) Enem - 2009 - As figuras a seguir exibem um trecho de um quebra-cabeças que está sendo montado. Observe que as peças são quadradas e há 8 peças no tabuleiro da figura A e 8 peças no tabuleiro da figura B. As peças são retiradas do tabuleiro da figura B e colocadas no tabuleiro da figura A na posição correta, isto é, de modo a completar os desenhos.



É possível preencher corretamente o espaço indicado pela seta no tabuleiro da figura A colocando a peça.

- 1 após girá-la  $90^\circ$  no sentido horário.
- 1 após girá-la  $180^\circ$  no sentido anti-horário.
- 2 após girá-la  $90^\circ$  no sentido anti-horário.
- 2 após girá-la  $180^\circ$  no sentido horário.
- 2 após girá-la  $270^\circ$  no sentido anti-horário.

**Justifique como pensou para chegar à resposta:** \_\_\_\_\_

---



---

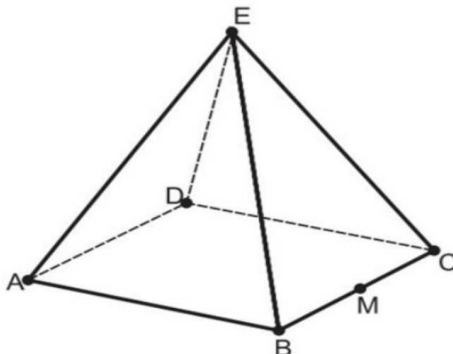


---

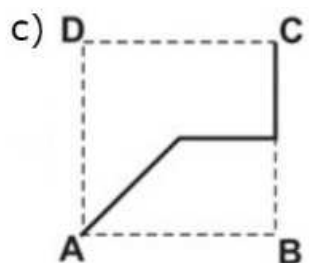
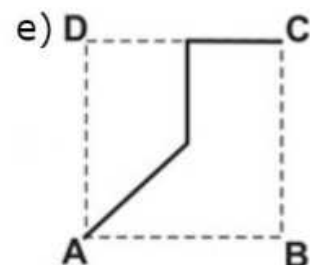
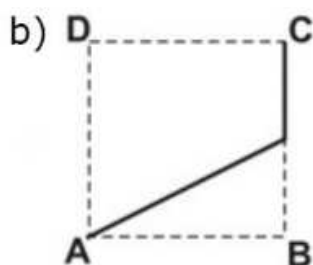
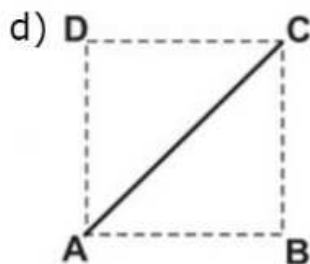
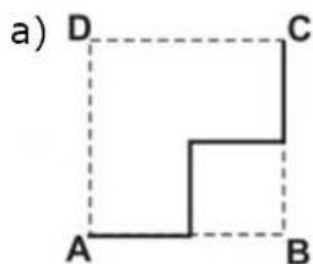


---

02) Enem - 2012 - João propôs um desafio a Bruno, seu colega de classe: ele iria descrever um deslocamento pela pirâmide a seguir e Bruno deveria desenhar a projeção desse deslocamento no plano da base da pirâmide.



O deslocamento descrito por João foi: mova-se pela pirâmide, sempre em linha reta, do ponto A ao ponto E, a seguir do ponto E ao ponto M, e depois de M a C. O desenho que Bruno deve fazer é:



Justifique como pensou para chegar à resposta: \_\_\_\_\_

---



---



---



---

03) Analise os comandos abaixo e escreva os valores que serão mostrados quando solicitado. Considere que os comandos ocorrem de forma sequencial, na ordem que estão dispostos. (Adaptado de CARVALHO, 2007)

- |                        |                        |
|------------------------|------------------------|
| a) $A = 25$            | Qual valor de C? _____ |
| $B = A + 1$            | $C = A + B$            |
| $C = B + 1$            | Qual valor de A? _____ |
| $A = B + 1$            | Qual valor de B? _____ |
| $B = C + 1$            |                        |
| Qual valor de A? _____ |                        |
| Qual valor de B? _____ |                        |
| Qual valor de C? _____ |                        |
- 
- |              |                        |
|--------------|------------------------|
| b) $A = 5$   | c) $A = 11$            |
| $B = 10$     | $B = 22$               |
| $C = A + B$  | $C = A + B$            |
| $B = 5 + C$  | Qual valor de C? _____ |
| $A = B - 10$ | $A = 22$               |
|              | $B = 11$               |
|              | $C = A - B$            |
|              | Qual valor de C? _____ |

**Justifique como pensou para chegar à resposta:** \_\_\_\_\_

---



---



---



---

04) Responda as expressões abaixo como Verdadeiro ou Falso, considere os valores das variáveis dados. (Adaptado de BORBA, 2019)

a = Verdadeiro, b = Verdadeiro, c = Verdadeiro, d = Falso, e = Falso

- a)  $( !d \text{ OU } !e ) \text{ E } ( !a \text{ E } !e )$  \_\_\_\_\_
- b)  $( !a \text{ OU } !c ) \text{ E } !d$  \_\_\_\_\_
- c)  $!( ( a \text{ OU } b \text{ OU } c ) \text{ E } ( d \text{ OU } e ) )$  \_\_\_\_\_
- d)  $( !d \text{ OU } a )$  \_\_\_\_\_
- e)  $!( !( a \text{ E } c ) \text{ E } !( e \text{ E } d ) )$  \_\_\_\_\_
- f)  $( !d \text{ OU } !b \text{ E } a )$  \_\_\_\_\_

05) A partir dos valores atribuídos a cada variável resolva as expressões e descubra se o resultado é Verdadeiro ou Falso. (Adaptado de BORBA, 2019)

$$a = 15, b = 8, c = 53, d = 10, e = 48, f = 24, g = 2$$

- a)  $(b * 10) >= 16$  \_\_\_\_\_
- b)  $(b * b = 16)$  E  $(d * e = 480)$  \_\_\_\_\_
- c)  $((c * b) + 55) >= 130$  \_\_\_\_\_
- d)  $(c * d = 530)$  OU  $(f * 2 = 48)$  E  $(15 * a = 225)$  \_\_\_\_\_

06) O comando ESCREVA deve ser utilizado no algoritmo quando:

- a) precisa apenas perguntar um texto
- b) precisa apenas mostrar um texto
- c) precisa apenas ler um texto
- d) precisa perguntar ou mostrar um texto
- e) precisa perguntar, mostrar ou ler um texto

**Justifique como pensou para chegar à resposta:** \_\_\_\_\_

---

---

---

07) O comando LEIA deve ser utilizado no algoritmo quando:

- a) precisa apenas perguntar um texto.
- b) precisa apenas mostrar um texto.
- c) precisa apenas ler um texto.
- d) precisa perguntar ou mostrar um texto.
- e) precisa perguntar, mostrar ou ler um texto.

**Justifique como pensou para chegar à resposta:** \_\_\_\_\_

---

---

---

---

08) O comando SE deve ser utilizado no algoritmo quando:

- a) precisa apenas mostrar um texto.
- b) precisa apenas ler um texto.
- c) tem uma condição.
- d) precisa ser perguntado algo.
- e) precisa ser escolhido uma opção.

**Justifique como pensou para chegar à resposta:** \_\_\_\_\_

---



---



---

09) Complete o pseudocódigo a seguir com as palavras ESCREVA, LEIA ou SE para que fique correto a resolução do problema descrito.

*Nome: Monitorar Temperatura*

*Tipo:*

*Entrada: temperatura (real)*

*Saída:*

*Auxiliar:*

*Constante: minima = 30.0, maxima = 40.0 (real)*

**Início**

\_\_\_\_\_ (“Informe a temperatura atual.”)

leia(temperatura)

\_\_\_\_\_ (temperatura <= minima) \_\_\_\_\_

*Início*

escreva(“Atenção!!!”)

\_\_\_\_\_ (“Ligue o aquecedor imediatamente!”)

*Fim*

senão se ( (temperatura > mínima) e (temperatura < maxima) ) então

escreva(“Temperatura apropriada!”)

\_\_\_\_\_ se (temperatura > máxima ) então

escreva(“Cuidado!!!! Local interdito, temperatura muito alta!”)

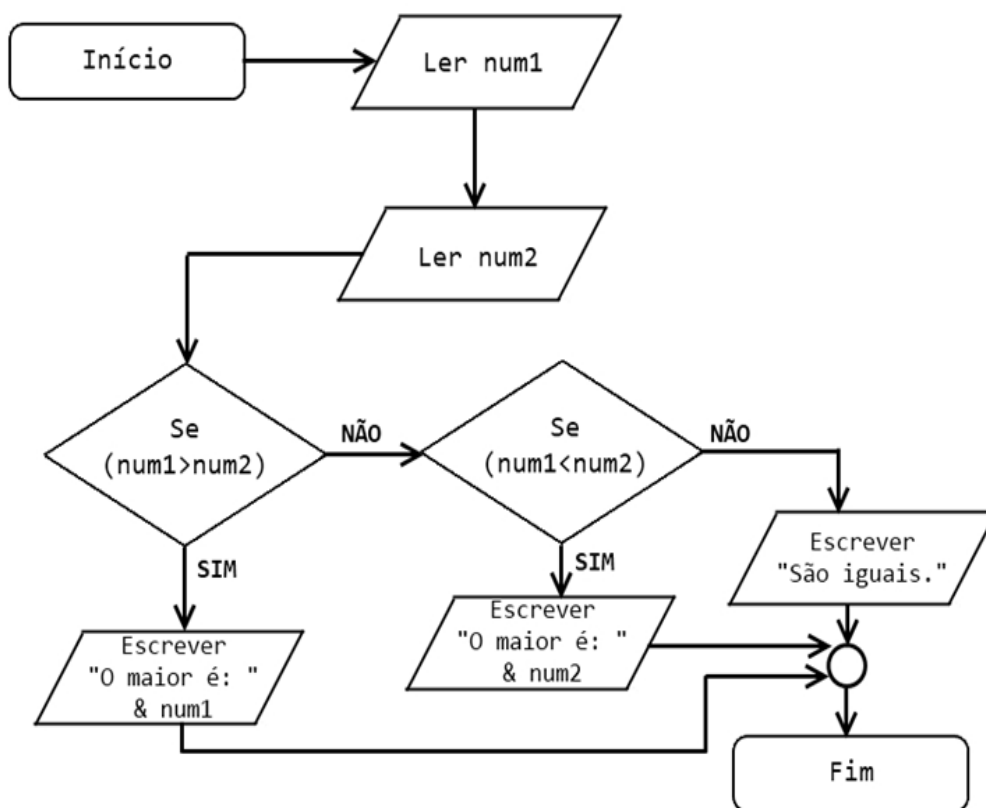
*Fim-se*

*Fim-se*

*Fim-se*

**Fim**

10) Dado o fluxograma abaixo marque a opção que representa o seu problema:



Fonte: <http://tilexmazedo.blogspot.com/2013/04/>

- Mostra se o primeiro número informado é o maior número.
- Mostra se o segundo número informado é o maior número.
- Mostra se o primeiro e o segundo número informado são iguais.
- Mostra qual número informado é o maior número ou se são iguais.
- Todas anteriores são verdadeiras.

11) Complete o pseudocódigo para que fique correto a resolução do problema: A pessoa deverá informar a quantidade de passos necessários para chegar a um destino, em cada passo será informado apenas um caractere que indicará a direção que a pessoa tem que seguir a cada novo passo.

Nome: Roteiro

Tipo:

Entrada: direção (Caractere)

qtdPassos (Inteiro)

Saída:

**Início**

escreva("Informe quantos passos são necessários para chegar ao destino")

---

\_\_\_\_\_ ("Para qual direção ir? D-Direita E-Esquerda F-Frente T-Retorne")

leia( direcao)

se( direcao == 'D' \_\_\_\_\_ direção == 'E' \_\_\_\_\_ direção == 'F' \_\_\_\_\_ direção == 'T' ) então

    escreva("ok")

senão \_\_\_\_\_ ("direção errada")

---

**Fim**



## Referências

ALTHAUS, Neiva. **Os jogos online como ferramentas na resolução de problemas com o uso de tecnologias digitais**. 2016. Dissertação de Mestrado.

BORBA, Fabrício Hartmann. **O Software VisuAlg como recurso didático no ensino da lógica de programação**. 2019. Dissertação de Mestrado.

BOUCINHA, Rafael Marimon. **Aprendizagem do pensamento computacional e desenvolvimento do raciocínio**. Tese (Doutorado em Informática na Educação), 2017.

BRASIL, Ministério da Educação Secretaria de Educação de Educação Média e Tecnológica. **Base Nacional Comum Curricular**. Disponível em: <[http://basenacionalcomum.mec.gov.br/images/BNCC\\_EI\\_EF\\_110518\\_versaofinal\\_s ite.pdf](http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_s ite.pdf)>. Acesso em: 06 dezembro 2019.

BRASIL, Ministério da Educação Secretaria de Educação de Educação Média e Tecnológica. **Parâmetros Curriculares Nacionais: Matemática. Ministério da Educação**. Brasília: SMT/MEC, 1997. Disponível em: <<http://portal.mec.gov.br/seb/arquivos/pdf/livro03.pdf>>. Acesso em: 30 outubro 2019.

GARLET, Daniela; BIGOLIN, Nara Martini; SILVEIRA, Sidnei Renato. **Uma proposta para o ensino de programação de computadores na educação básica**. Departamento de Tecnologia da Informação, Universidade Federal de Santa Maria, RS, 2016.

GOMES, Eduardo Rodrigues. **HelpBlock: uma ferramenta web baseada na biblioteca Blockly para apoio ao ensino de algoritmos**. 2018. Trabalho de Conclusão de Curso.

INACIO, Flamarion Assis Jeronimo. **Ensinando Algoritmo e Lógica de Programação: Modelo Construtivista Auxiliado pelo Scratch**. Dissertação (Mestrado Profissional em Educação Tecnológica), 2016. Disponível em: <[http://virtualif.iftm.edu.br/ERP/MAC/CC/controlador/ctrl\\_pagina\\_curso/carregarArquivo.php?src=260618115334\\_20160502\\_ndash\\_flamarion\\_assis\\_jeronimo\\_inacio\\_ndas\\_h\\_ensino\\_de\\_algoritmo\\_e\\_logica\\_de\\_programacao\\_\\_modelo\\_construtivista\\_auxiliado\\_pelo\\_scratch.pdf](http://virtualif.iftm.edu.br/ERP/MAC/CC/controlador/ctrl_pagina_curso/carregarArquivo.php?src=260618115334_20160502_ndash_flamarion_assis_jeronimo_inacio_ndas_h_ensino_de_algoritmo_e_logica_de_programacao__modelo_construtivista_auxiliado_pelo_scratch.pdf)> Acesso em: 18 junho 2019.

MAIA, Dennys Leite. **Ensinar Matemática com uso de Tecnologias Digitais: Um Estudo a Partir da Representação Social de Estudantes de Pedagogia**. Dissertação (Mestrado Acadêmico em Educação), 2012. Disponível em: <<http://www.uece.br/ppge/dmdocuments/Dennys.pdf>>. Acesso em: 18 junho 2019.

MASINI, Elcie Fortes Salzano; MOREIRA, Marco Antonio. **Aprendizagem Significativa: condições para ocorrência e lacunas que levam a comprometimentos**. 1º Edição. São Paulo: Vetor, 2008.

MOTA, José Carlos. **Da Web 2.0 ao e-Learning 2.0: Aprender na Rede. Dissertação** (Mestrado em Pedagogia do E-learning). Versão *Online*, Universidade Aberta. Portugal. 2009. Disponível em: <<https://repositorioaberto.uab.pt/handle/10400.2/1381>> Acesso em: 22 fevereiro 2019.

SIEMENS, George. **Uma Teoria de Aprendizagem para a Idade Digital**. 2004. Disponível em: <[http://usuarios.upf.br/~teixeira/livros/conectivismo\[siemens\].pdf](http://usuarios.upf.br/~teixeira/livros/conectivismo[siemens].pdf)>. Acesso em: 22 fevereiro 2019.

SOUZA, Marcelo Batista et al. **Uma abordagem metodológica voltada para o ensino-aprendizagem de algoritmos**. RENOTE, v. 11, n. 1, 2013.