



PROVA 03 – P3

Nome: _____

RGA: _____

Data: _____

Orientações Básicas

- Leia atentamente as instruções para cada questão.
- Desligue e coloque sobre a mesa do professor qualquer aparelho eletrônico que esteja utilizando (celulares, *smartphones*, *smartwatches*, *tablets*, etc...), caso seja encontrado com algum destes aparelhos terá automaticamente sua prova anulada.
- A prova deverá ser feita utilizando lápis ou caneta esferográfica de cor azul ou preta.
- Rasuras ou código que gere dupla interpretação poderá acarretar na anulação de sua resposta.
- Não se esqueça de assinar a prova, inserir a data e seu RGA.
- Não converse ou faça gestos que possam prejudicar a sua credibilidade (colas) perante o aplicador da prova.
- Verifique se você possui às n folhas que compõem esta prova (esta página inclusive) e se a impressão esta legível.
 - $n < -$
- Caso encontre algum exercício mal formulado ou com erro você deve descrever onde está e qual é o problema, caso seja confirmado o erro a questão será anulada e os pontos da questão somados a nota final desta prova.
- Esta prova tem duração de quatro horas a partir do início de sua aplicação.

Boa sorte!



1. (2.0) Escreva um módulo que complementa a solução do algoritmo "lê e soma dois inteiros".

```
algoritmo "lê e soma dois inteiros"  
  var  
    x, y : inteiro  
  inicio  
    escreva "Informe o primeiro número:"  
    leia x  
  
    escreva "Informe o segundo número:"  
    leia y  
  
    escreva somaInteiros(x,y)  
fimalgoritmo
```

Exemplo:

Entrada
2
6

Saída
8

R:

```
função inteiro somaInteiros(a,b : inteiro)  
  var  
    soma : inteiro  
  inicio  
    soma <- a + b  
    retorna soma  
fimfunção
```



2. (2.0) Escreva um módulo que complementa a solução do algoritmo "lê e imprime dois reais".

algoritmo "lê e imprime dois reais"

var

z, w : real

inicio

leDoisNumerosReais(z,w)

escreva z, w

fimalgoritmo

Exemplo:

Entrada
-3.4
4.0

Saída
-3.4
4.0

R:

procedimento leDoisNumerosReais(a,b : ref real)

var

inicio

escreva "Informe o primeiro número:"

leia a

escreva "Informe o segundo número:"

leia b

fimprocedimento



3. (2.0) Escreva um módulo que recebe uma matriz de caracteres $A_{n \times m}$ e um caractere x , seu módulo deve verificar quantas vezes o caractere x ocorre na matriz A e retornar essa quantidade.

Exemplo:

Entrada	Saída
$A = \begin{pmatrix} a & b \\ z & a \\ a & j \end{pmatrix}$	3
3	
2	
a	

```
função inteiro contaOcorrencia(M : ref m_caracter, n, m: inteiro, x : caractere)  
  var  
    i, j, cont : inteiro  
  inicio  
    cont <- 0  
    para i de 1 até n faça  
      para j de 1 até m faça  
        se x = M[i,j] então  
          cont <- cont + 1  
        fimse  
      fimpara  
    fimpara  
    retorna cont  
fimfunção
```



4. (2.0) Escreva uma função que dado um valor inteiro natural maior que 0, retorne VERDADEIRO caso este número seja primo ou FALSO em caso contrário. Lembrando que números primos são os números naturais que têm apenas dois divisores diferentes: o 1 e ele mesmo.

Exemplo:

Entrada

1

Saída

FALSO

```
função lógico isPrimo(x: inteiro)  
  var  
    cont, i : inteiro  
  inicio  
    se x = 1 então  
      retorna FALSO  
    senão  
      cont <- 1  
      para i de 1 até x/2 faça  
        se (i MOD x) = 0 então  
          cont <- cont+1  
        fimse  
      fimpara  
  
      se cont = 2 então  
        retorna VERDADEIRO  
      senão  
        retorna FALSO  
      fimse  
    fimse  
fimfunção
```



5. (2.0) Os módulos do algoritmo "lê e calcula notas" apresentam erros de lógica. Reescreva o algoritmo corrigindo estes erros.

```
algoritmo "lê e calcula notas"
  procedimento imprimeFormula( )
    var

    inicio
      escreva "( ( ( (n1+n2) / 2) * 8) + (n3*2) ) / 10"
    fimprocedimento
  procedimento calculaMedia(nota1, nota2, nota3 : ref real)
    var
      media : inteiro
    inicio
      media <- ( ( (n1+n2) / 2) * 8) + (n3*2) ) / 10

      retorna media
    fimprocedimento
  função preencheReg(a : reg_aluno, n : ref cadeia, n1, n2, n3 : real )
    var

    inicio
      a.nome <- n
      a.nota1 <- nota1
      a.nota2 <- nota2
      a.nota3 <- nota3
    fimfunção
  var
    definativo registro
      nome : cadeia
      nota1 : real
      nota2 : real
      nota3 : real
    fimregistro reg_aluno
    aluno : reg_aluno
    n1, n2, n3 : real
    nome : cadeia
  inicio
    escreva "Informe o nome do aluno:"
    leia nome

    escreva "Informe as notas:"
    leia n1, n2, n3

    preencheReg(aluno, nome, n1, n2, n3)
    imprimeFormula( )
    escreva "Média do aluno é: ",calculaMedia(aluno.nota1,aluno.nota2,aluno.nota3)
fimalgoritmo
```



algoritmo "lê e calcula notas"

procedimento imprimeFormula()

var

inicio

escreva "(((n1+n2) / 2) * 8) + (n3*2)) / 10"

fimprocedimento

função real calculaMedia(nota1, nota2, nota3 : ref real)

var

media : real

inicio

media <- (((nota1+nota2) / 2) * 8) + (nota3*2)) / 10

retorna media

fimfunção

procedimento preencheReg(a : ref reg_aluno, n : ref cadeia, n1, n2, n3 : real)

var

inicio

a.nome <- n
a.nota1 <- n1
a.nota2 <- n2
a.nota3 <- n3

fimprocedimento

var

definatipo registro

nome : cadeia
nota1 : real
nota2 : real
nota3 : real

fimregistro reg_aluno

aluno : reg_aluno

n1, n2, n3 : real

nome : cadeia

inicio

escreva "Informe o nome do aluno:"

leia nome

escreva "Informe as notas:"

leia n1, n2, n3

preencheReg(aluno, nome, n1, n2, n3)

imprimeFormula()

escreva "Média do aluno é: ",calculaMedia(aluno.nota1,aluno.nota2,aluno.nota3)

fimalgoritmo