|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Centro Paula SouzaGoverno do Estado de São PauloFatec São Caetano do Sul - Antonio RussoC:\Users\Secretaria01\Documents\selo-2015.png | | | | | | |
| **CURSO SUPERIOR EMANÁLISE E DESENVOLVIMENTO DE SISTEMAS – ADS** | | | | | |
| **AVALIAÇÃO OFICIAL** | | **DISCIPLINA: Algoritmos e Lógica de Programação** | | | **NOTA** |
| **DATA: 08/12/2021**  **N1** **N2 N3** | | **TURMA: 1º ADS - Tarde**  **PROFESSOR: MSc. Flávio Viotti\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | | |  |
| **ALUNO:** |  | | | **RM:** |  |
| **INSTRUMENTO DE AVALIAÇÃO:** **Prova Digital**  **CONDIÇÕES: Individual**  **TEMPO MÁXIMO DE DURAÇÃO: 100 minutos**  **TEMPO MÍNIMO DE DURAÇÃO: 0 minutos** | | | **Visto do Aluno (após correção)** | | |
| **Instruções:**   * A resolução da prova deverá ser de forma organizada. Prova com difícil entendimento ou ilegível na questão a mesma será considerada errada. * **Programas com erros de compilação ou lógica errada serão desconsiderados** * **Cole o programa criado logo abaixo do enunciado da questão, utilize a fonte “courier new”.** * **É permitido o uso das funções já existentes no Portugol Studio** * **Atenção, qualquer indício de cópia parcial ou total de qualquer questão tornará a prova ANULADA, recebendo zero o aluno(a).** | | | | | |

1. (1,0 ponto) Resolva as expressões abaixo assumindo os valores para P = Verdadeiro e Q = Falso. Será preciso que o aluno apresente todos os passos até chegar no resultado final.
   1. P e Q
   2. Q ou P e Q
   3. (P e Q) ou (P e não Q)
   4. não (P e P) e (P e Q)
   5. não ((não P ou não Q) ou não (P ou Q))
2. (1,0 ponto) Sejam as variáveis a=10, b=3 e c=-2, resolva as seguintes operações levando em conta os valores aqui fornecidos para cada uma das operações

|  |  |  |  |
| --- | --- | --- | --- |
|  | a | b | c |
| a++ |  |  |  |
| --b |  |  |  |
| a += 4 |  |  |  |
| c \*= 2 |  |  |  |
| c = ++a |  |  |  |
| b = c-- |  |  |  |

1. (1,0ponto) Faça um algoritmo cujo objetivo é informar se um determinado vetor de inteiros está ordenado em ordem decrescente. O nome do vetor é números. Se o vetor estiver em ordem, exiba “Vetor em ordem crescente”. Lembre-se, um vetor está em ordem decrescente se e somente se o elemento e maior que o elemento

funcaoinicio()

{

inteironumeros[100], n

para (n=0; n < 100; n++) {

leia(numeros[n])

}

//Verifique aqui se o vetor está ordenado em ordem decrescente.

logico ordenado = verdadeiro

para (n = 0; n < 99; n++) {

se (numeros[n] <= numeros[n+1]) {

ordenado = falso

para

}

}

se (ordenado) {

escreva("Vetor em ordem decrescente")

} senao {

escreva("Vetor não está em ordem decrescente")

}

}

}

1. (1,0 ponto) Joaozinho precisa fazer um programa que solicita a nota de um determinado aluno, o programa só pode ir para a frente se essa nota estiver entre 0 e 10 (inclusive). Complete o programa abaixo, acrescentando somente a rotina que faça essa validação

funcaoinicio() {

real nota

escreva(“Digite uma nota (0 a 10):”)

//rotina de entrada e validação de dados.

enquanto (nota < 0 OU nota > 10) {

escreva("Nota inválida. Digite novamente (0 a 10): ")

leia(nota)

}

// Continuar com o restante do programa...

}

1. (2,0 pontos) Para uma turma de 45 alunos, construir um algoritmo que determine armazene em uma matriz a idade e altura desses alunos. Portanto você deverá criar uma matriz 45 x 2, pois são 45 alunos e duas informações para cada um deles. Ao final apresente também:
   1. A idade média dos alunos com altura menor que 1.70m;
   2. A altura média dos alunos com mais de vinte anos;

funcao inicio() {

real matrizAlunos[45][2]

real somaIdadeMenor170 = 0

inteiro contadorMenor170 = 0

real somaAlturaMais20 = 0

inteiro contadorMais20 = 0

para (inteiro i = 0; i < 45; i++) {

escreva("Informe a idade do aluno ", i+1, ": ")

leia(matrizAlunos[i][0])

escreva("Informe a altura do aluno ", i+1, ": ")

leia(matrizAlunos[i][1])

se (matrizAlunos[i][1] < 1.70) {

somaIdadeMenor170 = somaIdadeMenor170 + matrizAlunos[i][0]

contadorMenor170 = contadorMenor170 + 1

}

se (matrizAlunos[i][0] > 20) {

somaAlturaMais20 = somaAlturaMais20 + matrizAlunos[i][1]

contadorMais20 = contadorMais20 + 1

}

}

real mediaIdadeMenor170 = 0

se (contadorMenor170 > 0) {

mediaIdadeMenor170 = somaIdadeMenor170 / contadorMenor170

}

real mediaAlturaMais20 = 0

se (contadorMais20 > 0) {

mediaAlturaMais20 = somaAlturaMais20 / contadorMais20

}

escreva("Idade média dos alunos com altura menor que 1.70m: ", mediaIdadeMenor170)

escreva("Altura média dos alunos com mais de vinte anos: ", mediaAlturaMais20)

}

1. (1,5 pontos) Ler um vetor W de 1000 elementos, depois ler um valor V. Contar e escrever quantas vezes o valor V ocorre no vetor W e escrever também em que posições (índices) do vetor W o valor V aparece. Caso o valor V não ocorra nenhuma vez no vetor W, escrever uma mensagem informando isto.

funcao inicio() {

inteiro vetorW[1000]

inteiro valorV

inteiro contador = 0

inteiro posicoes[1000]

inteiro contadorPosicoes = 0

para (inteiro i = 0; i < 1000; i++) {

escreva("Informe o valor para o elemento ", i+1, " do vetor W: ")

leia(vetorW[i])

}

escreva("Informe o valor V: ")

leia(valorV)

para (inteiro i = 0; i < 1000; i++) {

se (vetorW[i] == valorV) {

contador = contador + 1

posicoes[contadorPosicoes] = i

contadorPosicoes = contadorPosicoes + 1

}

}

se (contador > 0) {

escreva("O valor ", valorV, " ocorre ", contador, " vez(es) no vetor W.")

escreva("Posicoes no vetor W em que o valor V aparece: ")

para (inteiro i = 0; i < contadorPosicoes; i++) {

escreva(posicoes[i], " ")

}

} senao {

escreva("O valor ", valorV, " não ocorre nenhuma vez no vetor W.")

}

}

1. (1,0 ponto) Faça uma função chamada “verificaPar(inteiro num) logico” que irá receber um número qualquer inteiro e devolver “verdadeiro ou falso” informando “verdadeiro” se o número passado é par ou “falso” caso contrário. Não precisa fazer a rotina principal do programa.

funcao verificaPar logico

{

se (num % 2 == 0) {

retornar verdadeiro

} senao {

retornar falso

}

}

1. (1,5 ponto) Faça um programa que solicite do usuário a entrada de 1000 números. Esses números devem ser armazenados obrigatoriamente em um vetor chamado “entrada”. Crie um arquivo texto chamado “saída.txt” que deverá ser gravado neste arquivo na primeira linha a quantidade de valores pares existentes no vetor, e na segunda linha a quantidade de valores ímpares existentes no vetor.

funcao inicio()

{

inteiro entrada[1000]

para (inteiro i = 0; i < 1000; i++) {

escreva("Informe o valor ", i+1, ": ")

leia(entrada[i])

}

inteiro pares = 0

inteiro impares = 0

para (inteiro i = 0; i < 1000; i++) {

se (verificaPar(entrada[i])) {

pares = pares + 1

} senao {

impares = impares + 1

}

}

arquivo saida = abrirArquivo("saida.txt", "w")

escrever(saida, pares)

escrever(saida, "\n")

escrever(saida, impares)

fecharArquivo(saida)

}

funcao verificaPar logico

{

se (num % 2 == 0) {

retornar verdadeiro

} senao {

retornar falso

}

}