**Exercício de Fixação**

1. Determine o resultado lógico das expressões mencionadas, assinalando se são verdadeiras ou falsas. Considere para as respostas os seguintes valores: x=1, a=3,b=5,c=8 e d=7.
2. not (x>3) = (Falso)
3. (x<1) and not (b>d) = (Falso)
4. not (d<0) and (c>5) =(Verdadeiro)

d) not (x >3) or (c<7) =(Verdadeiro)

e) (a>b) or (c>b) = (Verdadeiro)

f) (x>=2) = (Falso)

g) (x<1) and (b>=d) = (Falso)

h) (d<0) or (c>5) (Verdadeiro)

1. not (d>3) or not (b<7) (Verdadeiro)
2. (a>b) or not (c>b) (Verdadeiro)
3. Desenvolva os algoritmos em ruby dos seguintes problemas computacionais:
4. Efetuar a leitura de um valor numérico inteiro positivo ou negativo representado pela variável N e apresenta o valor lido como sendo positivo. Dica: se o valor lido for menor que zero, ele deve ser multiplicado por -1.

puts "Informe um valor inteiro"

valor = gets.chomp.to\_i

if(valor < 0 )

    valor \*= (-1)

end

puts valor

1. Ler um numérico inteiro e apresentar uma mensagem informando se o valor fornecido é par ou impar.

puts "Informe um valor inteiro"

valor = gets.chomp.to\_i

if(valor < 0 )

    valor \*= (-1)

end

rest = valor % 2

if(rest === 0)

    puts "O numero informado #{valor} é par"

else

    puts "O numero informado #{valor} é ímpar"

end

1. Efetuara leitura de um valor numérico inteiro que esteja na faixa de valores de 1 até
2. O programa deve apresentar a mensagem "O valor está na faixa permitida", caso o valor informado esteja entre 1 e 9. Se o valor estiver fora da faixa, o programa deve apresentar a mensagem "O valor está fora da faixa permitida".

puts "Informe um valor inteiro entre 1 e 9"

valor = gets.chomp.to\_i

if (valor >= 0 and valor < 10)

    puts "O Valor informado está na faixa permitida!"

else

    puts "O Valor informado não está na faixa permitida!"

end

1. Fazer a leitura de um valor numérico inteiro qualquer e apresentá-lo caso não seja maior que 3. Dica: para a solução deste problema utilize o operador lógico de negação.

puts "Informe um valor inteiro"

valor = gets.chomp.to\_i

if( not valor > 3 )

    puts "O Valor é menor que 3 => #{valor}"

else

    puts "O Valor é maior que 3"

end

1. Efetuar a leitura de um nome(variável NOME) e o sexo (variável SEXO) de uma pessoa e apresentar como saída uma das seguintes mensagens: "Ilmo Sr.", caso seja informado o sexo masculino(utilizar como valor o caractere "M"), ou "Ilma. Sra.", caso seja informado o sexo feminino (utilizar como valor o caractere "F"). Após a mensagem de saudação, apresentar o nome informado. O programa deve, após a entrada do sexo, verificar primeiramente se o sexo fornecido é realmente válido, ou seja, se é igual a "M" ou "F". Não sendo essa condição verdadeira, o programa deve apresentar a mensagem "Sexo informado inválido".

puts "Informe seu nome"

nome = gets.chomp

puts "Informe seu sexo(F ou M)"

sexo = gets.chomp

if(sexo == 'F' || sexo == 'f' ||sexo == 'Feminino' || sexo == 'feminino')

    puts "Sra. #{nome}"

elsif(sexo == 'M' || sexo == 'm' ||sexo == 'Masculino' || sexo == 'masculino')

    puts "Sr. #{nome}"

else

    puts "Sexo invalido"

end

1. Efetuar a leitura de três valores inteiros desconhecidos representados pela variáveis A,B e C. Somar os valores fornecidos e apresentar o resultado somente se for maior ou igual a 100.

puts "Informe um valor"

valor1 = gets.chomp.to\_i

puts "Informe um segundo valor"

valor2 = gets.chomp.to\_i

puts "Informe um terceiro valor"

valor3 = gets.chomp.to\_i

somaTotal = valor1+valor2+valor3

if(  somaTotal >= 100 )

    puts "O Resultado da soma é #{somaTotal}"

else

    puts "A Soma é menor que 100"

end

1. Ler um número inteiro qualquer e multiplicá-lo por dois. Apresentar o resultado da multiplicação somente se o resultado for maior que 30.

puts "Informe um valor"

valor1 = gets.chomp.to\_i

puts "Informe um segundo valor"

valor2 = gets.chomp.to\_i

multiTotal = valor1\*valor2

if( multiTotal > 30 )

    puts "O Resultado da dá multiplicação é #{multiTotal}"

else

    puts "A multiplicação é menor que 30"

end

1. Elaborar um programa que calcule e apresente o valor do volume de uma caixa retangular, utilizando a fórmula volume = comprimento \* largura \* altura.

puts "Informe um o comprimento da caixa"

comprimento = gets.chomp.to\_f

puts "Informe a altura da caixa"

altura = gets.chomp.to\_f

puts "Informe um largura valor"

largura = gets.chomp.to\_f

volume = comprimento \* largura \* altura

puts "o volume da caixa é #{volume}"

1. Elaborar um programa que calcule e apresente o valor do resultado da área de uma circunferência. O programa deve solicitar a entrada do valor do raio da circunferência.

Para execução deste problema utilize a fórmula area = 3.14159265 \* raio \*\* 2

puts "Informe o valor: "

raio = gets.chomp.to\_f

area = 3.14159265

resultado = area \* raio \*\* 2

puts(resultado)

1. Construir um programa que calcule e apresente em metros por segundo o valor da velocidade de um projétil que percorre um distância em quilômetros a um espaço de tempo em minutos. Utilize a fórmula velocidade = (distancia \*1000) / (tempo \* 60).

puts "Informe a distancia percorrida"

distancia = gets.chomp.to\_f

puts "Informe o tempo percorrida"

tempo = gets.chomp.to\_f

velocidade  = (distancia \*1000) / (tempo \* 60)

puts "A velocidade do projetio é #{velocidade}"

1. Elaborar um programa de computador que calcule e apresente o valor do volume de uma esfera. Utilize a fórmula volume = (4/3) \* 3.14159 \* (raio \*\*3).

puts = "Informe o valor:"

raio = gets.chomp.to\_f

volume =  (4/3) \* 3.14159 \* (raio \*\*3)

puts(volume)

1. Construir um programa que leia um valor numérico inteiro e apresente como resultado os seus valores sucessor e antecessor.

puts "Informe o valor: "

valor = gets.chomp.to\_i

antecessor = valor - 1

sucessor = valor + 1

puts "o antecessor é #{antecessor} e o sucessor é #{sucessor}"