1. Crie um algoritmo que permita calcular a área de diferentes figuras geométricas: um círculo, um quadrado e um retângulo. O programa deve possuir três funções diferentes, cada uma para calcular a área de uma figura. O usuário deve ser capaz de escolher qual figura deseja calcular a área e fornecer os parâmetros necessários.

•A função calcularAreaCirculo deve receber o raio como parâmetro e retornar a área do círculo.

•A função calcularAreaQuadrado deve receber o lado como parâmetro e retornar a área do quadrado.

•A função calcularAreaRetangulo deve receber a largura e a altura como parâmetros e retornar a área do retângulo.

programa

{

  funcao inicio()

  {

    cadeia figura

    real raio, lado, largura, altura

    enquanto (figura != "Círculo" e figura != "Quadrado" e figura != "Retângulo")

    {

      escreva("Você deseja calcular a área de qual figura geométrica? (Círculo, Quadrado ou Retângulo) ")

      leia(figura)

    }

    se (figura == "Círculo")

    {

      escreva("\nDigite o raio: ")

      leia(raio)

      real areaC = calcularAreaCirculo(raio)

      escreva("\nA área do Círculo é: "+areaC+"")

      escreva("\n")

    }

    senao se (figura == "Quadrado")

    {

      escreva("\nDigite o lado: ")

      leia(lado)

      real areaQ = calcularAreaQuadrado(lado)

      escreva("\nA área do Quadrado é: "+areaQ+"")

      escreva("\n")

    }

    senao

    {

      escreva("\nDigite a largura: ")

      leia(largura)

      escreva("Digite a altura: ")

      leia(altura)

      real areaR = calcularAreaRetangulo(largura, altura)

      escreva("\nA área do Retângulo é: ", areaR)

      escreva("\n")

    }

  }

  funcao real calcularAreaCirculo (real raio)

  {

    real areaC = 3.14 \* (raio \* raio)

    retorne areaC

  }

  funcao real calcularAreaQuadrado(real lado)

  {

    real areaQ = lado \* lado

    retorne areaQ

  }

  funcao real calcularAreaRetangulo(real largura, real altura)

  {

    real areaR = largura \* altura

    retorne areaR

  }

}

2. Desenvolva um algoritmo que verifique se um número inteiro é primo. Para isso, crie uma função que receba um número inteiro como parâmetro e retorne verdadeiro se o número for primo, ou falso caso contrário. O algoritmo principal deve solicitar ao usuário um número, chamar a função, e então exibir se o número é primo ou não.

•A função ehPrimo deve receber um número inteiro e retornar um valor lógico (verdadeiro ou falso).

•A função deve verificar se o número é divisível apenas por 1 e por ele mesmo.

programa

{

  funcao inicio()

  {

    inteiro num

    escreva("Digite um número: ")

    leia (num)

    logico ehPrimo(num)

    se (ehPrimo == falso)

    {

      escreva("\nO número não é primo.")

      escreva("\n")

    }

    senao

    {

      escreva("\nO número é primo.")

      escreva("\n")

    }

  }

  funcao logico ehPrimo (inteiro num)

  {

    para (inteiro i = num - 1; i >= 2; i--)

    {

      se (num % i == 0)

      {

        retorne falso

      }

    }

    retorne verdadeiro

  }

}

3. Elabore um algoritmo que converta temperaturas entre Celsius, Fahrenheit e Kelvin. O algoritmo deve conter três funções que realizam as conversões necessárias. O usuário deve poder escolher a conversão que deseja realizar e fornecer o valor da temperatura.

•A função celsiusParaFahrenheit deve receber a temperatura em Celsius e retornar a temperatura em Fahrenheit.

•A função fahrenheitParaKelvin deve receber a temperatura em Fahrenheit e retornar a temperatura em Kelvin.

A função kelvinParaCelsius deve receber a temperatura em Kelvin e retornar a temperatura em Celsius.