

Exercícios - Funções

- 1) Implemente, na linguagem C, duas funções para calcular a área e o perímetro de uma circunferência. Em seguida utilize essas funções em um programa que receba, via `scanf()`, o raio de uma circunferência, e calcule sua área e perímetro. Seu programa deve definir o valor de π utilizando a diretiva `#define`. Lembre-se que o perímetro de uma circunferência é dado por $2 \cdot \pi \cdot r$, e sua área é $\pi \cdot r^2$. Segue um exemplo de entrada e saída. Os valores sublinhados foram fornecidos pelo usuário.

```
Forneça o raio da circunferencia: 5
Area: 78.500000
Perimetro:31.400000
```

- 2) Crie uma função na linguagem C que receba 2 números inteiros e retorne o maior valor. Em seguida, **usando a função criada**, crie uma segunda função que receba 3 números inteiros e retorne o maior deles.
- 3) Crie uma função na linguagem C chamada `dado()` que retorna um número aleatório entre 1 e 6. Em seguida, use a função implementada para simular o lançamento de um dado 1 milhão de vezes, contando quantas vezes cada número saiu. Seu programa deve mostrar, como saída, quantas vezes cada número apareceu na simulação. Como curiosidade, execute o programa para diferentes números de lançamentos de dados e verifique como o resultado se comporta!
- 4) Escreva uma função que recebe um inteiro positivo m e retorna 1 se m é primo e 0 caso contrário. Usando essa função, escreva um programa que leia um inteiro não negativo n e imprima a soma dos n primeiros números primos.
- 5) Um número a é dito *permutação* de um número b se os dígitos de a formam uma permutação dos dígitos de b . Exemplo: 5412434 é uma permutação de 4321445, mas não é uma permutação de 4312455. Obs.: Considere que o dígito 0 (zero) não aparece nos números.
- Faça uma função de nome `conta_digitos()` que, dados um inteiro n e um inteiro d , $0 < d \leq 9$, devolve quantas vezes o dígito d aparece em n .
 - Usando a função do item anterior, faça um programa que lê dois inteiros positivos a e b e responda se a é permutação de b .