

# Passagem de Parâmetros

---

ESTRUTURAS DE DADOS APLICADAS

PROF. FELIPE A. PRZYSIADA

# Introdução

---

➤ Uma função comunica-se com o "mundo externo" de duas maneiras: mudando os valores que são globais à função ou através de passagem e retorno de valores. A informação pode ser movida da/para função de duas maneiras:

- Uma cópia da variável inteira é feita;
- Somente o endereço da variável é transferido pelos limites;

# Introdução

---

- Passagem por valor: É feita uma cópia do argumento (ou variável), esta pode ser usada e alterada dentro da função sem afetar a variável da qual ela foi gerada;
- Passagem por referência: Um ponteiro é passado como parâmetro, sendo assim um endereço manipulado na função destino;
- Quando uma função precisa ser capaz de alterar os valores das variáveis usadas como argumentos, os parâmetros precisam ser explicitamente declarados como ponteiros;

# Vantagens

---

- A primeira é simplesmente pela eficiência;
- Sabendo-se que a variável será somente lida, então, geralmente é mais rápido passar o endereço quando se usam variáveis/objetos;
- O tamanho do endereço é sempre o mesmo, assim não há problema envolvendo cópia e inicialização;

# Vantagens

---

- A segunda razão para passar um endereço é manipular fisicamente a variável para o qual o endereço aponta, isto é, pode-se alterar uma variável que é externa a uma função;
- Normalmente, os resultados de uma chamada de função devem ser expressos pelo valor de retorno da função, mas existem situações em que isso não é satisfatório. Por exemplo, a função pode precisar alterar mais uma variável;
- Se os endereços são passados, a função pode afetar diretamente qualquer número de variáveis na função chamadora;

# Exemplo 1

---

- Crie uma função que some dois números inteiros;

# Exemplo 2

---

- Faça uma função que receba uma string que possa ter uma quantidade variável de caracteres;

# DÚVIDAS, PERGUNTAS, QUESTIONAMENTOS

---

