



SUMÁRIO

1. Objetivo	1
2. Base Teórica	1
2.1. Dividir para Conquistar: Módulos	1
2.2. Funções	1
2.3. Procedimentos	1
2.4. Passagem de Argumentos para Parâmetros	2
3. Seqüência da Prática	3
4. Exercícios	3
4.1. Maior Valor entre Três Números	3
4.2. Área e Perímetro de Círculos	3
4.3. Troca de Valores entre Variáveis	3

1. OBJETIVO

Introduzir o uso de funções/procedimentos em algoritmos/programas.

2. BASE TEÓRICA

2.1. Dividir para Conquistar¹: Módulos

Algoritmos podem ser simples, descritos por algumas poucas linhas em português estruturado, ou maiores e mais complexos, constituídos de vários comandos e estruturas de controle. Em tais situações, técnicas para dividir o algoritmo em vários trechos menores, ou **módulos**, porém integrados, são utilizadas para melhor administrar o tamanho e a complexidade do algoritmo. É mais fácil trabalhar com vários trechos reduzidos e menos complexos do que com um algoritmo inteiro, complexo e monolítico.

São dois os tipos de módulos: as funções e os procedimentos. Eles devem ser sempre colocados antes do algoritmo/programa principal.

2.2. Funções

Uma função é um algoritmo construído para resolver um problema específico, fornecendo como resultado um único valor, geralmente calculado a partir de dados de entrada. O resultado é colocado/usado exatamente no local da chamada da função.

Algoritmo 1

Português Estruturado	C
<pre>real soma(real:a,b) // a e b: parâmetros. inicio soma ← a+b; fim. inicio real: x,y; imprima("Digite x e y: "); leia(x,y); // x e y: argumentos da função soma. imprima(x, " + ",y, " = ",soma(x,y)); fim.</pre>	<pre>#include "stdio.h" float soma(float a, float b){ // a e b: parâmetros. return a+b; } int main(){ float x,y; printf("Digite x e y: "); scanf("%f %f", &a, &b); // x e y: argumentos da função soma. printf("%.1f + %.1f = %.1f",x,y,soma(x,y)); }</pre>
Resultado Digite x e y: 2 3 2 + 3 = 5	Resultado Digite x e y: 2 3 2.0 + 3.0 = 5.0

2.3. Procedimentos

Procedimentos, também conhecidos como sub-rotinas, são operacionalmente idênticos às funções a menos do valor de retorno, que não existe nesse caso. O procedimento realiza alguma ação que não envolve retornar um valor como, por exemplo, uma impressão.

Algoritmo 2

Português Estruturado	C
<pre>real soma(real:a,b) inicio soma ← a+b; fim. proc imp_xy(real: x,y) inicio imprima("x = ",x); imprima("y = ",y); fim. inicio real: x,y; imprima("Digite x e y: "); leia(x,y); imp_xy(x,y); imprima("x + y = ",soma(x,y)); fim</pre>	<pre>#include "stdio.h" float soma(float a, float b){ return a+b; } void imp_xy(float x, float y){ printf("x = %.1f\n",x); printf("y = %.1f\n",y); } int main(){ float x,y; printf("Digite x e y: "); scanf("%f %f", &x, &y); imp_xy(x,y); printf("x + y = %.1f",soma(x,y)); }</pre>
Resultado Digite x e y: 2 3 x = 2 y = 3 x + y = 5	Resultado Digite x e y: 2 3 x = 2.0 y = 3.0 x + y = 5.0

2.4. Passagem de Argumentos para Parâmetros

Na **passagem por valor**, cópias dos argumentos (valores usados na chamada da função) são passadas para os parâmetros da função (valores recebidos na função). Assim, se o parâmetro da função é modificado, nada acontece com o argumento. Neste tipo de passagem os argumentos podem ser variáveis (x), constantes (3), expressões (x+3) ou funções (sen(x)).

Algoritmo 3

Português Estruturado	C
<pre>real quadrado(real: a) // a é parâmetro passado inicio a=a*a; quadrado ← a; fim. inicio real: a,q; imprima("Digite a: "); leia(a); q ← quadrado(a); // a é argumento. imprima("a^2 = ",q); imprima("a = ",a); fim.</pre>	<pre>#include "stdio.h" float quadrado(float a){ // a é parâmetro passado a=a*a; return a; } int main(){ float a,q; printf("Digite a: "); scanf("%f", &a); q=quadrado(a); // a é argumento. printf("a^2 = %.1f\n",q); printf("a = %.1f",a); }</pre>
Resultado Digite a: 3 a^2 = 9 a = 3	Resultado Digite a: 3 a^2 = 9.0 a = 3.0

Na **passagem por referência**, tanto argumentos quanto parâmetros compartilham o mesmo local de memória: se o parâmetro for alterado, o argumento também o será. Dessa forma, neste tipo de passagem, os argumentos só podem ser variáveis! No português estruturado e no C, esse tipo de passagem é indicado colocando-se o símbolo & na frente do parâmetro.

¹ Uma técnica atribuída ao general e imperador romano Júlio César: dividir para conquistar. Obviamente, em outro contexto.

Algoritmo 4

Português Estruturado	C
<pre> real quadrado(real: &x) // x é parâmetro passado inicio x=x*x; quadrado ← x; fim. inicio real: a,q; imprima("Digite a: "); leia(a); q ← quadrado(a); // a é argumento. imprimal("a^2 = ",q); imprima("a = ",a); fim. </pre>	<pre> #include "stdio.h" float quadrado(float &x){ // x é parâmetro passado x=x*x; return x; } int main(){ float a,q; printf("Digite a: "); scanf("%f", &a); q=quadrado(a); // a é argumento. printf("a^2 = %.1f\n",q); printf("a = %.1f",a); } </pre>
Resultado Digite a: 3 a^2 = 9 a = 9	Resultado Digite a: 3 a^2 = 9.0 a = 9.0

3. SEQÜÊNCIA DA PRÁTICA

- Implementar em C o [algoritmo 2](#) acima e testá-lo.
- Fazer os exercícios

4. EXERCÍCIOS

4.1. Maior Valor entre Três Números

Escrever um algoritmo em português estruturado que leia um trio de valores e imprima o valor do maior elemento do trio. Para obter o maior elemento do trio utilizar a função MaiorNumero. Implementar esse algoritmo em C e testá-lo.

4.2. Área e Perímetro de Círculos

Escrever um algoritmo em português estruturado que leia 5 valores que representam raios de círculos e os guarde em um vetor. Para cada valor informado, imprimir a área e o seu perímetro. Para obter o valor da área do círculo o algoritmo deverá chamar a função AreaCirculo. Para obter o valor do seu perímetro o programa deverá chamar a função PerCirculo. Implementar esse algoritmo em C e testá-lo.

4.3. Troca de Valores entre Variáveis

Construa um algoritmo que leia duas variáveis A e B inteiras, imprima os valores lidos, troque os valores das variáveis entre elas através da chamada a um procedimento com passagem por referência e imprima os valores de A e B novamente (que deverão estar trocados). Implementar esse algoritmo em C e testá-lo.

Dica. Imagine que você tenha um copo A com Guaraná e um copo B com Coca-Cola, e você deseja passar a Coca-Cola para o copo A e o Guaraná para o copo B. Como você faria? A resposta a essa pergunta indicará como realizar a troca de valores entre variáveis no procedimento.