

# Terceira Lista de Exercícios de Laboratório

SCC0121 - Introdução à Programação

Prof. Alneu de Andrade Lopes

Estagiário PAE: Nils Ever Murrugarra Llerena

24 de maio de 2010

## 1 Ponteiros

1. Usando ponteiros, faça uma função que troque os valores de duas variáveis inteiras.
2. Escreva uma função *mm* que receba um vetor inteiro  $v[0..n - 1]$  e os endereços de duas variáveis inteiras, digamos *min* e *max*, e deposite nessas variáveis o valor de um elemento mínimo e o valor de um elemento máximo do vetor. Escreva também uma função *main* que use a função *mm*.
3. Usando ponteiros, escreva uma função que determine se uma string é ou não um palíndromo (ou seja, se o inverso da string é igual a ela). Escreva um programa para testar a função. Por exemplo: anilina, reviver, etc.
4. Usando ponteiros, dado 2 vetores *A* e *B* de números inteiros, determinar os elementos comuns nos vetores *A* e *B*.

## 2 Vetores

Desenvolver algoritmos e escrever as correspondentes funções em C para os seguintes problemas:

1. Faça um programa que leia 10 números reais e preencha um vetor com estes números. Feito isso, calcule:
  - A média aritmética destes valores.
  - Imprima o maior valor contido no vetor.
  - Imprima o menor valor contido no vetor.
2. Dado um polinômio de grau 4:  $P(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4$  representado por um vetor  $v = [a_0 \ a_1 \ a_2 \ a_3 \ a_4]$ . Dado o valor do  $x$ ,

calcular o valor de  $P(x)$ . Por exemplo,  $P(x) = 2 + 3x + x^2 + x^4$  é representado no vetor  $v = [2 \ 3 \ 1 \ 0 \ 1]$ . E com o valor de  $x = 2$ ,  $P(x) = 28$ .

3. Dado 2 vetores *A* e *B* de números inteiros, determinar os elementos comuns nos vetores *A* e *B*.

## 3 Matrizes

Desenvolver algoritmos e escrever as correspondentes funções em C para os seguintes problemas:

1. Dada uma matriz  $N * N$  de inteiros, exiba:
  - A diagonal principal.
  - A diagonal secundária.
2. Dada uma matriz de  $m * n$  e um vetor de tamanho  $n$ , encontrar a linha da matriz que igual ao vetor.
3. Jogo da velha. Para jogar ao jogo da velha se utiliza uma matriz de  $3 * 3$ . Uma posição ocupada na matriz pelo primeiro jogador tem o número 1, a posição ocupada pelo segundo jogador tem o número 2 e as posições livres tem o número 0.
  - Faça uma função que indique se o jogador 1 fez três marcações na mesma linha.
  - Faça uma função que indique se o jogador 1 fez três marcações na mesma coluna.
  - Faça uma função que indique se o jogador 1 fez três marcações em uma das diagonais.

- Usando as funções feitas, faça uma função que indique se o jogador 1 fez três marcações na mesma linha, coluna ou em uma das diagonais.

## 4 Alocação Dinâmica

1. Escreva um programa para armazenar num array um certo número de inteiros. O programa deverá começar por perguntar ao usuário o número de inteiros a armazenar; deverá de seguida alocar dinamicamente um array de inteiros capaz de armazenar esse número de inteiros; e por fim deverá lê-los do teclado e reescrevê-los na tela. Antes de terminar não esquecer de libertar a memória alocada.
2. Faça uma função que receba um valor  $n$  e crie dinamicamente um vetor de  $n$  elementos e retorne um ponteiro. Crie uma função que receba um ponteiro para um vetor e um valor  $n$  e imprima os  $n$  elementos desse vetor. Construa também uma função que receba um ponteiro para um vetor e libere esta área de memória. Ao final, crie uma função principal que leia um valor  $n$  e chame a função criada acima. Depois, a função principal deve ler os  $n$  elementos desse vetor. Então, a função principal deve chamar a função de impressão dos  $n$  elementos do vetor criado e, finalmente, liberar a memória alocada através da função criada para liberação.
3. Construa um programa (main) que aloque em tempo de execução (dinamicamente) uma matriz de ordem  $m \times n$  (linha por coluna), usando  $1 + m$  chamadas a função malloc. Agora, aproveite este programa para construir uma função que recebendo os parâmetros  $m$  e  $n$  aloque uma matriz de ordem  $m \times n$  e retorne um ponteiro para esta matriz alocada. Crie ainda uma função para liberar a área de memória alocada pela matriz. Finalmente, crie um novo programa (main) que teste/use as duas funções criadas acima.
4. Escreva um programa em linguagem C que solicite ao usuário um vetor de notas (números reais) e imprime a média aritmética das notas. Considere:

- Apesar de não ser necessário utilize um vetor.
- O programa não deve limitar o tamanho do vetor.
- Não deve ocorrer desperdício de memória.
- Após ser utilizada a memória deve ser devolvida.

## 5 Cadeias de caracteres (strings)

1. Escreva uma função para contar as vogais numa string.
2. Escreva uma função que receba uma string e substitua cada segmento de dois ou mais espaços em branco por um só carácter ' '.
3. Dado o nome completo de uma pessoa imprimir apenas as iniciais seguidas cada uma de ponto e espaço.

## 6 Recursão

1. Faça uma função recursiva que leia um número real  $base(b)$  e um expoente( $e$ ) inteiro positivo para calcular:  $b^e$ .
2. O algoritmo de Euclides para o cálculo do máximo divisor comum é definida como segue:

$$f(m, n) = \begin{cases} m & \text{se } n=0 \\ f(m, m\%n) & \text{se } n>0 \end{cases}$$

Escreva um programa recursivo para calcular máximo divisor comum para  $m$  e  $n > 0$ .

3. Faça uma função recursiva para calcular o produto de dois números  $A$  e  $B$  de forma recursiva.  $A \geq 0$  e  $B \geq 0$ .
4. Faça uma função recursiva para calcular a divisão de dois números  $A$  e  $B$  de forma recursiva.  $A > 0$  e  $B > 0$ .