

## MC102— Algoritmos e Programação de Computadores

Priscila Saboia

21 de abril 2012

**Lista de Exercícios 03**

1. Escreva uma função chamada `primo` que retorne 1 quando receber um valor primo como parâmetro, caso contrário retorne 0.
2. Escreva uma função chamada `somavetor` que possua dois parâmetros: um vetor de inteiros e o número de elementos do vetor. A função deve retornar a soma dos elementos do vetor.
3. Faça duas funções: a primeira deve ser responsável por fazer a leitura dos elementos de um vetor (usar `scanf`) e a outra pela impressão na tela dos elementos do vetor.
4. Escreva uma função `mdc` que receba dois inteiros como parâmetros e devolva o máximo divisor comum entre eles.
5. O mínimo múltiplo comum (mmc) de dois números inteiros  $a$  e  $b$  é o menor inteiro positivo que é múltiplo simultaneamente de  $a$  e de  $b$ . Se não existir tal inteiro positivo, por exemplo, se  $a = 0$  ou  $b = 0$ , então o  $mmc(a, b)$  é zero por definição. Se nem  $a$  e  $b$  são zero, o mmc pode ser obtido usando o máximo divisor comum (mdc) entre  $a$  e  $b$ :

$$mmc(a, b) = \frac{a \cdot b}{mdc(a, b)}$$

Escreva uma função `mmc` que receba dois inteiros como parâmetro e retorne o mmc entre eles.

A função `mmc` deve calcular o mínimo múltiplo comum utilizando a função `mdc` feita no exercício 4.

6. Escreva uma função `transporMatriz` que receba por parâmetro uma matriz  $4 \times 5$  e outra matriz  $5 \times 4$ . A função deve transpor a matriz  $4 \times 5$  e armazenar os resultados na matriz  $5 \times 4$ .
7. Utilizando variáveis para especificar a dimensão da matriz, reescreva a função `transporMatriz` feita no exercício 6. A função deve receber o número de linhas e colunas por parâmetro e transpor a matriz com as dimensões especificadas.
8. Reescreva a função feita no exercício anterior de forma a utilizar variáveis globais em vez de parâmetros.

9. Entenda o programa abaixo, em seguida, reestruture tal programa utilizando **funções** e acrescente a opção “todas as verificações”.

```
int main() {
    int i, n, v;
    int op;

    do {
        printf("1 - Número primo\n");
        printf("2 - Número par\n");
        printf("3 - Sair\n");

        printf("Opção: ");
        scanf("%d", &op);

        switch (op) {
            case 1:
                printf("n: ");
                scanf("%d", &n);
                i = 2;
                v = 1;

                while (i <= n/2 && v == 1) {
                    if (n % i == 0)
                        v = 0;
                    i++;
                }

                if (v)
                    printf("O número %d é primo.\n", n);
                else
                    printf("O número %d não é primo.\n", n);
                break;
            case 2:
                printf("n: ");
                scanf("%d", &n);
                if (n % 2 == 0)
                    printf("O número %d é par.\n", n);
                else
                    printf("O número %d é ímpar.\n", n);
                break;
            case 3:
                break;
            default:
                printf("Opção inválida.\n");
        }

    } while (op != 3);

    return 0;
}
```

10. Escreva a função `inverteString` que está sendo invocada dentro da função `main()`, na linha 7, do programa abaixo:

```
1   int main (void) {
2       char st[100];
3
4       printf("Digite uma string.\n");
5       fgets(st, 100, stdin);
6
7       inverteString(st);
8
9       printf("String invertida: %s\n", st);
10
11      return 0;
12  }
```

11. Ao fazer a questão 10, dentro da função `inverteString` você deve ter sentido a necessidade de trocar elementos da string (vetor de caracteres passado por parâmetro). Faça uma função `trocaChar` que receba uma string por parâmetro e dois inteiros que indicam as posições dos caracteres que devem ser trocados. Além disso, modifique a função `inverteString` de forma que esta invoque a função `trocaChar`, quando for necessário trocar elementos da string.
12. Uma matriz quadrada inteira é chamada de “quadrado mágico” se a soma dos elementos de cada linha, a soma dos elementos de cada coluna e a soma dos elementos de cada diagonal (principal e secundária) são iguais. A matriz abaixo representa um quadrado mágico:

```
8   1   6
3   5   7
4   9   2
```

Escreva uma função que verifique se uma matriz de  $n$  linhas e  $c$  colunas representa um quadrado mágico.

13. Considere o programa abaixo. Depois de executado, quais são os valores associados aos itens de (a) a (h). Suponha que os endereços das variáveis  $u$  e  $v$  são 1000 e 1004, respectivamente.

```
int main() {
    int u, v;
    int *pu, *pv;
    v = 45;
    pv = &v;
    *pv = v + 1;
    u = *pv + 1;
    pu = &u;
    return 0;
}
```

- |       |       |        |      |
|-------|-------|--------|------|
| a) &u | b) pu | c) *pu | d) u |
| e) &v | f) pv | g) *pv | h) v |

14. Considere o programa abaixo. Depois de executado, quais são os valores associados aos itens de (a) a (i). Suponha que os endereços das variáveis  $a$ ,  $b$  e  $c$  são 1000, 1004 e 1008, respectivamente.

```
int main() {
    float a, b;
    float c, *pa, *pb;
    a = 0.001;
    b = 0.003;
    pa = &a;
    *pa = 2 * a;
    pb = &b;
    c = 3 * (*pa + *pb);
    return 0;
}
```

- |           |       |            |       |        |
|-----------|-------|------------|-------|--------|
| a) &a     | b) &b | c) &c      | d) pa | e) *pa |
| f) &>(*v) | g) pb | h) &>(*pb) | i) c  |        |

15. Considere o programa abaixo e escreva a função `gauss` (invocada na linha 13). A função deverá receber como parâmetro o vetor com  $n$  valores e as medidas `media`, `mediana` e `desvio`. A função deve calcular as medidas solicitadas (através dos parâmetros) e retornar com o uso do `return` a quantidade de elementos do vetor que estão no intervalo definido por `[media - desvio, media + desvio]`.

```
1  int main (void) {
2
3      float notas[60], media, mediana, desvio;
4      float lim_inf, lim_sup;
5      int i, n, res;
6
7      printf("Digite o tamanho do vetor.\n");
8      scanf("%d", &n);
9
10     for( i = 0; i < n; i++)
11         scanf("%f", &vetor[i]);
12
13     res = gauss(vetor, n, &media, &mediana, &desvio);
14     lim_inf = media - desvio;
15     lim_sup = media + desvio;
16
17     printf("A nota mediana da P1 foi %.2f.\n", mediana);
18     printf("%d alunos possuem nota da P1", res);
19     printf(" no intervalo [%.2f, %.2f].\n", lim_inf, lim_sup);
20
21     return 0;
22 }
```

16. Faça uma função que troca o valor de duas variáveis do tipo inteiro. Ex: A função seria chamada da seguinte forma:

```
int x=10, y=20;
troca(&x, &y);
printf("x=%d y=%d", x, y);
```

A saída desse printf seria: x=20 y=10.

17. Faça uma função que recebe uma matriz, as dimensões l (número de linhas) e c (número de colunas) da matriz e retorna a soma dos elementos da diagonal principal da matriz. O cabeçalho deve ser:

```
int soma_diagonal(int matriz[MAX][MAX], int l, int c);
```

18. Determine o valor especificado em cada item após a execução das seguintes instruções, supondo que o endereço da variável x é 1000 e da variável y é 1004.

```
#include <stdio.h>
int main(void) {
    int x,y;
    int* p1;
    int* p2;

    x = 10;
    y = 20;
    p1 = &x;
    p2 = &y;
    (*p1)++;
    return 0;
}
```

- |              |          |           |
|--------------|----------|-----------|
| a) x         | b) y     | c) &x     |
| d) &y        | e) p1    | f) p2     |
| g) *p1 + *p2 | h) *(&x) | i) &(*p2) |

19. Qual a saída do programa a seguir?

```
int main(void) {
    int i,j;
    int m[14] = {1,2,3,4,5,6,7,8,9,10,11,12,13,14};

    for(i=0; i<7; i++)
        for(j=0; j<2; j++){
            if(i>0 && j%2==0)
                printf("\n");
            printf("%d ", m[i*2 + j]);
        }
    return 0;
}
```

20. Mostre a saída do programa a seguir:

```
#include <stdio.h>

int x=10;

int dobro(int x){
    x = x*2;
    return x;
}

int main(void){
    printf("%d\n", x);
    int x=20;
    printf("%d\n", x);
    printf("%d\n", dobro(x));
    printf("%d\n", x);
    return 0;
}
```

21. Faça uma função que recebe uma string e um caractere e retorna a quantidade de vezes que esse caractere aparece na string.

22. Faça uma função que recebe duas strings e dois caracteres (c1 e c2). A função deve copiar a primeira string para a segunda, substituindo por c2 os caracteres que forem iguais a c1. O protótipo da função pode ser:

```
void substitui(char string1[MAX], char string2[MAX], char substituido, char substituto);
```

Exemplo:

Se a primeira string for “Casa amarela” e os caracteres forem 'a' e 'A', a segunda string deverá ter o valor “CAsA AmArelA” após a execução da função.

23. Faça uma função que recebe uma matriz, suas dimensões e um valor inteiro e modifica cada posição da matriz pela multiplicação do valor original da matriz pelo valor passado.