

# MC102—Algoritmos e Programação de Computadores

Turmas U e V

Segundo Semestre de 2007

Lista de exercícios 02

P: Qual a matéria da prova?

R: Aula 1 até aula 11.

P: O que será avaliado nos exercícios da prova que terei que fazer programas?

R: Será avaliado principalmente se o código faz **exatamente** o que foi pedido no enunciado. Também será avaliada a sintaxe da linguagem, principalmente uso das chaves nos blocos de comando, já que o esquecimento ou uso incorreto pode levar o programa a não fazer o que deveria. Note que não há implicações no esquecimento de um ou outro ponto e vírgula, mas poderá haver se houverem muitos comandos sem ponto e vírgula ou uso indevido/incorreto do ponto e vírgula. Outro ponto avaliado é a declaração de variáveis com tipos compatíveis com o uso. Por exemplo, se o enunciado diz que serão usados números reais mas os números são lidos como inteiros, será descontado. Ainda, se o enunciado diz que serão números inteiros e é usado float, também é passível de desconto. Obviamente há casos que declaração de tipo diferente do número lido é justificável, como por exemplo quando a variável recebe algum número não inteiro durante o programa. Nesse caso, deve haver justificativa do uso do tipo colocada como comentário no programa. Se houver questões que perguntam a saída de algum programa, a saída deve ser condizente com o tipo da variável.

P: Se eu fizer só os exercícios marcados com \* eu vou bem na prova?

R: O conjunto de exercícios marcados com \* compreende o CONTEÚDO que poderá ser cobrado na prova. Isso não quer dizer de forma alguma que se você fizer apenas esses exercícios você saberá fazer toda a prova ou que tais exercícios são parecidos com os da prova. Note que, para fazer os exercícios dessa lista, é necessário conhecer a matéria desde a primeira aula.

0. Escreva um programa em linguagem C que recebe um conjunto A de números reais não negativos como entrada. O programa deve parar de ler números quando o usuário digitar um número negativo, e este não deve ser incluído como um número do conjunto A. Lidos os números, o programa deve procurar pela maior seqüência não decrescente dentro do conjunto A e imprimir na tela o tamanho dessa seqüência e a seqüência em si. Caso haja mais de uma seqüência de tamanho máximo, o programa deve imprimir a primeira encontrada.

Exemplo de entrada: 1 2 2 3 4 3 0 98 7 6 17 38 91 100 293 649 1000 1000 2000 34 12 78 99 100 -1

Saída para a entrada acima:

Tamanho: 10

Seqüência: 6 17 38 91 100 293 649 1000 1000 2000

1. ★ Escreva um programa que lê um número N e imprime as N primeiras linhas do triângulo de pascal.

2. ★ Considere o programa abaixo:

```
#include
```

```
int main(void) {
    unsigned int b, n, d;

    printf("Informe um numero no intervalo [0,255]: ");
    scanf("%u", &n);
    b = 0;
    d = 1;
    if ((n <= 0) && (n <= 255)) {
        while (n >= 1) {
            b = b + d * (n % 2);
            d = d * 10;
            n = n / 2;
        }
        printf("Resultado: %u.\n", b);
    } else
        printf("Valor fora do intervalo valido.\n");
}
```

Preencha a tabela abaixo para cada iteração do loop while com os valores de b, d, n e a saída do programa, para os valores de n dados.

	n = 5			n=26		
Iteração	b	d	n	b	d	n
1						
2						
3						
4						
5						
6						
7						
8						
	Saída do programa			Saída do programa		

3. Escreva um programa que calcule a média de n números, sendo n também informado pelo usuário.

4. ★ O que podemos dizer sobre um loop while que, dentro de seu bloco de comandos, não faz alteração em nenhuma variável da sua condição?

5. ★ Escreva um programa que lê uma lista de inteiros positivos e imprime a lista em ordem inversa. A lista pode ter até 100 elementos e a leitura deve ser interrompida assim que o primeiro inteiro não positivo for digitado. Não será informado previamente o número de inteiros da lista de inteiros.

6. ★ Escreva um programa que leia as coordenadas de um vetor de dimensão n, n informado pelo usuário, e calcule o módulo desse vetor. O módulo de um vetor de dimensão n é calculado da seguinte forma:

$$modulo = \sqrt{\sum_{k=1}^n coordenada_k^2}$$

8. ★ Escreva um programa que leia um vetor de n posições, n fornecido pelo usuário, e mostre na tela a quantidade de elementos pares nesse vetor.

9. ★ Escreva um programa que leia um vetor de n posições, n fornecido pelo usuário, e mostre na tela o índice do maior elemento e o índice do menor elemento desse vetor.

10. Escreva um programa que lê um vetor com n inteiros, n fornecido pelo usuário, e mostra na tela todos os arranjos dos inteiros desse vetor. Ex: entrada: 1 2 3. Saída: 1 2 3, 1 3 2, 2 1 3, 2 3 1, 3 1 2, 3 2 1.

11. ★ Escreva um programa que calcula a soma de dois polinômios de grau  $n$  e  $m$ . O seu programa deve ler primeiramente os graus  $n$  e  $m$  de cada polinômio e em seguida os  $n + 1$  e  $m + 1$  coeficientes de cada um dos dois polinômios, ordenados do coeficiente de maior grau para o coeficiente de menor grau em cada polinômio. O programa deve escrever na tela o grau do polinômio resultante e o polinômio ordenado do maior grau para o menor grau. Assuma que o grau máximo do polinômio é 25 e que somente valores válidos são digitados. Um exemplo de execução seria:

```
$/pol
5 2
5.0 2.0 3.3 2.0 4.5 5.2
3.0 -2.5 2.8
Grau do polinomio: 5
Polinomio Resultante: 5.0x^5 + 2.0x^4 + 3.3x^3 + 5.0x^2 + 2.0x^1 + 8.0x^0
```

12. Considere o enunciado da questão 10 para escrever um programa que multiplica dois polinômios.

13. ★ Escreva um programa que leia um valor  $n$  da entrada padrão e preencha uma matriz  $n \times n$  de tal forma que esta torne-se a matriz identidade  $I_n$ . Ao final, imprima a matriz na tela. Considere  $0 < n \leq 100$ .

14. ★ Escreva um programa que leia da entrada padrão uma matriz de tamanho  $n \times m$  ( $n$  e  $m$  fornecidos pelo usuário) e escreva na saída padrão o valor máximo e o valor mínimo existentes naquela matriz. Considere  $0 < n \leq 100$  e  $0 < m \leq 100$ .

15. Escreva um programa que calcule as  $n$  ( $n \leq 10$ ) primeiras potências do número  $k$ . A função `pow(x,y)` pode ser usada. Para usá-la, deve-se incluir a diretiva `#include<math.h>`. Para  $n = 3$  e  $k = 2$ , a saída seria:

```
2 elevado a 1 = 2
2 elevado a 2 = 4
2 elevado a 3 = 8
```

16. ★ Escreva um programa que calcule o  $n$ -ésimo número da seqüência de Fibonacci,  $n$  informado pelo usuário. A séria de Fibonacci é definida como:

$$\text{Fib}(n) = \begin{cases} 0 & \text{caso } n \leq 0 \\ 1 & \text{caso } n = 1 \leq \\ \text{Fib}(n - 1) + \text{Fib}(n - 2) & \text{caso contrário} \end{cases}$$

17. ★ Considere o seguinte esqueleto de um código fonte C:

```
typedef float prova;
```

```

typedef float media;

struct provas{
    prova p[3];
    media M;
};

typedef struct provas Provas;

struct aluno{
    char ra[7];
    int frequencia;
    Provas Ps;
};

int main(void) {
    int n,i,j;
    float desvios_Ps[3]={0.0, 0.0, 0.0}, medias_Ps[3]={0.0, 0.0, 0.0};
    struct aluno alunos [MAX_ALUNOS];

scanf("%d", &n);

/* COMPLETE AQUI */

}

```

Complete o esqueleto acima. O programa deve ler um número inteiro  $n$  que indicará a quantidade de alunos na turma. Após isso, deve ler as notas das provas de cada aluno, calcular e mostrar na tela as notas e a média de cada aluno, a média geral de cada prova de todos os alunos e o desvio padrão de cada prova de todos os alunos. Note que não é necessária a declaração de mais nenhuma variável. Exemplo:

ENTRADA:

```

3
030034
15
4 5 6
123456
12
0 5 10
987654
10

```

6 9 3

SAIDA:

RA: 030034

Prova 0: 4.000000

Prova 1: 5.000000

Prova 2: 6.000000

Média: 5.000000

-----

RA: 123456

Prova 0: 0.000000

Prova 1: 5.000000

Prova 2: 10.000000

Média: 5.000000

-----

RA: 987654

Prova 0: 6.000000

Prova 1: 9.000000

Prova 2: 3.000000

Média: 6.000000

-----

Média geral P0: 3.333333

Desvio padrão P0: 2.494438

Média geral P1: 6.333333

Desvio padrão P1: 1.885618

Média geral P2: 6.333333

Desvio padrão P2: 2.867442

18. \* Escreva um programa que recebe duas strings (vetores de caracteres com '\0' no final) como entrada e procura cada ocorrência da primeira string dentro da segunda, imprimindo na tela o índice do início da primeira string na segunda cada vez que encontrá-la. Proibido uso da biblioteca string.h. Exemplo:

ENTRADA:

abc

abkpoeirhabckljuoperabababababceropabcp

SAIDA:

9

26

33

19. \* Faça um programa que recebe uma string de letras (a-z e A-Z) e converte todas as suas letras minúsculas para maiúsculas e vice-versa. O programa deve verificar se todos os caracteres digitados são letras e se houver algum que não é letra deve imprimir mensagem na tela. Dica: use a tabela ASCII. Exemplo:

ENTRADA:

aEiOu

SAIDA:

AeIoU