

Algoritmos e Programação de Computadores

3ª Prova

Nome:

RGA:

Questão	Valor	Nota
1	3,0	
2	2,5	
3	2,5	
4	2,0	
Total	10,0	

Instruções: Não é permitida consulta a qualquer material. Somente serão consideradas respostas nos espaços marcados. Use os versos das folhas como rascunho.

1. O programa a seguir faz a leitura de seu RGA e armazena os quatro últimos dígitos em um vetor de inteiros. Este vetor é utilizado para preencher uma matriz de inteiros.

```
1 #include <stdio.h>
2
3 int main() {
4     int vetor[4], matriz[4][4], i, j, coluna, x, maior;
5     long rga;
6     scanf("%ld", &rga);
7     for(i = 4; i > 0; i--) { /* Leitura dos quatros últimos dígitos do seu RGA */
8         vetor[i-1] = rga % 10;
9         rga /= 10;
10    }
11
12    /* Início da etapa de preenchimento */
13    for(i = 0; i < 4; i++)
14        for(j = 0; j < 4; j++)
15            matriz[i][j] = i + j;
16
17    for(i = 0; i < 4; i++)
18        matriz[i][i] = vetor[i];
19    /* Fim da etapa de preenchimento */
20
21    coluna = 3;
22    maior = matriz[3][coluna];
23    for(i = 3; i >= 0; i--) {
24        for(j = 3; j >= 0; j--) {
25            if(matriz[i][j] > maior) {
26                maior = matriz[i][j];
27                coluna = j;
28            }
29        }
30    }
31    x = 0;
32    for(i = 0; i < 4; i++)
33        x += matriz[i][coluna];
34
35    printf("x = %d\n", x);
36    return 0;
37 }
```

a) Mostre a matriz obtida por este programa após a etapa de preenchimento.

b) Qual o valor é impresso na linha 35? O que a variável x representa?

c) Mostre uma matriz de tamanho 4×4 , para a qual, em uma execução deste código, é máxima a quantidade de vezes em que as instruções das linhas 26 e 27 são executadas.

2. A seguinte função recursiva recebe um número inteiro positivo n e um vetor v contendo n números inteiros em ordem crescente. Preencha os espaços deixados em branco para que a função verifique se v possui números repetidos. Os valores 1 e 0 devem ser devolvidos, respectivamente, para indicar se há ou não repetições.

```
1 int rep(int v[], int n) {  
2     if(  ) //Base da recursão  
3         return 0;  
4     return  || ;  
5 }
```

[illegible]

4. O programa a seguir recebe uma cadeia de caracteres contendo uma mensagem escondida. A mensagem foi codificada considerando as diagonais de uma matriz de tamanho $n \times n$ (veja exemplo). O programa deverá mostrar a mensagem na tela. As funções `main` e `tamanho_frase` são dadas. Implemente as outras duas funções.

Exemplo de mensagem de entrada

1lcpskl+y5mcpj3bpbsllcrehfmecwitsnglrex(tj(xypynbqfxx)uvgfaj)

Mensagem em formato matricial

1	l	c	p	s	k	l	+
y	5	m	c	p	j	3	b
p	b	s	l	l	c	r	e
h	f	m	e	o	k	e	c
w	i	t	s	n	g	l	r
e	x	(t	j	(x	y
p	y	n	b	q	f	x	x
)	u	v	g	f	a	j)

Mensagem escondida

15sen(x)+3cos(y)

```
#include <stdio.h>
#include <math.h>

int tamanho_frase(char frase[]);
void preencher_matriz(char frase[], char matriz[100][100], int n);
void mostrar_mensagem(char matriz[100][100], int n);

int main()
{
    char matriz[100][100];
    char frase[10000];
    int n;

    scanf("%s", frase);
    n = sqrt(tamanho_frase(frase));
    preencher_matriz(frase, matriz, n);
    mostrar_mensagem(matriz, n);

    return 0;
}
```

```
int tamanho_frase(char frase[])
{
    int i = 0;
    while(frase[i] != '\0') i++;
    return i;
}
```

a) Implemente a função `preencher_matriz`, que recebe uma cadeia de caracteres de tamanho $n \times n$ e preenche a matriz com estes caracteres.

```
void preencher_matriz(char frase[], char matriz[100][100], int n) {
```

[illegible]

b) Implemente a função `mostrar_mensagem` que imprime a mensagem escondida nas diagonais da matriz.

```
void mostrar_mensagem(char matriz[100][100], int n) {
```

[illegible]