

Universidade Federal Fluminense
Instituto de Computação
Departamento de Ciência da Computação
Curso: Programação de Computadores II
Prof.: Anselmo Montenegro
Setembro de 2016

1ª Lista de Exercícios

Conceitos

- 1) Descreva e explique cada um dos componentes de um programa em C.
- 2) Defina os seguintes conceitos: identificador, literal, variável, tipo, declaração, atribuição, inicialização, expressão e precedência.
- 3) Diferencie variáveis globais de locais.
- 4) Defina o que é uma função auxiliar.
- 5) Explique o que é passagem por valor. O que seria passagem por referência?
- 6) Explique o conceito de escopo.
- 7) Qual o significado dos símbolos /*, */ e //?

- 8) Explique por que o código abaixo não efetua a troca das variáveis a e b conforme desejado. Faça o passo-a-passo e a representação em memória das variáveis locais e automáticas de cada função ativa.

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

```
void troca (int x, int y){
```

```
    intaux;
```

```
    aux = x;
```

```
    x = y;
```

```
    y = aux;
```

```
}
```

```
intmain (void){
```

```
    int a = 10;
```

```
    int b = 5;
```

```
    troca(a,b);
```

```
    return 0;
```

```
}
```

- 9) Explique por que o comando scanf requer após a string de formatação endereços das variáveis a serem modificadas.
- 10) Por que o comando printf se comporta de forma distinta do scanf?
- 11) Defina o conceito de efeito colateral de uma função.
- 12) Defina o que é um ponteiro. Dê exemplos.

Estruturas de seleção e repetição

- 13) Escreva um programa que recebe três valores inteiros a, b e c e os coloque em ordem. Generalize seu algoritmo para quatro valores a,b, c e d.
- 14) Implemente o jogo Pedra, Papel e Tesoura, onde um dos jogadores é o computador. A escolha do computador deve ser aleatória. Faça uma pesquisa sobre como gerar um inteiro

aleatório entre 0 e $n-1$ em C .

- 15) Um número a é dito *permutação* de um número b se os dígitos de a formam uma permutação dos dígitos de b .

Exemplo: 5412434 é uma permutação de 4321445, mas não é uma permutação de 4312455.

Obs.: Considere que o dígito 0 (zero) não aparece nos números.

Faça um programa que lê dois inteiros positivos a e b e responda se a é permutação de b .

- 16) Dado um número inteiro positivo n , determinar todos os inteiros entre 1 e n que são comprimentos das hipotenusas de triângulos retângulos com catetos inteiros.

- 17) Escreva uma função que recebe um inteiro em representação decimal e converta para representação binária

- 18) Escreva um programa que decomponha um inteiro em fatores primos.

- 19) Escreva um programa que recebe como entrada uma data, na forma de três inteiros d , m , a e retorne o dia da semana.

- 20) Escreva um programa que tome um inteiro N e imprima todos os inteiros que podem ser expressos como a soma de dois cubos de duas formas, isto é, encontre inteiros a, b, c e d tais que $a^3 + b^3 = c^3 + d^3$.

- 21) Caminho aleatório 2D. Um caminho aleatório 2D simula o comportamento de uma partícula que se move em um grid de pontos. Em cada passo, a partícula se move para norte, sul, leste e oeste com probabilidade $\frac{1}{4}$, independente dos movimentos anteriores. Escreva um programa que estime quantos passos são necessários para a partícula chegar a fronteira de um quadrado de $2N$ por $2N$ elementos centrado no ponto inicial.

- 22) O **dígito verificador** é um mecanismo de autenticação utilizado para verificar a validade e a autenticidade de um valor numérico, evitando dessa forma fraudes ou erros de transmissão ou digitação. Consiste em um ou mais dígitos acrescentados ao valor original e calculados a partir deste através de um determinado algoritmo. Números de documentos de identificação, de matrícula, cartões de crédito e quaisquer outros códigos numéricos, que necessitem de maior segurança, utilizam dígitos verificadores (Wikipedia).

Uma das rotinas mais tradicionais para cálculo do dígito verificador é denominada Módulo 11, que funciona da seguinte forma: cada dígito do número, começando da direita para a esquerda (menos significativo para o mais significativo) é multiplicado, na ordem, por 2, depois 3, depois 4 e assim sucessivamente, até o limite de multiplicação escolhido. Então novamente multiplica-se o número por 2, 3, etc.

2	6	1	5	3	3	-	9
x7	x6	x5	x4	x3	x2		
14	36	5	20	9	6		
14 + 36 + 5 + 20 + 9 + 6						=	90 x 10) / 11 = 81, resto 9 => DV = 9

Escreva um programa que receba um número inteiro, juntamente com um dígito verificador. Calcule o dígito verificador do número usando a técnica descrita acima, considerando que o limite de multiplicação é igual a 9 (após multiplicação por 9, a multiplicação retorna a 2). O algoritmo deve imprimir uma mensagem, indicando se o número é válido ou não segundo o código.

- 23) Mediana de 5. Escreva um programa que tome 5 valores inteiros e imprima a mediana do conjunto, isto é, o elemento x tal que dois são maiores e dois são menores. Tente escrever seu programa usando menos que 7 comparações (difícil).

Arrays e Matrizes

- 24) Escreva funções em C que calculem para um *array* de inteiros, juntamente com o número de elementos, as seguintes medidas estatísticas:
- a moda dos elementos no *array* (elemento mais freqüente)
 - a mediana dos elementos no *array* (elemento central)
 - a média.
- 25) Amostragem sem reposição. Escreva um programa tal que dado um conjunto de N números inteiros, sorteie $n < N$ elementos tal que os elementos sorteados não apareçam de forma repetida. Resolva o problema sem a necessidade de repetição de sorteio, isto é, elementos sorteados garantidamente não são novamente sorteados pelo gerador aleatório.
- 26) Escreva um programa que implemente o Crivo de Eratóstenes que é um algoritmo que, dado um número N , determina todos os números primos menores que N .
- 27) Dada uma seqüência de n números inteiros determinar o comprimento de um segmento crescente de comprimento máximo.
- Exemplos:
- Na seqüência 5, 10, 3, 2, 4, 7, 9, 8, 5 o comprimento do segmento crescente máximo é 4.
- Na seqüência 10, 8, 7, 5, 2 o comprimento de um segmento crescente máximo é 1.
- 28) Uma imagem em preto e branco, de tamanho $m \times n$, pode ser representada por uma matriz cujos elementos assumem valores no conjunto $\{0,1\}$. Dado um padrão representado por uma matriz 3×3 também assumindo valores em $\{0,1\}$, escreva um programa que determine se o

padrão existe ou não na imagem.

0	0	1	0	0	0	0	1	0
0	1	1	1	1	1	1	0	0
0	0	1	0	1	1	1	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	1	0	0
0	0	0	0	0	1	0	0	0
1	1	1	0	1	0	1	0	0
1	0	1	0	0	0	0	0	0
1	1	1	0	0	1	1	1	0

0	1	0
1	1	1
0	1	0

29) Escreva um programa C capaz de jogar o jogo da velha e que nunca perca.

30) **Sudoku** é um quebra-cabeça, cujo objetivo do jogo é preencher os números de 1 a 9 em cada uma das células vazias numa grade de 9×9, constituída por 3×3 subgrades chamadas regiões. Cada coluna, linha e região só pode ter um número de cada um dos números de 1 a 9. Exemplo de um jogo:

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

Imagem extraída de: Wikipédia

Faça um programa em C que, dado um jogo Sudoku, representado por uma matriz 9x9, verifica se o jogo está ou não correto.

31) **Minesweeper**. Escreva um programa que tome 3 parâmetros M, N e p e produza uma matriz de 0's e 1's onde cada posição é ocupada com probabilidade p. No jogo Minesweeper, células ocupadas representam bombas e vazias representam células seguras. Imprima a matriz usando * para bombas e "." para células seguras. Então, substitua cada célula segura pelo número de bombas vizinhas (acima, abaixo, à direita e à esquerda).

32) Escreva um programa C que resolva o problema do passeio do cavalo no tabuleiro de Xadrez (Wirth, 1976).

A partir de uma posição inicial (x_0, y_0) , o problema consiste em encontrar, se existir, um passeio do cavalo com $n^2 - 1$ movimentos tal que todos os pontos do tabuleiro são visitados uma única vez (difícil).

Alocação Dinâmica

- 33) Escreva uma função em C que aloque uma matriz bidimensional $n \times m$, de valores de tipo double de forma dinâmica usando um único array. Escreva as funções para leitura e modificação dos elementos usando dois índices i e j .

```
double *criar_matriz(int m, int n);  
double get_valor(double *matriz, int i, int j);  
void set_valor (double *matriz, int i, int j);
```

- 34) Repita o exercício 33 usando vetor de ponteiros.

```
double **criar_matriz(int m, int n);  
double get_valor(double **matriz, int i, int j);  
void set_valor (double **matriz, int i, int j);
```

- 35) Escreva uma função que insira um elemento em um array alocado de forma dinâmica de tamanho N na posição i . Sua função deve redimensionar o array para o dobro do tamanho caso elementos não possam mais ser inseridos.

- 36) Repita os exercícios 33 e 34, para uma matriz tridimensional, onde os parâmetros são a altura, largura e profundidade. Descreva a assinatura das funções de criação e acesso.

Arquivos Texto

- 37) Faça um programa em C que gere uma versão criptografada de um arquivo texto trocando cada caractere de código ASCII j pelo caractere de código ASCII $j+k$, onde k é um parâmetro especificado pelo usuário.

- 38) Faça um programa C que leia um arquivo texto chamado entrada.txt e imprima, em outro arquivo texto, denominado saída.txt, o total de letras, vogais, consoantes, espaços em branco, palavras e o total de linhas encontradas.

- 39) Considere dois arquivos “itens1.dat” e itens2.dat”, contendo registros sobre itens de estoque de um supermercado. Cada registro contém o nome do produto, preço, marca e data de validade. Em ambos os arquivos, os registros estão ordenados pelo nome do produto. Escreva um programa que leia os dois arquivos e gere um terceiro formado pela combinação dos dois anteriores de modo que os registros continuem ordenados pela chave nome. Os arquivos não devem ser lidos para memória.
- 40) Escreva um programa que receba dois textos e determine se existe pelo menos uma mesma sequência de palavras de tamanho maior ou igual a cinco (5) em ambos os arquivos.

Arquivos Binários

- 55) Escreva um programa em C que seja capaz de ler uma imagem de dimensões 256x256 em formato Raw (arquivo binário contendo apenas uma sequência de bytes indicando um tons de cinza entre 0 e 255) e gere seu histograma. O histograma é um mapa que associa a cada tom o número de suas ocorrências na imagem.
- 56) Um geofísico armazenou uma coleção de imagens, cada uma delas representada através de um par de arquivos. O primeiro arquivo é um arquivo texto com extensão .hdr, que descreve a largura e altura da imagem, cujos dados são armazenados num segundo arquivo. O segundo arquivo está organizado em um formato .raw (dados crus) e é simplesmente um conjunto de bytes consecutivos para os canais r (vermelho), b(azul), g(verde). Isto significa que cada ponto da imagem é representado por uma tupla (r,g,b) de bytes consecutivos. Escreva em C duas funções que efetuem as seguintes operações:

/ Dado o nome de uma imagem, a partir do arquivo nome.hdr determinar as dimensões do arquivo .rawcorrespondente. */*

voidlerDimensao (char * nome, int * altura, int *largura);

*/*Lê os bytes de um arquivo nome.raw e os armazena e os armazena em três arrays de byte distintos, um para cada canal de cor, todos com dimensão igual a altura *largura.*

voidlerImage (char * nome, int altura, int largura, byte * r, byte * g, byte * b);

*/*Escreve os arquivos nome.raw (binário) e nome.hdr(texto) correspondentes a uma imagem de dimensão altura*largura armazenada em três arrays de byte um para cada canal. */*

void Escrever(char * nome, int altura, int largura, byte * r, byte * g, byte * b);

Caracteres

- 57) Escreva uma função que verifique se um caractere é um dígito.
- 58) Escreva uma função que verifique se um caractere é uma letra do alfabeto (maiúscula ou minúscula).

59) Explique como são definidas cadeias de caracteres em C. Qual o objetivo do valor `\0`?

60) Escreva uma função que receba uma cadeia de caracteres, representada por um *array* de char, que armazena o caminho do diretório raiz até um arquivo específico.

Exemplo: `C:\musicas\somewhereintime\themeofpaganini.mp3`

A função deve gerar uma nova cadeia que contenha apenas o nome do arquivo sem o caminho de diretórios.

61) Escreva uma função que receba uma cadeia de caracteres e gera sua versão invertida.

62) Escreva uma função que verifique se uma cadeia de caracteres é um palíndromo.

63) Escreva uma função que tome uma cadeia de caracteres *s* e busque uma sub-cadeia de caracteres *t*, onde $|t| < |s|$, onde $| |$ indica o comprimento da cadeia.