

LISTA DE EXERCÍCIOS – MÊS 03

- Qual a faixa de valores para um inteiro representado em complemento de 2 que ocupa 7 bits?
 - $(1100)_2$
 - $(F0)_{16}$
- Converter de decimal para hexadecimal.
 - 1000
 - 500
 - 250
 - 125
 - 62,5
 - 31,25
- Converter de octal para decimal.
 - 54
 - 36
 - 13
 - 5,4
- Converter de octal para decimal
 - 52
 - 25
 - 12,4
 - 5,2
- Some 1 a cada um dos números a seguir, expressando o resultado na mesma base em que o número está expresso.
 - $(10111)_2$
 - $(19)_{10}$
 - $(19)_{16}$
 - $(67)_8$
- Subtraia 1 de cada um dos números abaixo, expressando o resultado na mesma base em que os números estão expressos.
 - $(500)_8$
 - $(1A)_{16}$
 - $(20)_{10}$
- Expresse o complemento de 2 de cada um dos números a seguir, representados na base 2 com 4 bits.
 - 0001
 - 0000
 - 1010
 - 0111
- Os seguintes nibbles (seqüências de 4 bits) representam inteiros em complemento de 2. Mostre como esses mesmos valores seriam representados em 8 bits.
 - 0000
 - 0011
 - 1010
 - 1111
- Colocar em notação de complemento de 2 para inteiros de 6 bits os seguintes números decimais.
 - 32
 - 31
 - 12
 - 12
 - 1
 - 0

10. Colocar em ordem crescente os seguintes inteiros de 8 bits em complemento de 2, representados em hexadecimal. Não há necessidade de converter para decimal. Preste atenção no sinal.

05 1F 80 AD FF 10

11. Faça um programa que converta um valor decimal para binário.
12. Faça um programa que converta um valor binário em valor decimal.
13. Faça um programa que conte quantas vezes um determinado carácter aparece dentro de um literal. Para tanto, leia duas informações: um carácter e um literal, e ao final escreva quantas vezes o carácter lido aparece dentro do literal lido.
14. Faça um programa que codifique uma mensagem segundo a Cifra de César. A codificação é feita substituindo cada letra pela sua terceira sucessora, enquanto que a decodificação é feita alterando-a pela terceira predecessora. Assim, leia a mensagem da entrada e escreva a mensagem codificada na saída. Faça com que o programa permita também a decodificação.
15. Auxiliado pela tabela padrão ASCII construir um programa que leia uma cadeia, supostamente um nome próprio, e converta a primeira letra de cada palavra para maiúscula e todas as outras em minúsculas.
16. Verifique se uma cadeia A contém uma subcadeia B. Escreva, na saída, a posição inicial da subcadeia B dentro de A ou zero caso contrário.
17. Faça um programa que leia um literal e transforme-o em seu inverso, escrevendo-o ao final na saída. Exemplo: “Forster” = “retsroF”.
18. Faça um programa que leia um literal e diga qual carácter ASCII padrão (imprimível) aparece mais vezes e quantas vezes ele aparece dentro do literal lido.
19. Dado um literal origem, construa um literal destino que não contenha qualquer ocorrência de um determinado carácter x. Ao final escreva o conteúdo da variável gerada.

Exemplo:

origem = “Engenharia da Computação”

x = “a”

Destino = “Engenhria d Computção”

20. Dado um literal origem, construa um literal destino que não contenha qualquer ocorrência de qualquer carácter existente em um literal x. Ao final escreva o conteúdo da variável gerada.

Exemplo:

origem = “Engenharia da Computação”

x = “aeiou”

destino = “ngnhr d Cmptçã”

21. Dados dois literais A e B, crie um literal C com os caracteres que aparecem tanto em A quanto em B (a intersecção). A caixa (ou seja, o tamanho) das letras não é importante, ou seja, “A” e “a” são a mesma coisa. Escreva o resultado ao final

Exemplo:

A = “Ciência da Computação”

B = “Informática”

C = “CINAMTO”

22. Dado um literal origem, crie um outro literal destino que contenha apenas os caracteres que aparecem em origem mas que não sejam repetidos. Escreva o resultado ao final.

Exemplo:

origem = "Ciência da Computação"

destino = "Ciêna domputção"

23. Faça um programa para ler um valor literal contendo o nome de uma pessoa, e gerar um outro literal onde os nomes do meio são abreviados, escrevendo o resultado ao final.

Exemplo:

"Michelle de Oliveira Parreira" = "Michelle d. O. Parreira"

24. Faça um programa que leia uma frase e coloque os seus caracteres em ordem alfabética (ou em ordem segundo a tabela ASCII, que é a mesma coisa), escrevendo a cadeia resultante ao final. Considere que os caracteres que aparecem na cadeia são apenas do conjunto ASCII padrão, e (para simplificar) considere também que letras maiúsculas e minúsculas são diferentes.

Exemplo:

"Ciencia da Computacao" = "CCaaaaccdeiimnooptu"

25. Faça um programa que leia um número inteiro na base 10 (decimal) e converta-o em sua representação inteira na base 2 (binária).

Exemplo:

6247943 = 10111110101011000000111

26. Faça um programa que verifique quantos caracteres digitados pelo usuário são dígitos, letras, sinais de pontuação ou outros caracteres, imprimindo a frequência (em porcentagem) de cada um deles ao final do algoritmo. O caracter "_" indica o fim dos dados e não deve ser computado na conta.

27. Quais as finalidades das seguintes funções de string?

- | | |
|-----------|-----------|
| a) strlen | c) strcat |
| b) strcpy | d) strcmp |

28. Quais as diferenças entre break, goto, return e exit()?

29. Por que se evita utilizar goto nos programas? Quais alternativas a linguagem C provê para se evitar o uso de goto?

30. Reescreva o trecho de programa a seguir substituindo o uso de goto por for, while ou do-while.

```
long i=2L;
long z=10L;
p1: i=i+1;
if(i*i*i<z*z) goto p3;
z=z+1;
goto p1;
p3: printf("%ld", i);
```

31. Reescreva o trecho de programa a seguir

- a) substituindo os if-else por switch-case.

b) utilizando um vetor de strings.

```
if(x==1) puts("um");  
else if(x==2) puts("dois");  
else if(x==3) puts("três");  
else if(x==4) puts("quatro");  
else puts("nenhum");
```

32. Quais as finalidades das seguintes funções de entrada e saída?

- | | |
|------------|------------|
| a) getchar | f) puts |
| b) printf | g) gets |
| c) scanf | h) getch |
| d) fprintf | i) putchar |
| e) fscanf | j) getche |

33. Quais as finalidades das seguintes funções de arquivos?

- | | |
|------------|-----------|
| a) fopen | d) fscanf |
| b) fclose | e) fseek |
| c) fprintf | f) ftell |

34. Escreva um programa que:

- abre um arquivo, lê caractere por caractere e os exibem na tela;
- imprime apenas os caracteres imprimíveis lidos (veja ctype.h);
- imprime linha a linha de um arquivo, truncando linhas maiores que 79 caracteres. Desprezar caracteres após estes até encontrar um fim de linha (\n). Só caracteres imprimíveis devem ser considerados;
- conte o número de linhas, de palavras e de caracteres lidos, apresentando um sumário com esses dados após apresentar o conteúdo do arquivo;
- (para pensar) exiba também a porcentagem das palavras que terminam em "ndo" ou "ndo," ou "ndo.", ou seja, dos gerúndios no texto.

35. Escreva um programa que leia palavras de um arquivo e escreva em outro arquivo cada palavra lida com a ordem das letras invertidas, exceto pela primeira e pela última letra da palavra.

36. Escreva um trecho de programa que inverta a ordem dos elementos de um vetor. Faça o mesmo para inverter a ordem dos elementos de um arquivo de inteiros em campos de 16 caracteres, utilizando fseek.

37. Verifique a construção dos tipos enumerativos do programa abaixo e explique o objetivo deste tipo de estrutura.

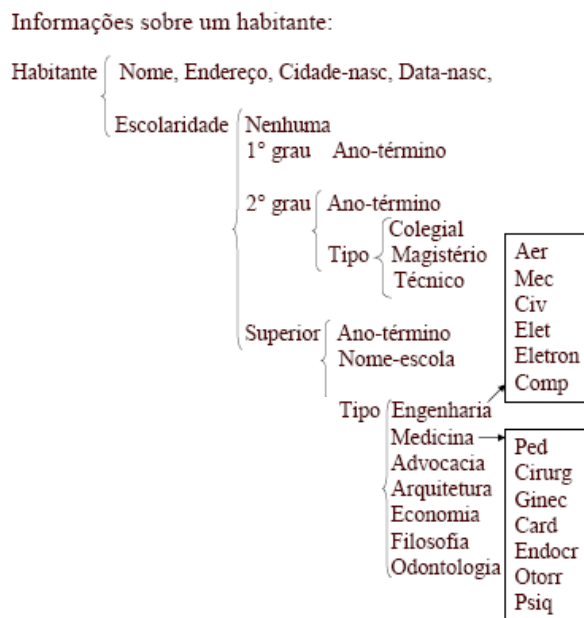
```
#include <stdio.h>
#include <stdlib.h>
#define N 10

enum diasemana {dom, seg, ter, qua, qui, sex, sab};
typedef enum diasemana diasem;

typedef char nome[N];

main () {
    diasem hoje, ontem, amanha;
    nome nomedia[7] = {"domingo","segunda","terca","quarta","quinta",
                      "sexta","sabado"};
    do {
        printf ("Dia de hoje: ");
        scanf ("%d", &hoje);
        if (hoje >= dom && hoje <= sab) {
            ontem = (hoje + 6) % 7;
            amanha = (hoje + 1) % 7;
            printf ("hoje = %s; ontem = %s; amanha = %s;\n\n",
                  nomedia[hoje], nomedia[ontem], nomedia[amanha]);
        }
    } while (hoje >= dom && hoje <= sab);
    system("PAUSE");
}
```

38. Monte a declaração dos tipos enumerativos para este exemplo abaixo:



HELP:

Estagiária: Michelle de Oliveira Parreira

Contato:

Sala 123 IEC

Email: michelle@ita.br