

# Linguagem C

**LIVRO DE EXERCÍCIOS**

V0.9

Professor Ricardo Antonello  
[www.antonello.com.br](http://www.antonello.com.br)

# Sumário

- Capítulo 1: Sistema binário ..... 3
- Capítulo 2: Algoritmos e fluxogramas ..... 5
- Capítulo 3: Introdução à Linguagem C..... 6
- Capítulo 4: Variáveis, operadores e funções de entrada e saída ..... 7
- Capítulo 5: Comandos Condicionais ..... 16
- Capítulo 6: Laços de repetição..... 19
- Capítulo 7: Funções ..... 29
- Capítulo 8: Vetores e Matrizes ..... 31
- Capítulo 9: Arquivos em disco..... 35
- Capítulo 10: Operadores de bits ..... 38

# Capítulo 1: Sistema binário

- 1.1) Qual é a representação binária para o número 7 em decimal?
- 1.2) Qual é a representação binária para o número 16 em decimal?
- 1.3) Qual é a representação binária para o número 34 em decimal?
- 1.4) Qual é a representação binária para o número 66 em decimal?
- 1.5) Qual é a representação binária para o número 127 em decimal?
- 1.6) Qual é a representação binária para o número 222 em decimal?
- 1.7) Qual é a representação binária para o número 3000 em decimal?
- 1.8) Qual é a representação binária para o número 5555 em decimal?
- 1.9) Qual é a representação binária para o número 19765 em decimal?
- 1.10) Qual é a representação hexadecimal para o número 7 em decimal?
- 1.11) Qual é a representação hexadecimal para o número 16 em decimal?
- 1.12) Qual é a representação hexadecimal para o número 23 em decimal?
- 1.13) Qual é a representação hexadecimal para o número 66 em decimal?
- 1.14) Qual é a representação hexadecimal para o número 127 em decimal?
- 1.15) Qual é a representação hexadecimal para o número 222 em decimal?
- 1.16) Qual é a representação hexadecimal para o número 3000 em decimal?
- 1.17) Qual é a representação hexadecimal para o número 5555 em decimal?
- 1.18) Qual é a representação hexadecimal para o número 19765 em decimal?
- 1.19) Qual é a representação decimal para o número 0x00001?
- 1.20) Qual é a representação decimal para o número 0x11?
- 1.21) Qual é a representação decimal para o número 0x3DF?

- 1.22) Qual é a representação decimal para o número 0x350?
- 1.23) Qual é a representação decimal para o número 0x3D45A?
- 1.24) Qual é a representação decimal para o número 0xAAA?
- 1.25) Qual é a representação binária para o número 0x00001?
- 1.26) Qual é a representação binária para o número 0x11?
- 1.27) Qual é a representação binária para o número 0x3DF?
- 1.28) Qual é a representação binária para o número 0x350?
- 1.29) Qual é a representação binária para o número 0x3D45A?
- 1.30) Qual é a representação binária para o número 0xAAA?
- 1.31) Qual é a representação decimal para o número 0b1111?
- 1.32) Qual é a representação decimal para o número 0b001?
- 1.33) Qual é a representação decimal para o número 0b1001001?
- 1.34) Qual é a representação decimal para o número 0b001001001?
- 1.35) Qual é a representação decimal para o número 0b11001?
- 1.36) Qual é a representação decimal para o número 0b001111001?
- 1.37) Qual é a representação decimal para o número 0b1010101010?
- 1.38) Qual é a representação hexadecimal para o número 0b111?
- 1.39) Qual é a representação hexadecimal para o número 0b001?
- 1.40) Qual é a representação hexadecimal para o número 0b1001001?
- 1.41) Qual é a representação hexadecimal para o número 0b001001001?
- 1.42) Qual é a representação hexadecimal para o número 0b11001?
- 1.43) Pesquise sobre como funciona a base octal. Opcionalmente, realize as conversões dos números dos exercícios acima “de” e “para” a base octal.

## Capítulo 2: Algoritmos e fluxogramas

2.1) O que é um algoritmo?

Resposta: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

2.2) Qual é a notação dos fluxogramas para início e fim do fluxo? Desenhe abaixo:

Resposta:

2.3) Qual é a notação dos fluxogramas para entrada e saída? Desenhe abaixo:

Resposta:

2.4) Qual é a notação dos fluxogramas para comandos de decisão (controle/alternativa)?  
Desenhe abaixo:

Resposta:

2.5) Qual é a notação dos fluxogramas para atribuição? Desenhe abaixo:

Resposta:

2.6) Faça um algoritmo (em texto) para

## Capítulo 3: Introdução à Linguagem C

3.1) Qual é a função do compilador na criação de programas em C?

3.2) Porque é necessário utilizarmos uma linguagem de programação? Não seria possível programar diretamente em linguagem de máquina?

3.3) Faça um algoritmo para calcular a nota necessária no exame, sendo que a entrada é a média. A formula para calcular a nota necessária no exame é:  $\text{Nota Exame} = 50 - (6 * \text{Média}) / 4$

3.4) Faça um fluxograma para o algoritmo do exercício anterior.

3.5) Faça um algoritmo (em linguagem natural) que recebe três números, determina qual é o maior dentre eles e apresenta este número.

1.1) Faça um fluxograma para o algoritmo do exercício anterior.

3.6) Para desenvolver um programa de computador é necessário elaborar uma sequencia lógica de instruções de modo que estas instruções resolvam o problema em questão. O conjunto de instruções que podem ser utilizadas para resolver os problemas é definido pela linguagem de programação utilizada. Para facilitar a tarefa dos programadores as linguagens de programação atribuem nomes compreensíveis a suas instruções, e estas instruções são então convertidas para a linguagem do computador. Sobre esse processo é correto afirmar:

- a) O compilador tem a função de converter linguagem de máquina em código de alto nível.
- b) Instruções em linguagem de alto nível são compreensíveis aos seres humanos.
- c) Um programa em linguagem de máquina é reconhecido apenas por computadores com sistema Windows.
- d) Programas em linguagens de programação como a Linguagem C não precisam ser compilados antes de serem executados.
- e) Nenhuma das alternativas esta correta.

3.7) Marque a opção que em que todos os nomes de variáveis são válidos. Os nomes estão separados por vírgulas.

- a) real\$, errado!, correta
- b) casa, olho gordo, valor
- c) terminou, int, k
- d) case, x, casa
- e) Nenhuma das alternativas

## Capítulo 4: Variáveis, operadores e funções de entrada e saída

4.1) Quais as regras básicas para a criação de nomes de identificadores na linguagem C?

Resposta: \_\_\_\_\_

---

---

4.2) Quais são as duas formas de fazer um comentário nos programas em C?

Resposta: \_\_\_\_\_

---

---

4.3) Construa um programa que apresenta na tela do computador o seu nome.

Resposta: \_\_\_\_\_

---

---

4.4) Descreva o funcionamento do operador % em duas situações: 1) Como resto nas operações matemáticas. 2) Como máscara no *printf* ou *scanf*.

Resposta: \_\_\_\_\_

---

---

4.5) Assinale a alternativa correta sobre os tipos básicos de dados na Linguagem C:

- I. char: Caractere: O valor armazenado é um caractere. Caracteres geralmente são armazenados em códigos (usualmente o código ASCII<sup>1</sup>).
- II. int: Número inteiro é o tipo padrão e o tamanho do conjunto que pode ser representado normalmente depende da máquina em que o programa está rodando.
- III. float: Número em ponto flutuante de precisão simples. São conhecidos normalmente como números reais.
- IV. void: Este tipo serve para indicar que um resultado não tem um tipo definido. Uma das aplicações deste tipo em C é criar um tipo vazio que pode posteriormente ser modificado para um dos tipos anteriores.

---

<sup>1</sup> ASCII (American Standard Code for Information Interchange ou Código Padrão Americano para o Intercâmbio de Informação)

- a) Somente I.
- a) Somente I e II.
- b) Somente III.
- c) Somente I, II e III.
- d) I, II, III e IV correspondem a tipos básicos de dados da Linguagem C.

4.6) Dados o código abaixo que compila e executa sem gerar erros, assinale a alternativa correta:

```
int main() {
    int a=7, b=4, soma;
    soma = a + b;
    printf("%i", soma);
    return(0);
}
```

- a) O resultado em tela deverá ser o número 11.
- b) O resultado em tela deverá ser o número 10.
- c) O resultado em tela deverá ser o número 9.
- d) O resultado em tela deverá ser o número 2.
- e) O resultado em tela deverá ser o número 28.

4.7) São regras para criar nomes de variáveis válidos em Linguagem C.

- I. São aceitos somente letras, números e o caractere “\_”.
- II. Deve começar com letras ou com o caractere “\_”.
- III. Podem possuir números desde que não como primeiro caractere.
- IV. Não podem ter espaços.

- a) Somente I.
- b) Somente I e II.
- c) Somente III.
- d) Somente I, II e III.
- e) I, II, III e IV.

4.8) São exemplos válidos de nomes de variáveis:

- I. Idade
- II. idade1
- III. idade\_1
- IV. \_idade
- V. \_idade1
- VI. idade01
- VII. \_idade01
- VIII. melhor\_idade

- a) Somente I.
- b) Somente I e II e III.
- c) Somente III e IV
- d) Somente I, II e III e IV.
- e) Todos os itens de I até VIII.

4.9) Sobre a afirmação abaixo assinale a alternativa correta.

*Uma convenção na linguagem C é que todos os nomes de variáveis deve possuir apenas letras minúsculas ou ao menos iniciar com uma letra minúscula, pois letras maiúsculas são reservadas para nomes de “constantes” na linguagem C. Por se tratar de convenção, o código irá funcionar normalmente se você utilizar letras maiúsculas, contudo, não é recomendado por dificultar a leitura do código por outros programadores.*

- a) A afirmação é verdadeira.
- b) A afirmação é falsa.

#### 4.10) Sobre a afirmação abaixo assinale a alternativa correta.

*O tipo boolean ou booleano é um tipo de dados que armazena apenas dois valores, verdadeira ou falso. Contudo, na linguagem C padrão ANSI não existem variáveis do tipo booleano, então consideramos valores diferentes de zero como verdadeiros (true) e iguais a zero como falso (false).*

- a) A afirmação é verdadeira.
- b) A afirmação é falsa.

#### 4.11) Sobre a afirmação abaixo assinale a alternativa correta.

*Um tipo de dados importante em C é o tipo "char". Ele é na verdade um tipo de dados numérico, com tamanho de 8 bits, ou seja, 1 byte e consequentemente com capacidade para armazenar valores inteiros de 0 a 255. Contudo, em C utilizamos a tabela de código ASCII - American Standard Code for Information Interchange que atribui um símbolo para cada número de 0 a 255. Dessa forma, a letra "A" é representada pelo número 65, a letra "B" pelo 66. Já o letra "a" em caixa baixa, ou seja, minúscula é representada pelo número 97, a letra "b" por 98 e assim por diante.*

- a) A afirmação é verdadeira.
- b) A afirmação é falsa.

#### 4.12) Sobre "range" de variáveis responda.

- I. O tipo char possui 1 byte e armazena de -128 até 127.
- II. Com o modificador *unsigned* o tipo char armazena de 0 a 255.
- III. O tipo char é um tipo inteiro apesar de ser utilizado para armazenar caracteres.
- IV. É possível fazer contas somando os valores inteiros de duas variáveis do tipo char.

- a) Somente I.
- b) Somente I e II.
- c) Somente III.
- d) Somente I, II e III.
- e) I, II, III e IV.

Qual o limite máximo e mínimo para cada um dos tipos de dados a seguir?

- a) unsigned char
- b) float
- c) unsigned int
- d) int

#### 4.13) Quais dos seguintes nomes de identificadores são válidos e quais são inválidos? Justifique?

- a) Variavel           ( ) Válido   ( ) Inválido
- b) nome 1           ( ) Válido   ( ) Inválido
- c) entrada2       ( ) Válido   ( ) Inválido
- d) saida\_2         ( ) Válido   ( ) Inválido
- e) 1contador       ( ) Válido   ( ) Inválido
- f) \_valor\_TOTAL   ( ) Válido   ( ) Inválido

#### 4.14) Sobre a função printf() considere as afirmações abaixo e assinale a alternativa que corresponde as afirmações corretas.

- I. %c imprime o conteúdo da variável com representação ASCII;
- II. %d ou %i imprime o conteúdo da variável com representação decimal com sinal;
- III. %u imprime o conteúdo da variável com representação decimal sem sinal;

- IV. %o imprime o conteúdo da variável com representação octal sem sinal;  
 V. %x imprime o conteúdo da variável com representação hexadecimal sem sinal.
- a) Somente I e IV.  
 b) Somente I e II e V.  
 c) Somente III e V.  
 d) Somente I, II e III.  
 e) I, II, III e IV e V.

4.15) Sobre a função printf() considere as afirmações abaixo e assinale a alternativa que corresponde as afirmações corretas.

- I. printf(“%f”); imprime 1.230000  
 II. printf(“%.3f”); imprime 1.230  
 III. printf(“%.2f”); imprime 1.23  
 IV. printf(“%.1f”); imprime 1.2  
 V. printf(“%6.2f”); imprime 1.23 colocando espaços à esquerda para que no total, o número tenha 6 caracteres, contando o ponto decimal como um desses caracteres.

Assinale a alternativa correta:

- a) Somente I e IV.  
 b) Somente I e II e V.  
 c) Somente III e V.  
 d) Somente I, II e III.  
 e) I, II, III e IV e V.

4.16) Sobre a função printf() considere as afirmações abaixo e assinale a alternativa que corresponde as afirmações corretas.

- I. printf(“%06.2f”); imprime 001.23 colocando espaços à esquerda para que no total, o número tenha 6 caracteres, contando o ponto decimal como um desses caracteres.  
 II. printf(“%+06.2f”); imprime +01.23 colocando sinais positivos quando o número é positivo. Isso não interfere em nada quando o número é negativo, pois neste caso sempre aparece o sinal – antes do número, conforme seria esperado.  
 III. printf(“%+08.2f”, 1.23); imprime: +0001.23

Assinale a alternativa correta:

- a) Somente I.  
 b) Somente I e II.  
 c) Somente I e III.  
 d) Somente II e III.  
 e) Todas são verdadeiras.

4.17) Sobre a função printf() considere as afirmações abaixo e assinale a alternativa que corresponde as afirmações corretas.

Afirmação	Comando	Comportamento ou caracter representado
I.	\a	Beep (Aviso sonoro)
II.	\b	Backspace (Apaga o caracter anterior)
III.	\f	<a href="#">Formfeed (avança linha)</a>
IV.	\n	<a href="#">Newline/Line Feed (Equivalente ao [Enter])</a>
V.	\\	<a href="#">Backslash (Imprimir uma barra invertida)</a>
VI.	\'	<a href="#">Single quotation mark (Imprime aspas simples)</a>
VII.	\"	<a href="#">Double quotation mark (Imprime aspas duplas)</a>

Assinale a alternativa correta:

- a) Somente I e IV.
- b) Somente I e II.
- c) Somente I, II, III e VII.
- d) Somente II e III, VI e VII.
- e) Todas são verdadeiras.

4.18) Qual é a saída do programa abaixo?

```
#include <stdio.h>
int main(void) {
    float result, a=1, b=2, c=3, d=7, e=8;
    result=(b-(c*7))/(4-(a*c*a));
    printf("%f",result);
    return 0;
}
```

- a) -19.000000
- b) -48.000000
- c) -22.000000
- d) -22.00
- e) 22.000000

4.19) Qual é a saída do programa abaixo?

```
#include <stdio.h>
int main(void) {
    int i=10, i2=18;
    printf("%f", ((++i+i2++)/2.0));
    return 0;
}
```

- a) 14.000000
- b) 14.500000
- c) 15.000000
- d) 14.0
- e) 14.5

4.20) Referente ao código abaixo:

```
int x=7;
printf("%d",++x%2);
```

- a) Imprime 6.
- b) Imprime 7.
- c) Imprime 8.
- d) Imprime 1.
- e) Imprime 0.

4.21) Qual é saída do seguinte programa?

```
int main() {
    int x;
    int y;
    int z;
    x=10;
    y=-20;
    z=x-y;
    printf("Resultado 1 = %d\n",z);
    z=y+x;
```

```

printf("Resultado 2 = %d\n",z);
z=z/(x/5);
printf("Resultado 3 = %d\n",z);
return(0);
}

```

4.22) Qual será a saída no terminado referente a execução do código abaixo?

```

#include <stdio.h>
#include <stdlib.h>
int main(){
    char a=97;
    float x = 2.3;
    int resultado = x+a;
    printf("%.2f\n", resultado);
    return 0;
}

```

- a) 99.3
- b) 97.3
- c) 99
- d) 99.30
- e) Nenhuma das alternativas

4.23) Qual será a saída no terminado referente a execução do código abaixo?

```

#include <stdio.h>
#include <stdlib.h>
int main(){
    char a=97;
    float x = 2.3;
    int resultado = x+a;
    printf("%.2f\n", resultado);
    return 0;
}

```

- a) 99.3
- b) 97.3
- c) 99
- d) 99.30
- e) Nenhuma das alternativas

4.24) Qual é a saída do programa abaixo se informadas as notas: 6, 8 e 10?

```

#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[]) {
    //Media de 3 notas
    float n1, n2, n3, media=0;
    puts("Digite a primeira nota:");
    scanf("%f",&n1);
    puts("Digite a segunda nota:");
    scanf("%f",&n2);
    puts("Digite a terceira nota:");
    scanf("%f",&n3);
    printf("Media: %f", media);
    return 0;
}

```

- a) Média: 0.000000
- b) Média: 8.000000
- c) Média: 8.0
- d) 8.0
- e) 0.000000

4.25) Faça um programa que lê três números reais do teclado e apresenta na tela a média destes números.

4.26) Faça um programa para ler dois números inteiros, x e y, e imprimir o quociente e o resto da divisão inteira entre eles.

4.27) Escreva abaixo o código de um programa que calcule a média de quilômetros feitos com cada litro de combustível. Solicite a entrada de dados com quilômetros e litros e depois exiba o cálculo.

4.28) Faça um programa para resolver a equação:  $x = (32 - y) / (2a + b)$ . Sendo  $a = 10$ ,  $b = 2$  e  $y = 5$ .

4.29) Escreva um programa para ler uma temperatura em graus Celsius, calcular e escrever o valor correspondente em graus Fahrenheit.

$$^{\circ}\text{F} = ^{\circ}\text{C} \times 1,8 + 32$$

4.30) Escreva um programa para ler o raio de um círculo, calcular e escrever a sua área.

$$A = \pi r^2$$

4.31) Faça um programa que pede para o operador digitar uma letra, um número inteiro e um número real. Em seguida o programa deve ler estes dados, armazená-los nos tipos de dados adequados e imprimi-los na tela.

4.32) Faça um Programa que pergunte quanto você ganha por hora e o número de horas trabalhadas no mês. Calcule e mostre o total do seu salário no referido mês, sabendo-se que são descontados 14% para o Imposto de Renda, 11% para o INSS e 1% para o sindicato, faça um programa que nos dê:

- a. salário bruto.
  - b. quanto pagou ao INSS.
  - c. quanto pagou ao sindicato.
  - d. o salário líquido.
  - e. calcule os descontos e o salário líquido, conforme a tabela abaixo:
  - f. + Salário Bruto : R\$
  - g. - IR (14%) : R\$
  - h. - INSS (11%) : R\$
  - i. - Sindicato (1%) : R\$
  - j. = Salário Líquido : R\$
- Obs.: Salário Bruto - Descontos = Salário Líquido.

4.33) Faça um Programa que receba um valor de combustível, calcule e mostre o valor desse mesmo combustível saído da refinaria e todos os valores incluídos no preço final, com base nos seguintes percentuais de despesas e impostos (Referência: <http://www.petrobras.com.br/pt/produtos/composicao-de-precos/>)

- 17% - Distribuição e Revenda
- 12% - Custo Etanol Anidro
- 28% - ICMS
- 7% - CIDE, PIS/PASEP e COFINS

A saída do programa deve obedecer ao seguinte formato:

```
+ Valor na bomba: R$ <valor>
- Distribuição e Revenda (17%): R$ <valor>
- Custo Etanol Anidro (12%): R$ <valor>
- ICMS (28%): R$ <valor>
- CIDE, PIS/PASEP e COFINS (7%): R$ <valor>
= Valor da refinaria: R$ <valor>
```

Exemplo de programa

Entrada:  
3.00

Saída:

```
+ Valor na bomba: R$ 3.00
- Distribuição e Revenda (17%): R$ 0.51
- Custo Etanol Anidro (12%): R$ 0.36
- ICMS (28%): R$ 0.84
- CIDE, PIS/PASEP e COFINS (7%): R$ 0.21
= Valor da refinaria: R$ 1.08
```

**Importante:** esta saída corresponde ao exemplo de entrada acima e quando for verificado o programa, serão utilizados outros valores.

4.34) O cardápio de uma lanchonete é dado abaixo. Prepare um algoritmo que leia a quantidade de cada item que você consumiu e calcule a conta final.

X-salada.....	R\$5.50
X-Bacon.....	R\$7.25
X-Tudo.....	R\$10.30
X-Egg.....	R\$7.00
Cerveja.....	R\$4.50
Refrigerante.....	R\$6.00

A saída deve apresentar:

Comprovante			
Item.....	QTD.....	ULR..	TOTAL
X-salada:.....	2	R\$ 5.50...	R\$11.00
X-bacom:.....	2	R\$ 7.25...	R\$14.50
X-tudo:.....	2	R\$ 10.30...	R\$20.60
Refrigerante:.....	2	R\$ 6.00...	R\$12.00
X-Egg:.....	2	R\$ 7.00...	R\$14.00
Cerveja:.....	2	R\$ 4.50...	R\$9.00
Total a pagar.....			R\$81.10

4.35) Faça um programa para uma loja de tintas. O programa deverá pedir o tamanho em metros quadrados da área a ser pintada. Considere que a cobertura da tinta é de 1 litro para cada 3 metros quadrados e que a tinta é vendida em latas de 18 litros, que custam R\$ 80,00. Informe ao usuário a quantidades de latas de tinta a serem compradas e o preço total.

4.36) Escreva um programa que calcule a fórmula de Bhaskara. As entradas serão a, b, e c referentes a uma equação de segunda grau do tipo  $ax^2+bx+c$  e a saída serão as duas raízes  $r_1$  e  $r_2$ .

4.37) Escreva um programa que recebe como entrada a quantidade de maçãs para o transporte e o preço de cada maçã. Como resultado o programa deve informar o valor total da carga e a quantidade de caixas de maçã necessárias para o transporte. Considere que uma caixa de maçãs possui espaço para 48 unidades.

4.38) Escreva um programa para ler as dimensões de uma cozinha retangular (comprimento, largura e altura), calcular e escrever a quantidade de caixas de azulejos para se colocar em todas as suas paredes (considere que não será descontada a área ocupada por portas e janelas). Cada caixa de azulejos possui 3 m<sup>2</sup>.

4.39) Escreva um programa que leia um número inteiro e calcule e mostre a sua decomposição em unidade, dezena, centena e milhar. Considere que o número máximo recebido via teclado será de 9999. Exemplo: A entrada 8531 terá a saída: unidade = 1 dezena = 3 centena = 5 milhar = 8.

4.40) Escreva um programa para calcular e imprimir o número de lâmpadas necessárias para iluminar um determinado cômodo de uma residência. Dados de entrada: a potência da lâmpada utilizada (em watts), as dimensões (largura e comprimento, em metros) do cômodo. Considere que a potência necessária é de 18 watts por metro quadrado.

4.41) A equipe Benneton-Ford deseja calcular o número mínimo de litros que deverá colocar no tanque de seu carro para que ele possa percorrer um determinado número de voltas até o primeiro reabastecimento. Escreva um programa que leia o comprimento da pista (em metros), o número total de voltas a serem percorridas no grande prêmio, o número de reabastecimentos desejados e o consumo de combustível do carro (em Km/L). Calcular e escrever o número mínimo de litros necessários para percorrer até o primeiro reabastecimento. OBS: Considere que o número de voltas entre os reabastecimentos é o mesmo.

4.42) Faça um Programa para uma loja de tintas. O programa deverá pedir o tamanho em metros quadrados da área a ser pintada. Considere que a cobertura da tinta é de 1 litro para cada 6 metros quadrados e que a tinta é vendida em latas de 18 litros, que custam R\$ 80,00 ou em galões de 3,6 litros, que custam R\$ 25,00. Informe ao usuário as quantidades de tinta a serem compradas e os respectivos preços em 3 situações. Sempre arredonde os valores para cima, isto é, considere latas cheias:

- I. comprar apenas latas de 18 litros;
- II. comprar apenas galões de 3,6 litros;
- III. misturar latas e galões, de forma que o preço seja o menor.

## Capítulo 5: Comandos Condicionais

5.1) Qual é a saída do programa abaixo?

```
int main()
{
    float result, a=1, b=2, c=3, d=7, e=8;
    result=(b-(c*7))/(4-(a*c*a));
    if(result >= -19)
        printf("Ok");
    else
        printf("%.2f", result);
    return 0;
}
```

- a) -19.000000
- b) -48.000000
- c) -22.000000
- d) 22.00
- e) Ok

5.2) Escreva um programa para ler 2 valores (considere que não serão informados valores iguais) e escrever o maior deles.

5.3) Escreva um programa para ler o ano de nascimento de uma pessoa e escrever uma mensagem que diga se ela poderá ou não votar este ano (não é necessário considerar o mês em que ela nasceu).

5.4) Escreva um programa que verifique a validade de uma senha fornecida pelo usuário. A senha válida é o número 1234. Devem ser impressas as seguintes mensagens:

ACESSO PERMITIDO caso a senha seja válida.  
 ACESSO NEGADO caso a senha seja inválida.

5.5) Faça um programa que lê um número inteiro de 1 a 7 e imprime o dia da semana, se o número estiver fora deste intervalo o programa deve imprimir uma mensagem de erro.

5.6) Faça um programa que lê três números do teclado, descubra qual deles é o menor e imprime este número na tela.

5.7) Faça um programa que lê três notas (números reais), calcula a média entre elas, imprime a média, testa se a média é menor que 7, se for imprime "Aluno em exame!" e se não for imprime "Aluno aprovado!"

5.8) Faça um programa que lê três notas (números reais), calcula a média entre elas, imprime a média, testa se a média é menor que 7, se for imprime "Aluno em exame!" e se não for imprime "Aluno aprovado!". Se o aluno estiver em exame, peça para o usuário informar a nota do exame, calcule a fórmula (média das notas + exame)/2. Se o resultado for maior ou igual a 5 informe "Aluno passou com exame", senão informe "Aluno reprovado".

5.9) Escreva abaixo o código de um programa que calcule a média de quilômetros feitos com cada litro de combustível. Solicite a entrada de dados com quilômetros e litros e depois exiba o cálculo da média e caso seja menor que 6 km/l informe que é preciso trocar o carro por um mais econômico.

5.10) Tendo como entrada a altura e o sexo (codificado da seguinte forma: 1:feminino 2:masculino) de uma pessoa, construa um programa que calcule e imprima seu peso ideal, utilizando as seguintes

Fórmulas:

- para homens:  $(72.7 * \text{Altura}) - 58$
- para mulheres:  $(62.1 * \text{Altura}) - 44.7$

5.11) Escreva um programa que leia as medidas dos lados de um triângulo e escreva se ele é Equilátero, Isósceles ou Escaleno. Sendo que:

- Triângulo Equilátero: possui os 3 lados iguais.
- Triângulo Isóscele: possui 2 lados iguais.
- Triângulo Escaleno: possui 3 lados diferentes.

5.12) Escreva um programa que leia o valor de 3 ângulos de um triângulo e escreva se o triângulo é Acutângulo, Retângulo ou Obtusângulo. Sendo que:

- Triângulo Retângulo: possui um ângulo reto. (igual a  $90^\circ$ )
- Triângulo Obtusângulo: possui um ângulo obtuso. (maior que  $90^\circ$ )
- Triângulo Acutângulo: possui três ângulos agudos. (menor que  $90^\circ$ )

5.13) Escreva um programa para ler o número de lados de um polígono regular e a medida do lado (em cm). Calcular e imprimir o seguinte:

- Se o número de lados for igual a 3 escrever TRIÂNGULO e o valor da área
- Se o número de lados for igual a 4 escrever QUADRADO e o valor da sua área.
- Se o número de lados for igual a 5 escrever PENTÁGONO.
- Caso o número de lados seja superior a 5 escrever POLÍGONO NÃO IDENTIFICADO.

5.14) Quais expressões abaixo resultariam em resultado verdadeiro “true” se incluídas em um comando condicional “if” após a seguinte declaração de variáveis: `int x=0, y=10;`

- a) `x<10 && y>10`
- b) `x<=10 && y<10`
- c) `x<0 || y>10`
- d) `x>0 && y>=10`
- e) `x<=0 || y<10`

5.15) Escreva um programa para ler 3 valores inteiros (considere que não serão lidos valores iguais) e escrevê-los em ordem crescente.

5.16) Faça um programa que pede ao operador para digitar a temperatura ambiente e utilizando comandos “if” ou “if-else”, faz o seguinte. Se a temperatura for maior ou igual a 40, imprime “Muito quente”, se a temperatura menor que 40 e maior ou igual a 30, imprime “Quente”, se a temperatura for menor que 30 e maior ou igual a 20, imprime “Agradável” e se a temperatura for menor que 20, imprime “Frio”

5.17) Escreva um programa para ler 2 valores inteiros e uma das seguintes operações a serem executadas (codificada da seguinte forma: 1. Adição, 2. Subtração, 3. Divisão, 4. Multiplicação). Calcular e escrever o resultado dessa operação sobre os dois valores lidos. Observação: Considere que só serão lidos as operações 1, 2, 3 ou 4.

5.18) Faça um programa utilizando o comando “switch”. O programa deve se comportar como uma calculadora, onde o operador escolhe entre as operações soma (+), subtração (-), multiplicação (\*), divisão (/), potência (^), e raiz quadrada (r). O programa deve apresentar todas as opções no início e o operador escolhe a operação desejada digitando o símbolo. O programa deve então ler os dois números e realizar a operação.

5.19) Verifique o programa abaixo:

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

int main(){
    int a, b, c;
    printf("digite a:");
    scanf("%d", &a);
    printf("digite b:");
    scanf("%d", &b);
    printf("digite c:");
    scanf("%d", &c);
    if (a >= b){
        if (a >= c && b >= c)
            printf("%d %d %d\n", c, b, a);
        if (a >= c && b <= c)
            printf('%d %d %d\n", b, c, a);
    }
    if (b >= a){
        if (b >= c && a >= c)
            printf('%d %d %d\n", c, a, b);
        if (b >= c && a <= c)
            printf("%d %d %d\n", a, c, b):
    }
    if (c >= a){
        if (c >= b && b >= a)
            printf("%d %d %d\n", a, b, c);
        if (c >= b && b <= a)
            printf('%d %d %d\n", b, a, c);
    }
    return 0;
}
```

O que será impresso por esse programa:

- O programa funciona e imprime os números na ordem decrescente;
- O programa funciona e imprime o menor número;
- O programa funciona e imprime os números na ordem decrescente;
- O programa funciona e imprime o maior número;
- Nenhuma das alternativas;

5.20) Faça um programa que lê do teclado um número do tipo unsigned char (valores de 0 a 255) e apresenta na tela o valor dos oito bits que compõem este número. Por exemplo, de a entrada for 12 a saída deve ser 00001100. (utilize o comando “if – else” para determinar o valor de cada bit)

## Capítulo 6: Laços de repetição

6.1) Faça um programa utilizando o comando “for” que apresenta a sequência de números de 5 até 20. Refaça o código utilizando o comando “while”.

6.2) Modifique o programa anterior para que a sequência apresentada seja de 20 a 5.

6.3) Faça um programa que lê dois números inteiros do teclado, testa se o primeiro é menor que o segundo, se for imprime todos os números do intervalo entre eles.

6.4) Escreva um programa que imprima todos os números pares entre 0 e 50 e em seguida imprima todos os números ímpares. Deixe um espaço entre os números.

6.5) Para que a divisão entre 2 números possa ser realizada, o divisor não pode ser nulo (zero). Escreva um programa para ler 2 valores e imprimir o resultado da divisão do primeiro pelo segundo. OBS: O programa deve validar a leitura do segundo valor (que não deve ser nulo). Enquanto for fornecido um valor nulo a leitura deve ser repetida. Utilize a estrutura de repetição na construção da repetição de validação.

6.6) Escreva abaixo um programa que imprime de 1 (inclusive) a 100 (inclusive) na tela mas que não imprime o número 50, ou seja, após imprimir o 49 a próxima saída deverá ser 51.

6.7) Utilizando o comando “for”, faça um programa que imprime de 1 a 10 mil na tela, um abaixo do outro, com exceção dos que são divisíveis por 10.

6.8) Crie um programa que recebe números inteiros do usuário, imprime o quadrado do número recebido e solicita o próximo número. Caso o número informado seja zero o programa deve finalizar.

6.9) Dado um número inteiro positivo  $n$ , calcular a soma dos  $n$  primeiros números naturais.

6.10) Dado um número inteiro positivo  $n$ , imprimir os  $n$  primeiros naturais ímpares. Exemplo: Para  $n=4$  a saída deverá ser 1,3,5,7.

6.11) Dados  $n$  e uma sequência de  $n$  números inteiros, determinar a soma dos números pares.

6.12) Escreva abaixo o código de um programa que calcule a média de quilômetros feitos com cada litro de combustível de uma viagem. O programa deve solicitar a entrada de dados com quilômetros e litros e depois exibir o cálculo informando que o usuário deve procurar um carro “mais econômico” caso a média seja inferior a 10 km/litro. Após o primeiro cálculo o programa deve solicitar ao usuário que informe o número 1 para executar um novo cálculo ou 0 para sair do programa.

6.13) Faça um programa utilizando ‘while’ que apresenta a tabuada do 7 na tela.

- 6.14) Faça um programa utilizando 'for' que apresenta a tabuada do 3 na tela.
- 6.15) Faça um programa que apresenta as tabuadas de 1 a 10 utilizando dois laços de repetição um dentro do outro. Utilize 'while'.
- 6.16) Faça um programa que apresenta as tabuadas de 1 a 10 utilizando dois laços de repetição um dentro do outro. Utilize 'for'.
- 6.17) Utilizando os comandos "for" e "if", faça um programa que lê 5 valores do teclado e informa quantos deles são negativos e quantos são positivos.
- 6.18) Escreva um programa que lê 10 números reais. O programa deve imprimir a média, o maior e o menor dos números.
- 6.19) Faça um programa utilizando o comando "while", que fica lendo números inteiros do teclado e só finaliza quando o número digitado for 10 ou após 5 iterações, ou seja, após 5 números serem digitados.
- 6.20) Utilizando o comando "while" faça um programa que calcula o valor de x elevado a y, onde x é um número real lido do teclado e y é um número inteiro lido do teclado.
- 6.21) Faça um programa que apresenta na tela toda a tabela ASCII, ou seja, apresenta a tela todos os caracteres cujos valores estão entre 0 e 255.
- 6.22) Escreva um programa que lê um número inteiro do teclado e imprime todos os seus divisores.
- 6.23) Faça um programa que lê um número real do teclado e utilizando o comando "while" calcula o fatorial deste número. (ex: 5! é igual a  $5 * 4 * 3 * 2 * 1$ ).
- 6.24) Utilizando o comando "for", faça um programa que imprime de 1 a 10 mil na tela, um abaixo do outro, com exceção dos que são divisíveis por 33.
- 6.25) As laranjas custam R\$ 0,60 cada se forem compradas menos do que uma dúzia, e R\$ 0,50 se forem compradas pelo menos doze. Escreva um programa que leia o número de laranjas compradas, calcule e escreva o valor total da compra. Ao final, peça se o usuário quer realizar novo cálculo.
- 6.26) Faça um programa que imprime a figura abaixo utilizando laços de repetição.

```
*  
**  
***  
****  
*****
```

Restrição: use pra imprimir na tela somente os comandos `printf("**");` ou `printf("\n");`

6.27) Altere o programa anterior para imprimir a figura abaixo:

```

****1
***22
**333
*4444
55555

```

Dica: ao invés de usar `printf("**");` use `printf("%d",i);` onde `i` é a variável de contagem. Perceba que você deve gerar as sequências 1 2 3 4 5 (para números) e 4 3 2 1 0 (para \*).

Sugestão: como no anterior, faça primeiro um programa que, dado um número pelo teclado, imprime asteriscos e o número. Por ex.: o usuário digita 3 e o programa imprime `**333` (2 \*'s e 3 números)

6.28) Faça um programa que lê um número do teclado, e utilizando dois comandos “while” apresenta na tela a seguinte matriz, onde `n` é o número lido e cada elemento da matriz é o resultado da operação indicada. A matriz deve possuir o valor zero nas primeiras e últimas linhas e colunas.

```

N+1 N+2 N+3 N+4
N+2 N+3 N+4 N+5
N+3 N+4 N+5 N+6
N+4 N+5 N+6 N+7

```

Exemplo: para `n = 10`, temos:

```

0      0      0      0
0     13     14     0
0     14     15     0
0      0      0      0

```

6.29) Crie um programa para verificar se um número dado é primo. Utilize apenas números inteiros.

6.30) Escrever um programa que calcula e escreve a soma dos números primos entre 92 e 1478.

6.31) Escreva um método que recebe dois números reais `a` e `b` e retorna a soma de todos os números primos existentes entre esses dois.

6.32) Escrever um programa que lê um valor `i` inteiro e positivo e que calcula e escreve o valor da constante de Euler (`e`), com aproximação de `i` termos. A fórmula a seguir calcula o valor de (`e`), e o resultado deve ser 2,718282. O programa deve ficar se repetindo enquanto o operador não escolher `i = 0`;

$$e = \sum_{i=0}^{\infty} \frac{1}{i!}$$

6.33) Faça um programa utilizando o comando “while”, que recebe uma entrada que é o valor do lucro de um investidor em Fundo de Renda Fixa e calcula como saída o imposto de renda (IR) que é de 20% sobre o lucro. Ao final peça se o usuário quer realizar novamente o cálculo informando ‘s’ para continuar ou ‘n’ para sair.

6.34) Escreva um programa para imprimir as letras de A a Z.

6.35) Escreva um programa para imprimir as letras de AA a ZZ na mesma sequência existente em planilhas eletrônicas como Microsoft Excel ou LibreOffice Calc.

6.36) Faça um programa utilizando o comando “do – while”, este programa deve ficar lendo uma letra por vez do teclado até que o operador digite a letra x. Após o operador digitar x o programa deve apresentar na tela o número de vezes que o operador digitou dois caracteres iguais em sequência.

6.37) Faça um programa que apresente quatro opções: (a) consulta saldo, (b) saque, (c) depósito e (d) sair. O saldo deve iniciar em R\$ 0,00. A cada saque ou depósito o valor do saldo deve ser atualizado e apresentado na tela.

6.38) Faça um programa utilizando o comando “switch”. O programa deve se comportar como uma calculadora, onde o operador escolhe entre as operações soma (+), subtração (-), multiplicação (\*), divisão (/), potência (^), e raiz quadrada (r). O programa deve apresentar todas as opções no início e o operador escolhe a operação desejada digitando o símbolo entre parênteses. O programa deve então ler os dois números, realizar a operação e reiniciar o processo até que o operador escolha uma operação inválida (qualquer outro símbolo ou letra).

6.39) Dados n e dois números inteiros positivos i e j diferentes de 0, imprimir em ordem crescente os n primeiros naturais que são múltiplos de i ou de j e ou de ambos. Exemplo: Para n = 6 , i = 2 e j = 3 a saída deverá ser : 0,2,3,4,6,8.

6.40) Dizemos que um inteiro positivo n é perfeito se for igual à soma de seus divisores positivos diferentes de n. Exemplo: 6 é perfeito, pois 1+2+3 = 6. Dado um inteiro positivo n, verificar se n é perfeito.

6.41) Escreva um programa que apresente a série Fibonacci até o décimo sétimo termo. A sequência de Fibonacci é uma sucessão de números obtidos pela soma dos anteriores, a fórmula é sintetizada conforme segue:

$$F(n) = \begin{cases} 0, & \text{se } n=0 \\ 1, & \text{se } n=1 \\ F(n-1) + F(n-2), & \text{outros casos} \end{cases}$$

Resultante em 0, 1, 1, 2, 3, 5, 8, 13, ...

6.42) O matemático alemão Gottfried Leibniz estabeleceu a fórmula matemática para aproximar o valor de  $\pi$ :

$$\pi = 4 \times \left( \frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \frac{1}{11} + \frac{1}{13} \dots \right)$$

Em notação de somatório:

$$\pi = 4 \times \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1}$$

Faça um programa que aproxime o valor de PI (como não é possível calcular até o infinito, crie uma variável com valor elevado)

6.43) Qual sequência de comandos abaixo, quando executada, resulta no processamento de 10 repetições dos comandos dentro do bloco { ... }?

- a) `int i=0; while(i++<9) { ... }`
- b) `int i=1; while(i++<9) { ... }`
- c) `int i=0; while(i++<10) { ... }`
- d) `int i=0; while(i++<=10) { ... }`
- e) `int i=1; while(++i<=10) { ... }`

6.44) Em relação a saída do programa abaixo se informadas as notas: 5, 7 e 9, assinale a alternativa correta:

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[]) {
    //Media de 3 notas
    float n1, n2, n3, media=0;
    puts("Digite a primeira nota:");
    scanf("%f",&n1);
    puts("Digite a segunda nota:");
    scanf("%f",&n2);
    puts("Digite a terceira nota:");
    scanf("%f",&n3);
    if (media > 7) {
        media = (n1 + n2 + n3) / 3;
        printf("Aprovado. Media: %f", media);
    } else {
        printf("Reprovado. Media: %f", media);
    }
    return 0;
}
```

- a) Imprime "Aprovado. Media: 7.000000"
- b) Imprime "Reprovado. Media: 7.000000"
- c) Imprime "Aprovado. Media: 0.000000"
- d) Imprime "Reprovado. Media: 0.000000"
- e) Nenhuma das alternativas anteriores.

6.45) Considere o seguinte programa:

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

int main(){
    int a=7;
    for(a=7;a>8;--a)
        printf("%d", a);
    if(a>7)
        printf("%d", --a);
    return 0;
}
```

O que será impresso por este programa?

- a) 78
- b) 7
- c) 67
- d) 6
- e) Nada é impresso na tela

6.46) O código abaixo compila e executa sem erros de sintaxe ou compilação:

```
int main(){
    int n = 0;
    puts("Digite um numero");
    scanf("%d", &n);
    printf("\n");
    int linha=0, coluna=0;
    int temp=0;
    while(linha<4)
    {
        while(coluna<4)
        {
            if(linha==coluna)
            {
                printf("%d ", n);
                coluna++;
            }
        }
        printf("\n");
        coluna=0;
        linha++;
    }
    return 0;
}
```

O resultado da execução deste código será:

- Será impresso o número digitado 16 vezes.
- Será impresso o número digitado 4 vezes, somente quando coluna e linha forem iguais.
- O código entra em looping na linha 10, pois a variável coluna nunca será incrementada, e portanto, nunca terá um valor igual a linha.
- Nada será impresso porque a variável linha sempre será maior que a variável coluna.
- Nada será impresso porque a variável coluna sempre será maior que a variável linha.

6.47) O código abaixo compila e executa sem erros de sintaxe ou compilação:

```
int main(){
    puts("Informe o numero: ");
    int num=0, i=0;
    scanf("%d", &num);
    while(i<10000)
    {
        num = num*i;
        i++;
        if(num>5){
            break;
        }
    }
    printf("Resultado: %d", num);
}
```

Qual será o resultado da execução deste código caso o usuário informe o número 5?

- "Resultado: 0" com a saída do laço de repetição através da linha 10 no comando "break".
- "Resultado: 0" com a saída do laço de repetição após 10 mil execuções".
- "Resultado: 75" com a saída do laço de repetição através da linha 10 no comando "break".
- Programa entra em looping pois nunca alcança a linha 10 no comando "break".
- "Resultado: 0" com a saída do laço de repetição através da linha 10 no comando "break".

6.48) O código abaixo compila e executa sem erros de sintaxe ou compilação, o resultado será:

```
int n1, n2;
puts("Digite o primeiro numero: ");
scanf("%d", &n1);
puts("Digite o segundo numero: ");
scanf("%d", &n2);
```

```

if(n1<n2){
    int temp=++n1+1;
    while(temp<n2){
        printf("%d ", temp);
        temp++;
    }
}

```

- Impressão em tela dos números do intervalo entre n1 e n2 de 1 em 1.
- Impressão em tela dos números do intervalo entre n1 e n2 de 2 em 2.
- Impressão em tela dos números do intervalo entre n1 e n2 de 3 em 3.
- O programa não irá imprimir nada na tela.
- Nenhuma das alternativas.

6.49) O código abaixo compila e executa sem erros de sintaxe ou compilação, em relação ao resultado da execução do programa assinale a alternativa correta:

```

float temp=0.0;
while(1){
    scanf("%f", &temp);
    printf("O numero informado: %.2f\n\n", temp);
    if(temp==10 && temp==15){
        printf("Saindo...");
        break;
    }
}

```

- Se informado 12 o programa é encerrado.
- Se informado 10 o programa é encerrado.
- Se informado entre 10 (inclusive) e 15 (inclusive) o programa é encerrado.
- Se informado entre 11 (inclusive) e 14 (inclusive) o programa é encerrado.
- Nenhuma das alternativas anteriores.

6.50) O código abaixo compila e executa sem erros de sintaxe ou compilação, em relação ao resultado da execução do programa assinale a alternativa correta (1 ponto):

```

float temp=0.0;
scanf("%f", &temp);
while(temp>=0){
    printf("%.2f\n\n", temp);
    if(temp>=10 && temp<=15){
        printf("Saindo...");
    }
    temp+=2;
}

```

- Se informado 12 o programa é encerrado sem mostrar a mensagem "Saindo...".
- Se informado 10 o programa entra em looping eterno.
- Se informado entre 10 e depois 15 o programa encerra sem mostrar a mensagem "Saindo...".
- Se informado entre 11 (inclusive) e 14 (inclusive) o programa é encerrado mostrando a mensagem "Saindo...".
- Nenhuma das alternativas anteriores.

6.51) Qual é a saída do programa abaixo?

```

int main(int argc, char *argv[]) {
    float result=0, a=1, b=2, c=3, d=7, e=8;
    while(a++>=1 && ++b<=2){
        result=(b-(c*7))/(4-(a*c*a));
    }
    printf("%f",result);
    return 0;
}

```

- a) -19.000000
- b) -48.000000
- c) -22.000000
- d) 0.000000
- e) 22.000000

6.52) Qual é a saída do programa abaixo?

```
int main(int argc, char *argv[]) {
    int i=1, i2=18;
    while(1){
        printf("%f", ((++i+i2++)/2.0));
        if(i==3){
            break;
        }
    }
    return 0;
}
```

- a) 12.000000
- b) 09.00000010.000000
- c) 10.00000011.000000
- d) 11.00000012.000000
- e) 0.000000

6.53) Leia o programa a seguir:

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

int main()
{
    int i, num, pos=0, neg=0;
    printf("Digite 5 numeros inteiros, positivos ou negativos\n");
    for(i=0; i<5; i++){
        scanf("%d", &num);
        if(num>=0)
            pos++;
        else
            neg++;
    }
    printf("Positivos : %d\nNegativos : %d", pos, neg);
    return 0;
}
```

Este programa irá imprimir:

- a) a) um número positivo e um negativo;
- b) b) todos os números que o usuário digitar, podendo serem positivos ou negativos;
- c) c) a quantidade de números positivos e negativos que o usuário digitou;
- d) d) o contrário que o usuário digitou, ou seja, os números positivos serão impressos negativos e os negativos serão impressos positivos;
- e) e) nenhuma das alternativas.

6.54) Analise o programa a seguir:

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

int main()
```

```

{
    int numero,num=1,i;
    printf("\nDigite um numero positivo: ");
    scanf("%d",&numero);
    for(i=numero; i>1; i--)
    {
        num=num*i;
    }
    printf("fat: %u\n", num);
    return 0;
}

```

O programa irá imprimir:

- a) a) a tabuada na forma crescente do número digitado pelo usuário;
- b) b) a tabuada na forma decrescente do número digitado pelo usuário;
- c) c) o resultado da multiplicação do número digitado pelo número antecedente a ele;
- d) d) o resultado do fatorial do número digitado;
- e) e) nenhuma das alternativas.

### 6.55) Qual a função do programa abaixo?

```

int main()
{
    int n, par, impar, num, cont;
    printf("Digite o tamanho da sequencia: ");
    scanf("%d", &n)
    par = 0;
    impar = 0;
    cont = 0;
    while (cont < n){
        printf("Digite o %do. numero: ",cont+1);
        scanf("%d", &num);
        if (num%2 == 0)
            par = par + 1;
        else
            impar = impar + 1;
        cont = cont + 1;
    }
    printf("Sequencia com %d numeros pares e %d impares.\n\n",
        par,impar);
    return 0;
}

```

- a) a) O programa cria um valor a partir de uma sequencia de números digitados pelo operador, mostra esse na tela e informa ao final se este será par ou impar.
- b) b) O programa cria um valor a partir de uma sequencia de números digitados pelo operador, mostra esse na tela e informa ao final quantos números pares e quantos números ímpares formam o valor final.
- c) c) O programa analisa a sequencia de números digitados pelo operador, e diz quantos números digitados são pares e quantos são ímpares.
- d) d) O programa analisa a sequencia de números digitados pelo operador, e diz quais números digitados são pares e quais são ímpares.
- e) e) Nda.

### 6.56) O que o programa irá imprimir?

```

int main()
{
    int i;
    printf ("Digite um valor: ");
    scanf ("%d", &i);
    do

```

```
{
    if (i<=100)
    {
        printf (" O valor de i = %d\n", i);
    }
    else
    {
        printf ("Valor maior que 100");
    }
    i++;
}
while (i<=100);
return 0;
}
```

- a) Todos os valores de 0 a 100
- b) Todos os valores menores do número digitado
- c) Todos os valores maiores que o numero digitado
- d) O valor digitado e os valores até 100 (inclusive)
- e) N.D.A.

## Capítulo 7: Funções

7.1) Escreva uma função "float media(float n1, float n2)" que dá como resultado a média dos valores.

7.2) Escreva uma função que receba dois números inteiros retorne o menor número.

7.3) Faça uma função que recebe um valor inteiro e verifica se o valor é par. A função deve retornar 1 se o número for par e 0 se for ímpar.

7.4) Crie uma função "float validaNota(float n){...}" para validar as notas de um sistema acadêmico. As notas devem ser números reais positivos de 0 a 10. Caso a nota esteja fora deste intervalo a função deve retornar -1 e caso a nota seja válida a função deve retornar a própria nota.

7.5) Crie uma função "float capturaNota(){...}" para capturar as notas de um sistema acadêmico. Esta função deverá possuir um scanf() dentro dela. As notas devem ser números reais positivos de 0 a 10. Caso a nota esteja fora deste intervalo a função deve informar ao usuário e pedir que ele informe novamente a nota. Quando o usuário informar uma nota válida a função retorna a própria nota.

7.6) Faça uma função que recebe um valor inteiro e verifica se o valor é positivo, negativo ou zero. A função deve retornar 1 para valores positivos, -1 para negativos e 0 para o valor 0.

7.7) Construa uma função float elevado(float x, int y), que calcula  $x^y$ . Não utilize a função pow. Preveja condições de erro (exemplo: y negativo).

7.8) Crie a função void imprime\_espacos (int n\_espacos) e a função void imprime\_asteriscos (int n\_asteriscos); elas devem imprimir na tela, sem pular linha, vários caracteres (espacos e asteriscos, respectivamente), de acordo com o número passado pelo parâmetro.

7.9) Escreva uma função que recebe como parâmetro um inteiro positivo ano e devolve 1 se ano for bissexto, 0 em caso contrário. (Um ano é bissexto se  $(ano \% 4 == 0 \ \&\& \ (ano \% 100 != 0 \ || \ ano \% 400 == 0))$ .)

7.10) Escreva um programa em C que leia 5 valores inteiros e imprima para cada um o seu correspondente valor absoluto. Para obter o valor absoluto do número utilize a função Absoluto especificada abaixo:

- a) Nome: Absoluto
- b) Descrição: Retorna o valor absoluto do número fornecido.
- c) Entrada: int n
- d) Saída: (int) O respectivo valor absoluto de n.

7.11) Escreva uma função que calcule e retorne a distância entre dois pontos (  $x_1, y_1$ ) e (  $x_2, y_2$ ). Todos os números e valores de retorno devem ser do tipo float.

7.12) Escreva uma função potencia(base, expoente) que, quando chamada, retorna  $\text{base}^{\text{expoente}}$ . Por exemplo, potencia(3,4) deve retornar 81. Crie uma outra função para utilizar dentro da função potencia para validar que o expoente é um inteiro maior ou igual a 1.

7.13) Escreva um programa que leia 5 números inteiros positivos (utilize uma função que leia esse número e verifique se ele é positivo). Para cada número informado escrever a soma de seus divisores (exceto ele mesmo). Utilize a função SomaDivisores para obter a soma.

- a) Nome: SomaDivisores
- b) Descrição: Calcula a soma dos divisores do número informado (exceto ele mesmo).
- c) Entrada: Um número inteiro e positivo.
- d) Saída: A soma dos divisores.
- e) Exemplo: Para o valor 8:  $1+2+4 = 7$

7.14) Escreva uma função que receba 3 notas de um aluno e uma letra. Se a letra for A a função retorna a média aritmética das notas do aluno, se for P, a sua média ponderada (pesos: 5, 3 e 2) e se for H, a sua média harmônica.

7.15) Faça um programa que use a função valorPagamento para determinar o valor a ser pago por uma prestação de uma conta. O programa deverá solicitar ao usuário o valor da prestação e o número de dias em atraso e passar estes valores para a função valorPagamento, que calculará o valor a ser pago e devolverá este valor ao programa que a chamou. O programa deverá então exibir o valor a ser pago na tela. Após a execução o programa deverá voltar a pedir outro valor de prestação e assim continuar até que seja informado um valor igual a zero para a prestação. Neste momento o programa deverá ser encerrado, exibindo o relatório do dia, que conterá a quantidade e o valor total de prestações pagas no dia. O cálculo do valor a ser pago é feito da seguinte forma. Para pagamentos sem atraso, cobrar o valor da prestação. Quando houver atraso, cobrar 3% de multa, mais 0,1% de juros por dia de atraso.

## Capítulo 8: Vetores e Matrizes

**Abaixo temos exercícios envolvendo vetores, ou seja, matrizes de apenas uma dimensão.**

8.1) Faça um programa que cria um vetor com 3 elementos inteiros, lê os 3 números do teclado, armazena os números em um vetor e imprime o vetor na saída padrão.

8.2) Faça um programa que cria um vetor com 5 elementos inteiros, lê 5 números do teclado, armazena os números no vetor e imprime o vetor na ordem inversa.

8.3) Escreva um programa que lê 5 números inteiros. Após a captura dos 5 números mostrar os todos os 5 números na tela um ao lado do outro separados por espaços. O programa deve imprimir na linha abaixo o maior dos números com a seguinte frase: "O maior dos números acima é: <numero>".

8.4) Escreva um programa que lê 10 números reais. O programa deve imprimir a média, o maior e o menor dos números.

8.5) Escreva um programa para ler as notas de 5 avaliações de um aluno (utilize vetores para armazenar), depois deve-se calcular e imprimir a média semestral. Crie uma função para validar as notas. Faça com que o algoritmo só aceite notas válidas (uma nota válida deve pertencer ao intervalo [0,10]). Cada nota deve ser validada separadamente. Deve ser impressa a mensagem "Nota inválida" caso a nota informada não pertença ao intervalo [0,10].

8.6) Reescreva o programa anterior para que no final seja impressa a mensagem Novo cálculo (1.sim 2.não) solicitando ao usuário que informe um código (1 ou 2) indicando se ele deseja ou não executar o programa novamente. Se for informado o código 1 deve ser repetida a execução de todo o programa para permitir um novo cálculo, caso contrário ele deve ser encerrado. Valide a resposta do usuário para a pergunta Novo Cálculo (1.sim 2.não)? (aceitar apenas o código 1 ou 2).

8.7) Faça um programa que leia um número indeterminado de idades (crie um vetor com 10 mil elementos). A última idade lida, que não entrará nos cálculos, deverá ser igual a zero, ou seja, se a idade informada for zero o programa finaliza e exibe os dados de quantas idades foram lidas. Deve-se também calcular e escrever a média de idade desse grupo de idades.

8.8) Faça um programa para ler a altura e o sexo (feminino ou masculino) de 10 pessoas. Ao final calcular e escrever na saída padrão:

- a) a maior e a menor altura
- b) a média de altura das mulheres
- c) o número de homens

8.9) Faça um programa que toma as notas dos alunos (15 números fracionários) do usuário e guarda em um vetor. Depois disso, calcula e imprime na tela a média dos alunos, a nota mais alta, a nota mais baixa e o número de notas 7.0.

8.10) Faça um programa que ordene um vetor de dez números em ordem crescente. Ex.: inicialmente ele vale {2,78,5,-1,0,3,3,56,7,77} e é impresso na tela na sequência -1, 0, 2, 3, 3, 4, 5, 7, 56, 77. Sugestão: ache o menor valor do vetor e coloque na primeira posição; daí ache o segundo menor (basta procurar da segunda posição em diante) e coloque na segunda posição, e assim por diante. Este método é conhecido como inserção direta. Há vários outros (pode pesquisar).

**Abaixo temos exercícios envolvendo matrizes com duas dimensões.**

8.11) Faça um programa que lê uma matriz de 3 x 3 elementos usando um comando for, multiplica cada elemento por 5 e imprime o resultado.

8.12) Faça um programa que lê um vetor de 3 elementos e uma matriz de 3 x 3 elementos. Em seguida o programa deve fazer a multiplicação do vetor pelas colunas da matriz.

8.13) Faça um programa que apresente automaticamente uma matriz de dimensões 4x5, sendo que o valor de  $A(i, j) = 3i + j^2$ .

8.14) Faça um programa que receba duas matrizes de dimensões 3x3 e apresente a soma de seus elementos.

A adição de duas matrizes se dá pela soma de seus termos correspondentes.

Esquemáticamente, se  $A + B = C$ , então  $c_{ij} = a_{ij} + b_{ij}$

Assim, sendo a soma de duas matrizes só está definida quando elas possuem a mesma ordem (número de linhas e colunas):

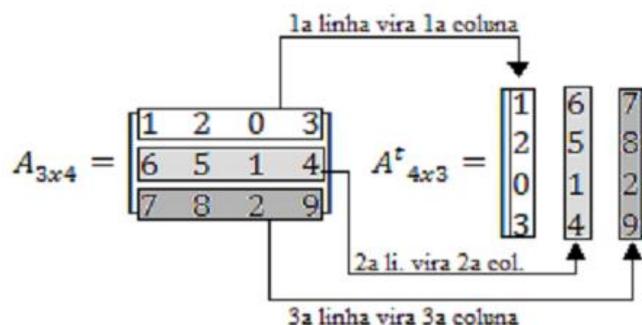
$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} + \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$$

Onde:

$$\begin{cases} c_{11} = a_{11} + b_{11} \\ c_{12} = a_{12} + b_{12} \\ c_{21} = a_{21} + b_{21} \\ c_{22} = a_{22} + b_{22} \end{cases}$$

8.15) A matriz transposta de uma matriz A qualquer é denominada por  $A^t$ . Esta matriz nada mais é do que a matriz A com suas linhas transformadas em colunas e suas colunas transformadas em linha.

Observe:



Faça um programa que receba as dimensões da matriz, depois os valores e apresenta a matriz transposta.

8.16) Através de laços de repetição e decisão imprima uma matriz  $B = (b_{ij})_{5 \times 5}$  tal que:

$$b_{ij} \begin{cases} -2 & \text{se } i > j \\ 1 & \text{se } i = j \\ 2 & \text{se } i < j \end{cases}$$

**Abaixo temos exercícios envolvendo *strings* que são vetores de caracteres ou ainda matrizes de caracteres que podem armazenar várias *strings*.**

8.17) Faça um programa que lê duas palavras do teclado e diz se elas são iguais ou diferentes. O programa deve conferir e exibir um relatório informando se alguma das palavras digitadas é igual a “Brasil” ou “Santa Catarina”.

8.18) Faça um programa que lê uma frase do teclado e indica na tela quantas vezes a letra “s” aparece na frase.

8.19) Faça um programa que lê três palavras do teclado e imprime as três palavras na ordem inversa.

8.20) Faça um programa que criptografa uma mensagem informada pelo usuário (via `scanf()` ou `gets()`) e exibe a mensagem criptografada na tela. A criptografia deverá ser realizada alterando os caracteres através da soma de 1 inteiro referente ao código da tabela ACSII ou seja, a letra ‘a’ na mensagem será exibida como ‘b’, a letra ‘d’ como ‘e’ e assim por diante.

8.21) Escreva um programa em linguagem C para capturar o nome e idade de várias pessoas. Ao final do primeiro cadastramento de nome e idade o programa deve solicitar ao usuário que seja informado ‘c’ para continuar ou ‘s’ para sair. Ao final do cadastramento deve ser exibida uma lista com os nomes ao lado das idades e na última linha a quantidade de nomes e a média das idades informadas.

8.22) Dada a matriz de duas dimensões de caracteres abaixo que foi definida pelo nome de variável `cincoPalavras`

	0	1	2	3	4	5	6
0	P	E	D	R	O		
1	M	A	R	I	A		
2	J	E	S	U	S		
3	I	F	C				
4	L	U	Z	E	R	N	A

O que será impresso em tela quando da execução da linha abaixo?

```
printf("%c", cincoPalavras[2][2]);
```

### 8.23) Qual a função do programa a seguir?

```
int main()
{
    int z;
    float vetor_cubo[5], vetor[5];
    for(z=0; z<5; z++)
    {
        printf("Digite um numero: ");
        scanf("%f",&vetor[z]);
        printf("\n");
        vetor_cubo[z]= pow (vetor[z],3);
    }
    for(z=0; z<5; z++)
        printf("%3.f\t", vetor_cubo[z]);
    printf("\n");
    system("pause");
    return(0);
}
```

- Ele preenche um vetor com 5 números e mostra outro vetor com os números em ordem crescente;
- Ele preenche um vetor com 5 números e mostra outro vetor com o cubo dos números do primeiro vetor;
- Ele preenche um vetor com 5 números e mostra outro vetor com o quadrado dos números do primeiro vetor;
- Ele preenche um vetor com 5 números e mostra outro vetor com o cubo dos números do primeiro vetor e a soma deles;
- Nda.

### 8.24) Um aluno escreveu o seguinte programa. Marque a resposta que corresponde ao que o programa imprime.

```
#include <stdio.h>
int main (void)
{
    int v [10] = {-1, -2, -3, -4, -5, -6, -7, -8, -9, -10 };
    int i, maior , s;
    maior = s = 0;
    for (i = 0; i < 10; i++) {
        s += v[i];
        if (v[i] > maior ) {
            maior = v[i];
        }
    }
    printf ("%d %d\n", maior , s);
    return 0;
}
```

- 1 -55
- 0 -55
- 10 -55
- 55 -10
- 55 0

## Capítulo 9: Arquivos em disco

9.1) Faça um programa que grava uma palavra em um arquivo chamado “palavra.txt”, a palavra deve ser digitada pelo operador.

9.2) Faça um programa que lê a palavra do arquivo do exercício anterior e a apresenta na tela.

9.3) Faça um programa que grava uma matriz de 3 x 3 números inteiros em um arquivo chamado “mat.txt”, os números devem ser digitados pelo operador.

9.4) Faça um programa que lê a matriz do arquivo do exercício anterior e a apresenta na tela.

9.5) Faça um programa utilizando o comando “for” que lê 10 números do teclado e armazene estes 10 números em um arquivo chamado “num.txt” na área de trabalho do computadores.

9.6) Faça um programa que lê 20 palavras de um arquivo e verifica se a palavra “casa” esta entre elas.

9.7) Faça um programa que lê 5 notas de um arquivo chamado “notas.txt”, calcula a média destas notas, imprime a média destas notas e salva a média no final deste mesmo arquivo.

9.8) Faça um programa utilizando o comando “switch-case” que pede para o operado escolher entre três opções: (S) para sair, (L) para ler a matriz e (G) para gravar a matriz. Se o operador escolher a opção S o programa é finalizado. Se o operador escolher a opção L o programa deve pedir ao operador qual o nome do arquivo e em seguida ler a matriz deste arquivo e apresentar a matriz na tela. E se o operador escolher a opção G o programa deve pedir para o operador digitar o nome do arquivo, o número de linhas e o número de colunas da matriz, e em seguida deve pedir para o operador digitar os números que compõem a matriz. A matriz deve ser salva no arquivo especificado. Não esqueça de incluir no arquivo o número de linhas e colunas da matriz para que a leitura dos dados seja possível.

9.9) Escreva um programa que leia todos os caracteres do arquivo “texto.txt” e que imprima por ordem decrescente as percentagens das ocorrências das letras encontradas; os caracteres que não são letras são ignorados.

### **Exemplo**

```
$ ./programa.exe
```

```
f[a] = 11.4%
```

```
f[i] = 11.1%
```

```
f[e] = 9.7%
```

```
f[t] = 9.3%
```

```
f[r] = 9.0%
```

```
f[f] = 8.0%
```

```
f[l] = 7.6%
```

$f[n] = 6.9\%$

...  
 $f[b] = 0.0\%$

Sugere-se o seguinte método para resolver este problema. Usar um vector  $f[]$  para contar o número de ocorrências de cada letra e um vector  $let[]$  com as letras; por exemplo, no início  $f[0]=\dots=f[25]=0$ ,  $let[0]='a', \dots, let[25]='z'$

- Para cada caracter lido  $c$ , se for uma letra:
- Se  $c$  for maiúscula converte-se para minúscula
- Faz-se:  $f[c-'a']++$
- Ordenam-se por ordem decrescente de frequência os pares  $(f[i], let[i])$
- Escrevem-se esses pares para  $i=0, \dots, 25$  usando um formato apropriado.

9.10) Controle de cotas de disco. A ACME Inc., uma organização com mais de 1500 funcionários, está tendo problemas de espaço em disco no seu servidor de arquivos. Para tentar resolver este problema, o Administrador de Rede precisa saber qual o espaço em disco ocupado pelas contas dos usuários, e identificar os usuários com maior espaço ocupado. Através de um aplicativo baixado da Internet, ele conseguiu gerar o seguinte arquivo, chamado “usuarios.txt”:

```
alexandre      456123789
anderson       1245698456
antonio        123456456
carlos         91257581
cesar          987458
rosemary       789456125
```

Neste arquivo, o primeiro campo corresponde ao login do usuário e o segundo ao espaço em disco ocupado pelo seu diretório home. A partir deste arquivo, você deve criar um programa que gere um relatório, chamado “relatório.txt”, no seguinte formato:

```
ACME Inc.                Uso do espaço em disco pelos usuários
-----
Nr.  Usuário                Espaço utilizado      % do uso

1    alexandre              434,99 MB             16,85%
2    anderson               1187,99 MB            46,02%
3    antonio                117,73 MB             4,56%
4    carlos                 87,03 MB              3,37%
5    cesar                  0,94 MB               0,04%
6    rosemary              752,88 MB            29,16%

Espaço total ocupado: 2581,57 MB
Espaço médio ocupado: 430,26 MB
```

O arquivo de entrada deve ser lido uma única vez, e os dados armazenados em memória, caso sejam necessários, de forma a agilizar a execução do programa. A conversão da espaço ocupado em disco, de bytes para megabytes deverá ser feita através de uma função separada, que será chamada pelo programa principal. O cálculo do percentual de uso também deverá ser feito através de uma função, que será chamada pelo programa principal.

9.11) Faça um programa para ler os dados de um arquivo no seguinte formato (layout):

```
3
ZE
8.5
10.0
ANTONIO
7.5
8.5
SEBASTIAO
5.0
6.0
```

Nesta entrada, a primeira linha contém o número de alunos que serão inseridos. As linhas seguintes contêm os seguintes dados:

- nome do aluno com no máximo 50 caracteres;
- nota da primeira prova;
- nota da segunda prova

9.12) Escreva um programa que leia um arquivo no layout acima e imprima os nomes de todos os alunos que têm a média das duas notas menor que 7.0.

## Capítulo 10: Operadores de bits

10.1) Escreva uma função que receba um valor em decimal, converta para binário e dentro da própria função exiba na tela o resultado. A função terá tipo de retorno *void*.

10.2) Faça uma função chamada “int bin2dec(int b1, int b2, int b3, int b4)” que receba um número binário de 4 bits e converta para decimal. Cada bit deverá ser um argumento inteiro da função. Exemplo de utilização: “int decimal = bin2dec(1,0,1,1);”

10.3) Faça a função qtdeDeBitsParesLigados(int n) que retorna o número de bits pares que estão ligados.

```
Ex:
main() {
    qtdeDeBitsParesLigados(73);
    //se 73 = 0100 1001, então qtdeDeBitsParesLigados retorna 2
}
```

10.4) Escreva uma função criptografa(int n) que recebe um inteiro n com 8 bits (índices: 7,6,5,4,3,2,1,0) e que retorna esse inteiro embaralhando esses bits para a seguinte sequência (7,5,3,1,6,4,2,0)

```
Ex:
main() {
    criptografa(73); // se 73 = 0100 1001
    //então criptografa(73) == 0010 1001 == 41
}
```

10.5) Faça uma função pisca() que ora liga todos os bits pares e ora liga todos os bits ímpares num intervalo de 2 segundo entre as mudanças, imitando o efeito de um pisca-pisca de natal. Apresente os valores em binários de como ficaria a saída. A palavra de trabalho deverá ter 8 bits. O valor de milissegundos é usado para fazer uma espera entre a alternância, e para isso use a função Sleep(<int milissegundos>) da biblioteca windows.h. A saída terá as linhas abaixo sendo impressas uma a uma a cada 2 segundos:

```
Saída:
01010101
10101010
01010101
10101010
01010101
10101010
```

10.6) Faça uma função piscaUmIndoEVoltando(int milissegundos) que faz os leds pares serem ligados sequencialmente, um de cada vez e de forma crescente, e em seguida, os leds ímpares serem ligados de forma decrescente. Apresente na tela o resultado em formato binário. A palavra de trabalho deverá ter 8 bits. Cada linha será impressa apenas depois de x milissegundos (parâmetro da função). A sequência gerada deverá ser:

```
Saída:
01000000
01010000
01010100
```

```
01010101
01010111
01011111
01111111
11111111
```

10.7) Faça um programa que leia um byte do teclado e a seguir zere os bits 3 e 4, e inverta os bits 0 e 7. O resultado deverá ser mostrado em hexadecimal na tela.

10.8) Construa funções que realizem as seguintes operações aplicadas aos valores 0b10101, 0b11111, 0b11100:

- a) Mostre o valor do primeiro bit da direita
- b) Mostre o terceiro bit da direita para a esquerda
- c) Mostre o primeiro bit da esquerda
- d) Mostre o terceiro bit da esquerda para a direita
- e) Mostre os dois primeiros bits da esquerda para a direita
- f) Mostre os bits pares
- g) Altere o primeiro bit da direita para 0
- h) Altere o primeiro bit da esquerda para 0
- i) Inverta todos os bits (o que for 0 deverá ser 1 e o que for 1 deverá ser 0)
- j) Mostre apenas 4 bits da direita, removendo os outros

10.9) Faça uma função consecutivos(int x) que retorne 1, caso o inteiro c possua 2 ou mais bits consecutivos ligados.

10.10) Crie a função rotacionaParaDireita(int x, int qtde) que retorna x com n rotações à direita sem perder os bits.

10.11) Faça a função cripto(int x, int rotacoes) que recebe um inteiro x com 8 bits com os bits tendo índices base zero (7,6,5,4,3,2,1,0) e que retorna esse inteiro rotacionado para a esquerda, sem perder bits, com o número de rotações do parâmetro "rotações". Faça posteriormente a função descripto(int x, int rotações) que faz a rotação inversa e recupera as informações.