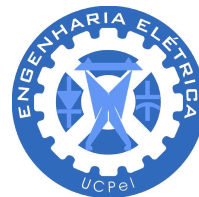


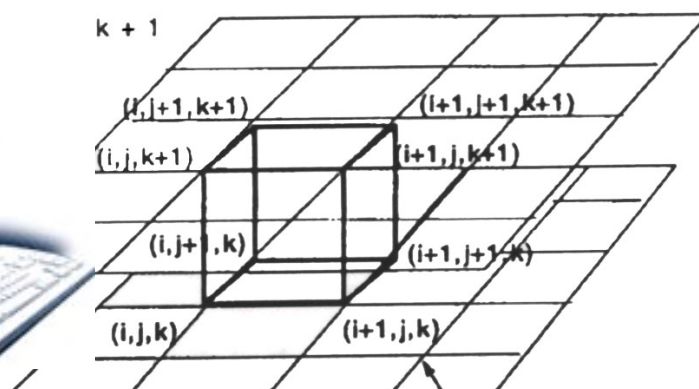


Universidade Católica de Pelotas
Engenharia Elétrica/Eletrônica



Engenharia Computacional I

(caderno de exercícios)



ÍNDICE

A lista de exercícios apresenta problemas distribuídos da seguinte forma:

Do problema 1 até o 11:	Tipos, precedência, operadores aritméticos	19
Do problema 12 até o 32:	Algoritmos seqüenciais	22
Do problema 33 até o 79:	Algoritmos de seleção	33
Do problema 80 até o 139:	Algoritmos de repetição	66
Do problema 140 até o 161:	Vetores	103
Do problema 162 até o 168:	Matrizes	120
Exercícios preparatórios para a primeira prova		127
Exercícios preparatórios para a segunda prova		135

O material que será apresentado foi escrito originalmente para a Disciplina de Algoritmos e Programação do Curso de Ciência da Computação da UCPel. O material foi escrito pelos professor Ricardo Andrade Cava em 1999. A íntegra do material pode ser obtido em

<http://atlas.ucpel.tche.br/%7Ecava/lp/labprog1/LPIExercicios.html>

Caderno de exercícios Algoritmos e Programação

Lista de exercícios de 001 a 168

1. Escreva um algoritmo para preparar uma limonada.
2. Um homem precisa atravessar um rio com um barco que possui capacidade de carregar apenas ele mesmo e mais uma de suas três cargas, que são: um lobo, um bode e um maço de alfafa. O que o homem deve fazer para conseguir atravessar o rio sem perder suas cargas?
3. Escreva os passos necessários para obter o resultado da expressão $34 \times 3 + 5 \times 15$.
4. Escreva um algoritmo para calcular o comprimento de uma circunferência de raio 3 cm.
5. Escreva um algoritmo que armazene o valor 10 em uma variável **A** e o valor 20 em uma variável **B**. A seguir (utilizando apenas atribuições entre variáveis) troque os seus conteúdos fazendo com que o valor que está em **A** passe para **B** e vice-versa. Ao final escrever os valores que ficaram armazenados nas variáveis.
6. Analise os seguintes algoritmos e diga o que será impresso ao serem executados.

a)

A ← 10
B ← 20
Escreva B
B ← 5
Escreva A,B

b)

A ← 30
B ← 20
C ← A + B
Escreva C
B ← 10
Escreva B,C
C ← A + B
Escreva A,B,C

c)

A ← 10
B ← 20
C ← A
B ← C
A ← B
Escreva A,B,C

d)

A ← 10
B ← A + 1
A ← B + 1
B ← A + 1
Escreva A
A ← B + 1
Escreva A

e)

A ← 10
C ← A + B
B ← 20
Escreva C

7. Escreva um algoritmo que calcule e imprima a média semestral de um aluno que tenha obtido as seguintes notas: 8,0 na 1ª avaliação e 6,0 na 2ª avaliação. OBS: Escrever nas 3 representações.

8. Escreva o valor que será atribuído a cada uma das variáveis.

a = 3+4*5	d = 5*(3+(2+3))/2+1	g = 21 div 4-2	j = 21 div 4 div 2
b = 8/4+2*3	e = 1+12/((7+2)/3)+(6-2)	h = 11 mod 4+8 div 3	
c = 2*(10-3*3)-1	f = 3+15/2+5	i = sqrt(9)+sqr(9)	

9. Os pares de instruções abaixo produzem o mesmo resultado?

a = (4/2)+(2/4)	e	a = 4/2+2/4
b = 4/(2+2)/4	e	b = 4/2+2/4
c = (4+2)*2-4	e	c = 4+2*2-4

10. Reescreva as instruções a seguir com o mínimo de parênteses possível sem alterar o resultado.

a = 6*(3+2)	d = 2*(8/(3+1))	g = ((3+(8/2))*4)+(3*2)	j = ((-12)*(-4))+(3*(-4))
b = 2+(6*(3+2))	e = 3+(13-2)/(2*(9-2))	h = (6*(3*3)+6)-10	
c = 2+(3*6)/(2+4)	f = (6/3)+(8/2)	i = (((10*8)+3)*9)	

11. Escreva as seguintes expressões matemáticas em C:

$$\begin{aligned}
 a &\leftarrow \frac{B + C}{D + E} & b &\leftarrow \frac{A}{C^2 + D} & c &\leftarrow \frac{A^2}{B - 1} \\
 d &\leftarrow \frac{A}{B + E} - \frac{E}{F \cdot G} & e &\leftarrow \frac{-B + \sqrt{B^2 - 4ac}}{2a} & f &\leftarrow \frac{A - B}{C + D} + \frac{D + E}{A - B} \\
 g &\leftarrow r^3 \cdot i + y & h &\leftarrow \frac{A + B}{C^2} + \frac{D - G}{E^2 + F^2} & i &\leftarrow \sin\left(\frac{\pi}{2}\right) \\
 j &\leftarrow \sin^2\left(3\frac{\pi}{2}\right) & k &\leftarrow \frac{E}{\sqrt{R^2 + \left[2\pi \cdot F \cdot L - \frac{1}{2\pi \cdot F \cdot C}\right]^2}}
 \end{aligned}$$

12. Escreva um algoritmo para ler o raio de um círculo, calcular e escrever a sua área.

$$\text{Área} = \pi \cdot R^2$$

13. Escreva um algoritmo para ler um valor inteiro e escrever o seu antecessor.

14. Escreva um algoritmo para ler as coordenadas cartesianas de dois pontos x_1, y_1 e x_2, y_2 , calcular e escrever a distância entre esses pontos, segundo a fórmula da geometria analítica.

$$D = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2}$$

15. Escreva um algoritmo para ler uma temperatura em graus Fahrenheit, calcular e escrever o valor correspondente em graus Celsius.

$$\frac{C}{5} = \frac{F - 32}{9}$$

16. Escreva um algoritmo para ler uma temperatura em graus Celsius, calcular e escrever o valor correspondente em graus Fahrenheit.

17. Escreva um algoritmo para ler a medida de um arco em graus, calcular e escrever a medida correspondente em radianos.

$$\pi \text{ radianos} = 180^\circ$$

18. Escreva um algoritmo para ler as medidas dos lados de um retângulo, calcular e escrever sua área e a medida da diagonal.

19. Escreva um algoritmo para ler as dimensões de uma cozinha retangular (comprimento, largura e altura), calcular e escrever a “metragem” (em metros quadrados) de azulejos para se colocar em todas as suas paredes (desconsidere portas e janelas).

20. Escreva um algoritmo para ler o número de eleitores de um município, o número de votos brancos, nulos e válidos. Calcular e escrever o percentual que cada um representa em relação ao total de eleitores.

21. Escreva um algoritmo para ler o salário mensal e o percentual de reajuste. Calcular e escrever o valor do novo salário.

22.O custo ao consumidor, de um carro novo, é a soma do custo de fábrica com a percentagem do distribuidor e dos impostos (aplicados ao custo de fábrica). Supondo que a percentagem do distribuidor seja de 28 % e os impostos de 45%, escrever um algoritmo para ler o custo de fábrica de um carro e escrever o custo ao consumidor.

23.Uma revendedora de carros usados paga a seus funcionários vendedores, um salário fixo por mês, mais uma comissão também fixa para cada carro vendido e mais 5 % do valor das vendas por ele efetuadas. Escrever um algoritmo que leia o número de carros por ele vendidos, o valor total de suas vendas, o salário fixo e o valor que recebe por carro vendido. Calcule e escreva o salário mensal do vendedor.

24.A empresa Vestebem resolveu fazer uma promoção especial e conceder um desconto de 30 % sobre o preço de venda de todo seu estoque. Escreva um algoritmo que leia o preço de venda antes da promoção e calcule quanto deve ser o preço promocional.

25.Um empregado deseja saber se o cálculo de seu salário está correto verificando o seu contra-cheque. Escreva um algoritmo que leia o valor do salário bruto, o valor descontado para o INSS, o valor descontado para o imposto de renda, calcule e escreva o percentual que foi utilizado para o cálculo do INSS e IR.

26.A turma C é composta de 60 alunos, e a turma D de 20 alunos. Escreva um algoritmo que leia o percentual de alunos reprovados na turma C, o percentual de aprovados na turma D, calcule e escreva:

- a) O número de alunos reprovados na turma C.
- b) O número de alunos reprovados na turma D.
- c) A percentagem de alunos reprovados em relação ao total de alunos das duas turmas.

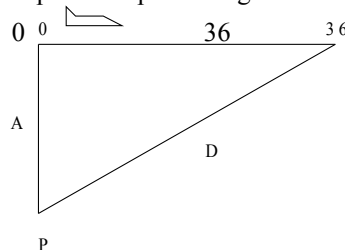
27.Escreva um algoritmo que calcule e escreva a quantidade de latas de tintas necessárias e o custo para pintar tanques cilíndricos de combustível, onde são fornecidos a altura e o raio desse cilindro, sabendo-se que:

-A lata de tinta custa R\$ 5,00.

-Cada lata contém 5 litros.

-Cada litro de tinta pinta 3 metros quadrados.

28.Um avião voando em linha reta a uma altitude A, passa sobre um ponto P num instante T=0. Se a sua velocidade é V, calcular a distância D do avião ao ponto P no instante T=36. Escrever um algoritmo para ler V (Km/H) e A(Km), calcular e escrever a distância (Km) ao ponto P após 36 segundos.



29.Um motorista de taxi deseja calcular o rendimento de seu carro na praça. Sabendo-se que o preço do combustível é de R\$ 0,53 , escreva um algoritmo para ler: a marcação do odômetro (Km) no início do dia, a marcação (Km) no final do dia, o número de litros de combustível gasto e o valor total (R\$) recebido dos passageiros. Calcular e escrever: a média do consumo em Km/L, o custo médio do Km em R\$/Km, e o lucro (líquido) do dia.

30.Uma loja vende bicicletas com um acréscimo de 50 % sobre o seu preço de custo. Ela paga a cada vendedor 2 salários mínimos mensais, mais uma comissão de 15 % sobre o preço de custo de cada bicicleta vendida, dividida igualmente entre eles. Escreva um algoritmo que leia o número de empregados da loja, o valor do salário mínimo, o preço de custo de cada bicicleta, o número de bicicletas vendidas, calcule e escreva: O salário final de cada empregado e o lucro (líquido) da loja.

31. Um mercado vende a dúzia de laranjas pelo dobro do preço de custo. Com a baixa em suas vendas o proprietário resolveu conceder um desconto a seus clientes. Escreva um algoritmo que leia o preço de custo da dúzia de laranjas (em R\$), o percentual do desconto (por dúzia) fornecido e o número de laranjas (unidade) adquiridas pelo cliente. Calcule e imprima o valor a ser pago pelo cliente em CR\$. Após a escrita do resultado acima, deverá ser lido o valor que o cliente pagou ao mercado (em CR\$) e escrever o troco que deverá ser fornecido em R\$.

$$\text{R\$ } 1,00 = \text{CR\$ } 2750,00$$

32. 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 algoritmo 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). Calcule e escreva 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.

33. Escreva um algoritmo para ler as notas das duas avaliações de um aluno no semestre, calcular e escrever a média semestral e a seguinte mensagem: 'PARABÉNS! Você foi aprovado' somente se o aluno foi aprovado.

34. Acrescente ao exercício acima a mensagem 'Você foi REPROVADO! Estude mais' caso a média calculada seja menor que 6,0.

35. Escreva um algoritmo para ler um valor e escrever a mensagem 'É maior que 10' se o valor lido for maior que 10, caso contrário escrever 'NÃO é maior que 10'.

36. Escreva um algoritmo para ler um valor e escrever se é positivo ou negativo. Considere o valor zero como positivo.

37. Escreva um algoritmo que verifique a validade de uma senha fornecida pelo usuário. A senha válida é a palavra UCPEL. Deve ser impresso as seguintes mensagens:

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

38. Escreva um algoritmo 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).

39. Escreva um algoritmo para ler 2 valores (considere que não serão lidos valores iguais) e escrever o maior deles.

40. Tendo como entrada a altura e o sexo de uma pessoa, construa um algoritmo que calcule e imprima seu peso ideal, utilizando as seguintes fórmulas:

- para homens: $(72.7 * h) - 58$
- para mulheres: $(62.1 * h) - 44.7$

41. As maçãs custam R\$ 0,30 se forem compradas menos do que uma dúzia, e R\$ 0,25 se forem compradas pelo menos doze. Escreva um algoritmo que leia o número de maçãs compradas, calcule e escreva o valor total da compra.

42. Escreva um algoritmo para ler um número inteiro (considere que serão lidos apenas valores positivos e inteiros) e escrever se é par ou ímpar.

43. Escreva um algoritmo para ler 2 valores (considere que não serão lidos valores iguais) e escreva-os em ordem crescente.

44. Escreva um algoritmo para ler um valor e escrever se é positivo, negativo ou zero.

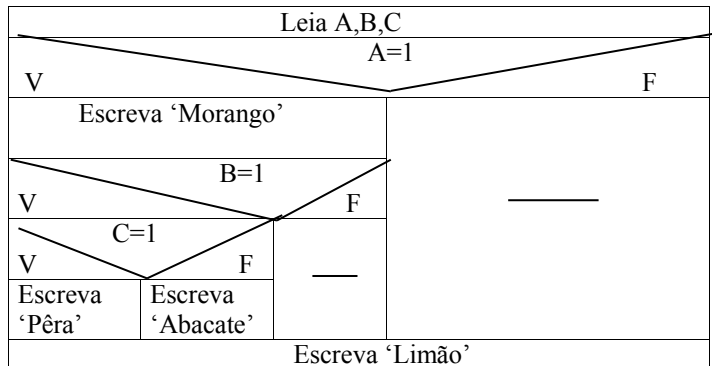
52. Analise o seguinte algoritmo e diga o que será impresso ao ser executado. Considere que para cada execução serão informados os seguintes valores para A,B,C:

a) A=1,B=1,C=1

b) A=1,B=1,C=0

c) Para ser impresso apenas a palavra LIMÃO, que valores devem ser informados para A,B,C.

d) Transcreva o diagrama de Chapin para o C sem acrescentar ou retirar comandos.



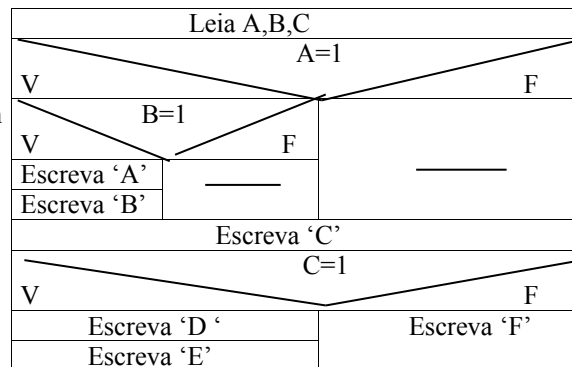
53. Analise o seguinte algoritmo e diga o que será impresso ao ser executado. Considere que para cada execução serão informados os seguintes valores para A,B,C:

a) A=1,B=1,C=1

b) A=1,B=1,C=2

c) A=2,B=1,C=2

d) Transcreva o diagrama de Chapin para o C sem acrescentar ou retirar comandos.



54. Analise o seguinte algoritmo e diga o que será impresso ao ser executado. Considere que para cada execução serão informados os seguintes valores para A,B,C,D,E:

a) A=1,B=1,C=1,D=1,E=1

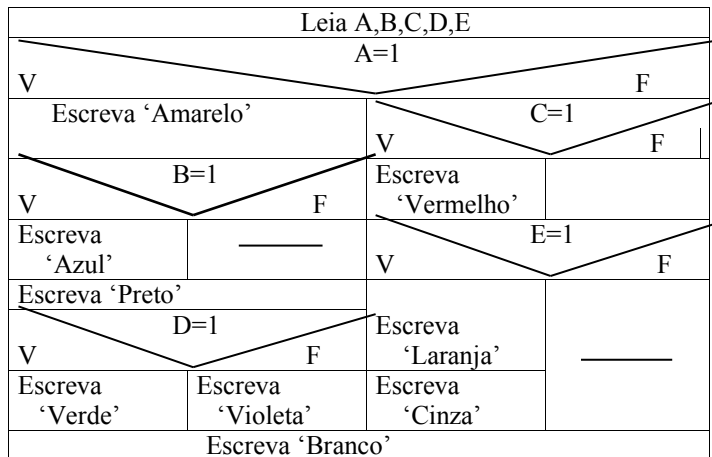
b) A=1,B=0,C=1,D=0,E=1

c) A=0,B=0,C=1,D=0,E=1

d) A=0,B=1,C=0,D=1,E=1

e) A=0,B=1,C=0,D=0,E=0

f) Transcreva o diagrama de Chapin para o C sem acrescentar ou retirar comandos.



55. Escreva um algoritmo para ler 3 valores (considere que não serão informados valores iguais) e escrever o maior deles.

56. Escreva um algoritmo para ler 3 valores (considere que não serão informados valores iguais) e escrever a soma dos 2 maiores.

57. Escreva um algoritmo para ler 3 valores (considere que não serão informados valores iguais) e escrevê-los em ordem crescente.

58. Escreva um algoritmo para ler 3 valores A, B e C representando as medidas dos lados de um triângulo, e escrever se formam ou não um triângulo.

OBS: Para formar um triângulo o valor de cada lado deve ser menor que a soma dos outros 2.

59. Escreva um algoritmo para ler as coordenadas (X,Y) de um ponto no sistema cartesiano e escrever o quadrante ao qual o ponto pertence.

OBS: Considere que nenhuma coordenada informada é igual a zero.

60. Escreva um algoritmo para ler as coordenadas (X,Y) de um ponto no sistema cartesiano e escrever o quadrante ao qual o ponto pertence. Se o ponto estiver sobre os eixos, ou na origem, escrever NÃO ESTÁ EM NENHUM QUADRANTE.

61. Para $A=5$, $B=3$ e $C=-2$ responda se as seguintes expressões lógicas são V ou F.

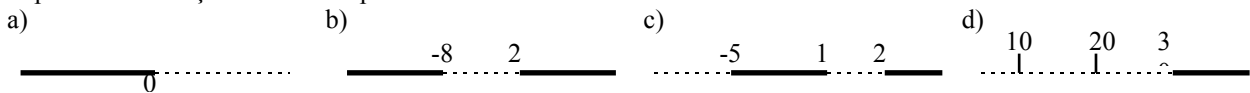
- | | | |
|------------------------|------------------------------------|--|
| a) $A > 3$ e $B > 5$ | d) não ($A=5$) | g) $C=B-2$ ou não ($C \neq B$ e $A < B$) |
| b) $A > 3$ ou $B > 5$ | e) $A > B+C$ e $C > -1$ e $B < 3$ | |
| c) $A > 3$ xou $B > 5$ | f) $C < 1$ ou $C \neq B$ e $A < B$ | |

62. Escreva um algoritmo para ler um valor e imprimir qual o tipo de faixa ele se encontra (cheia ou tracejada).

-Considere os limites dentro da linha cheia.

-Utilize apenas UM comando *se então senão* para cada exercício.

-Apresente 2 soluções diferentes para cada exercício.



63. Reescreva o exercício 55 utilizando operadores lógicos.

64. Reescreva o exercício 56 utilizando operadores lógicos.

65. Reescreva o exercício 57 utilizando operadores lógicos.

66. Reescreva o exercício 58 utilizando operadores lógicos.

67. Reescreva o exercício 60 verificando também se o ponto está sobre o eixo X, eixo Y ou na origem.

68. Escreva um algoritmo para ler a hora de início e fim de um jogo (considere apenas horas inteiras) e calcule a duração do jogo em horas, sabendo-se que o tempo máximo de duração do jogo é de 24 horas e que o jogo pode iniciar em um dia e terminar no dia seguinte.

69. Reescreva o exercício acima considerando que o tempo inicial e final do jogo será informado em horas e minutos. A duração do jogo também deverá ser impressa em horas e minutos.

70. A jornada de trabalho semanal é de 40 horas. O funcionário que trabalhar mais de 40 horas receberá hora extra, cujo valor é o valor da hora regular com um acréscimo de 50 %. Escreva um algoritmo que leia o número de horas trabalhadas em um mês, o salário por hora e escreva o salário total do funcionário (considere que o mês possua 4 semanas exatas).

71. Escreva um algoritmo para ler as 3 notas obtidas por um aluno nas 3 verificações e a média dos exercícios que fazem parte da avaliação. Calcular a média de aproveitamento, usando a fórmula:

$$\text{Média de aproveitamento} = \frac{N1 + N2 * 2 + N3 * 3 + \text{Média dos exercícios}}{7}$$

A atribuição dos conceitos obedece a tabela abaixo:

Média de aproveitamento	Conceito
$\geq 9,0$	A
$\geq 7,5$ e $< 9,0$	B
$\geq 6,0$ e $< 7,5$	C
$< 6,0$	D

72. Elabore um algoritmo que, dada a idade de um nadador, classifique-o em uma das seguintes categorias:

infantil A : 5 a 7 anos
 infantil B : 8 a 10 anos
 juvenil A : 11 a 13 anos
 juvenil B : 14 a 17 anos
 sênior : maiores de 18 anos.

73. Escreva um algoritmo para ler o salário fixo e o valor das vendas efetuadas pelo vendedor de uma empresa. Sabendo-se que ele recebe uma comissão de 3 % sobre o total das vendas até R\$ 1500,00 mais 5 % sobre o que ultrapassa este valor calcular e escrever o seu salário total.

74. Um posto está vendendo combustíveis com a seguinte tabela de descontos:

Álcool: Até 20 litros, desconto de 3 % por litro.
 Acima de 20 litros, desconto de 5 % por litro.
 Gasolina: até 15 litros, desconto de 3,5 % por litro.
 Acima de 15 litros, desconto de 6 % por litro

Escreva um algoritmo que leia o número de litros vendidos, o tipo de combustível (codificado da seguinte forma: A-álcool G-Gasolina), calcule e imprima o valor a ser pago pelo cliente, sabendo-se que o preço da gasolina é de R\$ 0,53 o litro e o álcool R\$ 0,47.

75. Escreva um algoritmo que leia as medidas dos lados de um triângulo e escreva se ele é EQUILÁTERO, ISÓSCELES ou ESCALENO.

OBS: triângulo equilátero: Possui os 3 lados iguais.

triângulo isósceles: Possui 2 lados iguais.

triângulo escaleno: Possui 3 lados diferentes.

76. Escreva um algoritmo que leia a idade de 2 homens e 2 mulheres (considere que a idade dos homens será sempre diferente, assim como das mulheres). Calcule e escreva a soma das idades do homem mais velho com a mulher mais nova, e o produto das idades do homem mais novo com a mulher mais velha.

77. Escreva um algoritmo que leia o valor de 3 ângulos de um triângulo e escreva se o triângulo é acutângulo, retângulo ou obtusângulo.

OBS: triângulo retângulo: possui um ângulo reto.

triângulo obtusângulo: possui um ângulo obtuso.

triângulo acutângulo: possui 3 ângulos agudos.

78. Um mercado está vendendo frutas com a seguinte tabela de preços:

	Até 5 Kg	Acima de 5 Kg
Morango:	R\$ 5,00 p/Kg	R\$ 4,00 p/Kg
Maça:	R\$ 3,00 p/Kg	R\$ 2,00 p/Kg

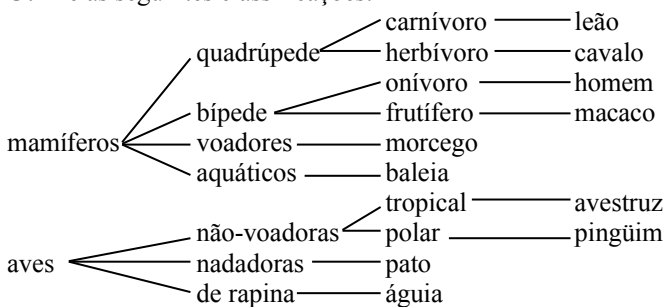
Se o cliente comprar mais de 8 Kg em frutas ou o valor total da compra ultrapassar R\$ 39,50, receberá ainda um desconto de 20 % sobre esse total. Escreva um algoritmo para ler a quantidade (em Kg) de morangos e a quantidade (em Kg) de maçãs adquiridas e escreva o valor a ser pago pelo cliente.

79. Construa um algoritmo que seja capaz de concluir qual dentre os animais seguintes foi escolhido, através de perguntas e respostas. Animais possíveis: leão, cavalo, homem, macaco, morcego, baleia, avestruz, pingüim, pato, águia.

Exemplo:

É mamífero? Sim
 É quadrúpede? Sim
 É carnívoro? Não
 É herbívoro? Sim
 Então o animal escolhido foi o cavalo.

Utilize as seguintes classificações:



80. Escreva um algoritmo para ler 2 valores e imprimir o resultado da divisão do primeiro pelo segundo. Se o segundo valor informado for ZERO, deve ser lido um novo valor. (Utilizar a estrutura REPITA).

81. Reescreva o exercício acima utilizando a estrutura ENQUANTO.

82. Acrescente uma mensagem de 'VALOR INVÁLIDO' no exercício 80, caso o segundo valor informado seja ZERO.

83. Acrescente uma mensagem de 'VALOR INVÁLIDO' no exercício 81, caso o segundo valor informado seja ZERO.

84. Escreva um algoritmo para ler as notas da 1a. e 2a. avaliações de um aluno, calcule e imprima a média semestral. Só deve aceitar valores válidos (0 a 10) para cada nota.

85. Acrescente uma mensagem de NOVO CÁLCULO (S/N)? ao final do exercício 82. Se for respondido 'S' deve retornar e executar um novo cálculo caso contrário deverá encerrar o algoritmo.

86. Escreva um algoritmo que imprima os números de 1(inclusive) a 10 (inclusive) em ordem crescente.

87. Escreva um algoritmo que imprima os números de 1(inclusive) a 10 (inclusive) em ordem decrescente.

88. Escreva um algoritmo que imprima os dez primeiros números inteiros maiores que 100.

89. Escreva um algoritmo que imprima os números de 100 até 200 de 10 em 10.
90. Ler um valor **N** e imprimir todos os valores inteiros entre 1 (inclusive) e **N** (inclusive). Considere que o **N** será sempre maior que ZERO.
91. Modifique o exercício anterior para aceitar somente valores maiores que 0 para **N**. Caso o valor informado seja menor que 1, deverá ser lido um novo valor para **N**.
92. Escreva um algoritmo que calcule e imprima a tabuada do 8 (1 a 10).
93. Ler um valor inteiro (aceitar apenas valores entre 1 e 10) e escrever a tabuada de 1 a 10 do valor lido.
94. Reescreva o exercício 86 utilizando a estrutura **REPITA** e um contador.
95. Reescreva o exercício 86 utilizando a estrutura **ENQUANTO** e um contador.
96. Ler 10 valores e escrever quantos destes valores são negativos.
97. Ler 10 valores e contar quantos estão no intervalo [10,20] e quantos deles estão fora deste intervalo, escrevendo estas informações.
98. Ler 10 valores, calcular e escrever a média aritmética destes valores.
99. Ler o número de alunos existentes em uma turma, ler as notas destes alunos, e calcular a média aritmética destas notas.
100. Ler 2 valores, calcular e escrever a soma dos inteiros existentes entre os 2 valores lidos (incluindo os valores lidos na soma). Considere que o primeiro pode ser menor que o segundo e vice-versa.
101. Ler 2 valores, calcular e escrever a soma dos quadrados dos números inteiros existentes entre eles (incluindo os 2 valores lidos). Considere que o primeiro pode ser menor que o segundo e vice-versa.
102. Escreva um programa que calcule e escreva a soma dos quadrados dos números positivos múltiplos de 3 menores que 100.

Analise os seguintes algoritmos e diga o que será impresso ao serem executados. Transcreva-os para o C sem acrescentar ou retirar comandos.

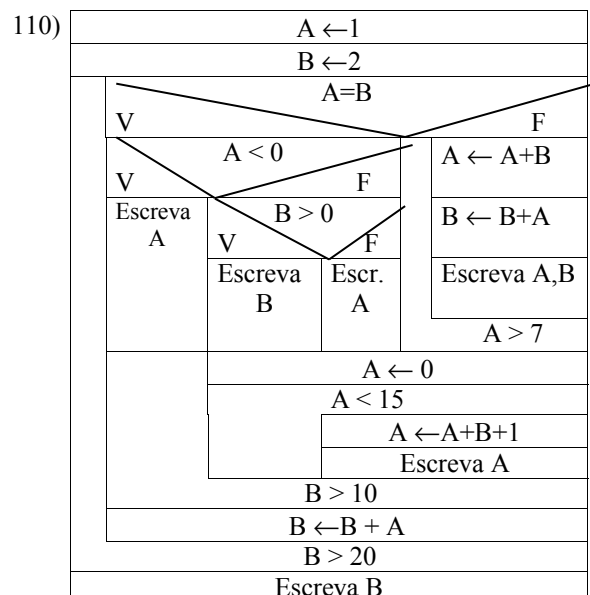
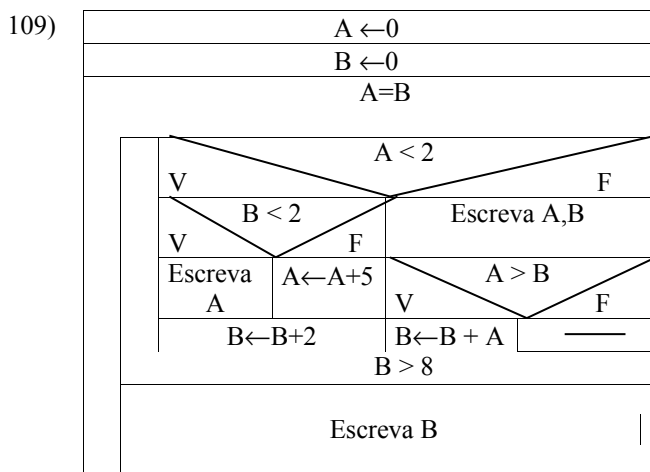
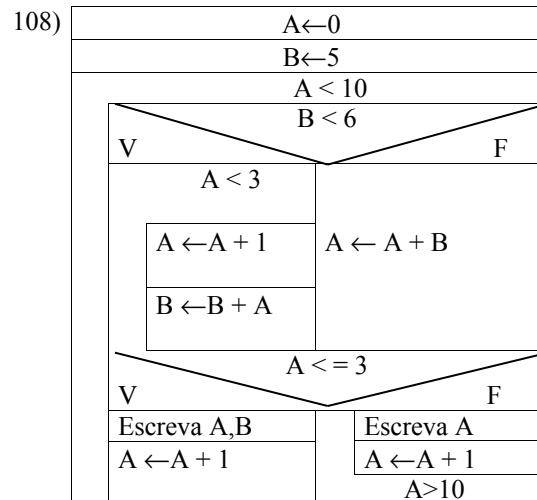
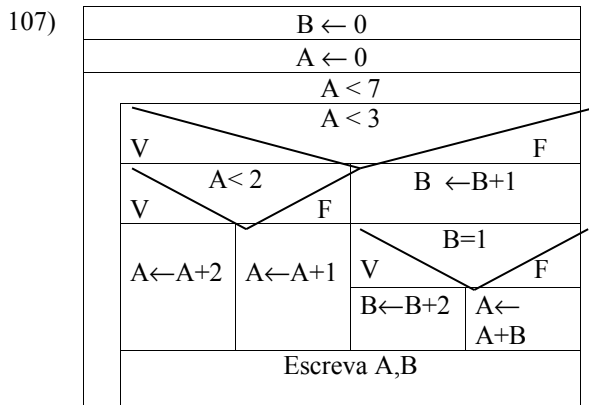
103)

$B \leftarrow 5$
$A \leftarrow 0$
$A < B$
$A \leftarrow A + 1$
$B \leftarrow B - 1$
Escreva A,B

104)

$B \leftarrow 5$
$A \leftarrow 0$
Escreva A,B
$A \leftarrow A + 1$
$B \leftarrow B - 1$
$B > A$

105. Ler um número inteiro **N** (só aceitar valores positivos-Caso o **N** não seja positivo deverá ser lido um novo valor para o **N**) e escrever os divisores de **N**.
106. Ler um número inteiro **N** (só aceitar valores positivos-Caso o **N** não seja positivo deverá ser lido um novo valor para o **N**) e escrever se é ou não **PRIMO**.



111. Ler um número indeterminado de dados, contendo cada um, a idade de um indivíduo. O último dado, que não entrará nos cálculos, contém o valor de idade igual a **ZERO**. Calcular e imprimir a idade média deste grupo de indivíduos.

112. Escreva um programa que imprima a tabuada (de 1 a 10) para os números de 1 a 10.

113. Escrever os números de 1 a 20 com seus respectivos divisores.

114. Escreva um algoritmo que imprima todos os números primos existentes entre N1 e N2, onde N1 e N2 são números naturais fornecidos pelo usuário.

115. Ler 2 valores inteiros **A** e **B**. Calcular e escrever o resultado de **A** elevado ao expoente **B**. Considere que **B** pode ser positivo, negativo ou nulo. Utilizar uma estrutura de repetição.

116. Ler um valor (só deve aceitar valores positivos) e calcular e escrever seu respectivo fatorial.

Fatorial de $n = n * (n-1) * (n-2) * (n-3) * \dots * 1$

Ex: Fatorial de $4 = 4 * 3 * 2 * 1 = 24$

117. Calcular e escrever o **maior** e o **menor** valor entre **N** valores lidos do teclado. O valor de **N** será lido do teclado.

118. Ler um conjunto de dados contendo, cada um, uma nota. Determinar e escrever a **maior** e a **menor** nota da turma. O último dado, e que não será processado, contém, nota = 99.

119. Ler o nome e a altura de cada aluno de uma turma de 10 alunos. Escrever o nome e a altura do aluno mais alto.

120. Ler um conjunto de 10 pares de dados contendo, cada um, a altura e um código para masculino (1) e outro para feminino (2) de uma turma de alunos. Calcular e escrever:

- a maior e a menor altura da turma.
- a média de altura das mulheres.
- a média de altura da turma.

121. Escreva um algoritmo para calcular e escrever o valor de S.

$$S = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{100}$$

122. Escreva um algoritmo para calcular e escrever o valor de S.

$$S = 1 + \frac{3}{2} + \frac{5}{3} + \frac{7}{4} + \dots + \frac{99}{50}$$

123. Escreva um algoritmo para calcular e escrever o valor de S.

$$S = \frac{1}{1} - \frac{2}{4} + \frac{3}{9} - \frac{4}{16} + \frac{5}{25} - \frac{6}{36} + \dots - \frac{10}{100}$$

124. Escreva um algoritmo que calcule e escreva a soma dos 30 primeiros termos da série:

$$\frac{1000}{1} - \frac{997}{2} + \frac{994}{3} - \frac{991}{4} + \dots$$

125. Uma agência de publicidade quer prestar seus serviços somente para as maiores companhias - em número de funcionários- em cada um das classificações: Grande, Média e Pequena. Escreva um algoritmo que leia um conjunto de dados que representam o código, o número de funcionários e o porte da empresa (G,M ou P) e imprima:

-O código de cada empresa com maiores recursos humanos dentro de cada categoria.

Utilizar como finalizador o código de empresa igual a 0.

126.Foi realizada uma pesquisa de algumas características físicas da população de uma certa região, a qual coletou os seguintes dados referentes a cada habitante para serem analisados:

- sexo (M-masculino, F-Feminino)
- Cor dos olhos (A-Azuis, V-Verdes, C-Castanhos)
- Cor dos cabelos (L-Louros,C-Castanhos,P-Pretos)
- idade

Escreva um algoritmo que determine e escreva:

- a maior idade dos habitantes.
- a percentagem de indivíduos do sexo feminino cuja idade está entre 18 e 35 anos inclusive e que tenham olhos verdes e cabelos louros.

O final do conjunto de habitantes é reconhecido pelo valor -1 fornecido como idade.

127.Suponha que exista um prédio de 1000 andares, onde existam três elevadores, denominados A,B e C. Para otimizar o sistema de controle dos elevadores, foi realizado um levantamento no qual cada usuário respondia:

- o elevador que utilizava com maior frequência.
- o andar ao qual se dirigia.
- o período que utilizava o elevador entre (M-matutino, V-vespertino, N-noturno).

Escreva um algoritmo que calcule e imprima:

- qual é o andar mais alto a ser utilizado.
- qual é o elevador mais freqüentado e em que horário se encontra seu maior fluxo.
- qual o horário mais usado de todos e a que elevador pertence.

128.Escreva um algoritmo para calcular e escrever o n-ésimo termo da série abaixo, onde o valor de N é lido do teclado.

5,6,11,12,17,18,23,24,

129.A seguinte sequência de números 0 1 1 2 3 5 8 13 21 ... é conhecida como série de Fibonacci. Nessa sequência, cada número, depois dos 2 primeiros, é igual à soma dos 2 anteriores. Escreva um algoritmo que mostre os 20 primeiros números dessa série.

130.Escreva um algoritmo que forneça quantos números devem existir em sequência a partir do 1 (1,2,3,4,...) para que a sua soma ultrapasse a 100 o mínimo possível.

131.No planeta Arcturus IV vive a criatura Nevrlhungri, que come precisamente 1/2 de seu suprimento de comida disponível todos os dias. Escreva um algoritmo que leia a capacidade inicial de suprimento de comida, e calcule quantos dias passarão antes de Nevrlhungri coma todo esse suprimento até atingir um quilo ou menos.

132. Escreva um algoritmo que encontre o valor mínimo de $\frac{A + B - 2C}{A \cdot B \cdot C}$ quando A,B e C podem ser qualquer um dos dígitos 1,2,3,4,5,6,7,8,9.

133. Considere uma sequência de números que tenha a propriedade que começa com o quarto deles, e cada número é a soma do anterior mais duas vezes o anterior ao anterior a ele, menos o anterior dele. Por exemplo, o 8o. deles é (o 7o. deles) mais duas vezes (o 6o. deles) menos (o 5o. deles). Suponha que os primeiros três números na sequência são: 1,2,3, escreva um algoritmo que encontre o 15o deles.

134. Ler um número indeterminado de notas de alunos, calcular e escrever as seguintes médias:

- média dos alunos aprovados (nota maior ou igual a 7)
- média dos alunos em recuperação (nota maior ou igual a 3 e inferior a 7)

O último dado que não entrará nos cálculos contém uma nota igual a 99.

135. Ler um valor A e um valor N. Imprimir a soma dos N números a partir de A (inclusive). Caso N seja negativo ou ZERO, deverá ser lido um novo N (apenas N).

Valores para teste

A	N	SOMA	
3	2	7	(3+4)
4	5	30	(4+5+6+7+8)

136. Ler 2 valores A e B. Se A for igual a B devem ser lidos novos valores para A e B. Se A for menor que B calcular e imprimir a soma dos números ímpares existentes entre A (inclusive) e B (inclusive). Se A for maior que B calcular e imprimir a média aritmética dos múltiplos de 3 existentes entre A (inclusive) e B (inclusive).

OBS: Considere que só serão informados valores inteiros positivos.

137. Ler um número indeterminado de dados, contendo cada um o peso de um indivíduo. O último dado que não entrará nos cálculos, contém um valor negativo. Calcular e imprimir:

- A média aritmética das pessoas que possuem mais de 60 Kg.
- O peso do mais pesado entre aqueles que possuem menos de 60 Kg.

138. Ler um valor X e um valor Z (se Z for menor que X deve ser lido um novo valor para Z). Contar quantos números inteiros devemos somar em sequência (a partir do X inclusive) para que a soma ultrapasse a Z o mínimo possível. Escrever o valor final da contagem.

Ex:	X	Z	Reposta	
	3	20	5	(3+4+5+6+7=25)
	2	10	4	(2+3+4+5=14)
	30	40	2	(30+31=61)

139. Supondo que a população de uma cidade A seja da ordem de 100.000 habitantes com uma taxa anual de crescimento de 3,1% ao ano e que a população de uma cidade B seja de 210.000 habitantes com uma taxa anual de crescimento de 1,5 %, escreva um algoritmo que calcule e escreva quantos anos serão necessários para que a população da cidade A ultrapasse a da cidade B, mantidas as taxas anuais de crescimento.

140. Preencher um vetor **X** de 10 elementos com o valor inteiro 30. Escrever o vetor **X**, após seu total preenchimento.

141. Preencher um vetor **A** de 10 elementos com os números inteiros 1,2,3,4,5,...,10. Escrever o vetor **A** após o seu total preenchimento.

142. Preencher um vetor **B** de 10 elementos com 1 se o índice do elemento for **ímpar**, e 0 se for **par**. Escrever o vetor **B** após o seu total preenchimento.

143. Dado o seguinte vetor:

v[1]	v[2]	v[3]	v[4]	v[5]	v[6]	v[7]	v[8]
5	1	4	2	7	8	3	6

qual será o seu conteúdo depois de executado o seguinte algoritmo ?

i de 8 até 5 passo -1
aux ← v[i]
v[i] ← v[8-i+1]
v[8-i+1] ← aux
v[3] ← v[1]
v[v[3]] ← v[v[2]]

144. Ler um vetor **C** de 10 elementos inteiros, trocar todos os valores negativos do vetor **C** por 0. Escrever o vetor **C** modificado.

145. Ler um vetor **D** de 10 elementos. Crie um vetor **E**, com todos os elementos de **D** (na mesma ordem). Escrever todo o vetor **D** e logo após todo o vetor **E**.

146. Ler um vetor **X** de 10 elementos. Crie um vetor **Y** da seguinte forma. Os elementos de ordem par de **Y** (elementos com índice 2,4,6,...,100) receberão os respectivos elementos de **X** dividido por 2. Os elementos de ordem ímpar de **Y** (elementos com índices 1,3,5,...,99) receberão os respectivos elementos de **X** multiplicados por 3. Escrever o vetor **Y**.

147. Ler um vetor **Z** de 10 elementos. Crie um vetor **W**, com todos os elementos de **Z** (na ordem inversa). Escrever o vetor **W**.

148. Ler um vetor **A** de 10 elementos e um valor **V**. Conte e escreva quantas vezes o valor **V** ocorre no vetor **A**.

149. Ler um vetor **B** de 10 elementos e um valor **X**. Escreva as posições do vetor **B** em que aparece um valor igual a **X**.

150. Ler um vetor **C** de 10 nomes e uma variável **A** que contenha o nome de uma pessoa. Escrever a mensagem **ACHEI** se o nome armazenado em **A** estiver no vetor **C**, e **NÃO ACHEI** caso contrário.

151. Ler um vetor que contenha as notas de uma turma de 10 alunos. Calcular a média da turma e contar quantos alunos obtiveram nota acima da média. Escrever a média e o resultado da contagem.

152. Ler um vetor **X** de 10 elementos. A seguir copie todos os valores negativos de vetor **X** para o vetor **R** (sem deixar elementos vazios entre os valores copiados). Escrever o vetor **R**.

153. Ler 2 vetores: **R** de 5 elementos e **S** de 10 elementos. Gere um vetor **X** de 15 elementos cuja as 5 primeiras posições contenha os elementos de **R** e as 10 últimas posições os elementos de **S**. Escrever o vetor **X**.

154. Ler um vetor **Q** de 10 posições (aceitar somente números positivos). Escreva a seguir o valor do maior elemento de **Q** e a respectiva posição que ele ocupa no vetor.

155. Ler um vetor **U** de 10 elementos. Troque a seguir o 1o. elemento com o último, o 2o. com o penúltimo, etc. até o 5o. com o 6o. e escreva o vetor **U** assim modificado.

156. Ler um vetor **M** de 10 elementos. Troque a seguir o 1o. elemento com o 6o., o 2o. com o 7o., etc. até o 5o. com o 10o. e escreva o vetor **M** assim modificado.

157. Ler 2 vetores, **R** de 5 elementos e **S** de 10 elementos. Gere um vetor **X** que possua os elementos comuns a **R** e a **S**. Considere que no mesmo vetor não haverá números repetidos. Escrever o vetor **X**.

158. Ler um vetor **R** de 5 elementos contendo o gabarito da LOTO. A seguir ler um vetor **A** de 10 elementos contendo uma aposta. A seguir imprima quantos pontos fez o apostador.

159. Ler um vetor **D** de 20 elementos. Compacte, a seguir o vetor **D**, retirando dele todos os valores nulos ou negativos. Escrever o vetor compactado.

160. Ler as notas finais de uma turma de 10 alunos armazenando-as em um vetor **N**. A seguir calcule a média aritmética das notas dos alunos aprovados (nota maior ou igual a 6). Armazene em um vetor **P** a posição (índice) que cada aluno ocupa no vetor **N**, dos alunos que obtiveram nota maior que a média calculada. Imprimir a média calculada e logo após o vetor **P**.

OBS: Não deixar valores em branco entre os elementos de **P**.

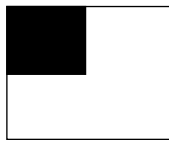
161. Ler 2 vetores **X** e **Y** de 10 elementos cada um (ocupando as posições de 1 a 10 em cada vetor). Intercalar os elementos desses 2 vetores formando assim, um novo vetor **R** de 20 elementos, onde nas posições ímpares de **R** estejam os elementos de **X** e nas posições pares os elementos de **Y**. Após a geração completa do vetor **R**, escreva-o.

162. Ler uma matriz **M** 5 x 5, calcular e escrever as seguintes somas:

- a) da linha 4 de **M**
- b) da coluna 2 de **M**
- c) da diagonal principal
- d) da diagonal secundária
- e) de todos os elementos da matriz

163. Ler uma matriz **A** de 4 x 4, calcular e escrever as somas das partes hachuradas. Utilizar estruturas de repetição.

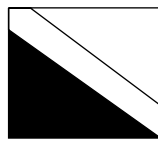
a)



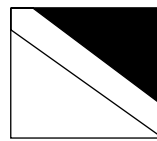
b)



c)



d)



164. Ler 2 matrizes, **A** 4 x 6 e **B** 4 x 6 e criar :

- a) uma matriz **S** que seja a soma de **A** e **B**.
- b) uma matriz **D** que seja a diferença de **A** e **B**. (**A** - **B**).

Escrever as matrizes **S** e **D**.

165. Ler uma matriz **G** 5 x 5 e criar 2 vetores **SL** e **SC** de 5 elementos que contenham respectivamente as somas das linhas e das colunas de **G**. Escrever os vetores criados.

166. Ler uma matriz **D** 5 x 5 (considere que não serão informados valores duplicados). A seguir ler um número **X** e escreva uma mensagem indicando se o valor de **X** existe ou NÃO na matriz.

167. Ler uma matriz **A** 12 x 13 e divida todos os 13 elementos de cada uma das 12 linhas de **A** pelo valor do maior elemento daquela linha. Escrever a matriz **A** modificada.

168. Ler um vetor **G** de 13 elementos que contenha o gabarito da loteria esportiva codificado da seguinte forma: 1-coluna um, 2-coluna do meio, 3-coluna dois.

Logo após, ler uma matriz 13 x 3 que contenha a aposta de um jogador. Considere que cada posição da matriz armazenará o valor 1 se for apostado, 0 caso contrário. Calcular e escrever o número de pontos obtidos pelo jogador. Escrever também o número de apostas simples, dupla ou tripla utilizadas pelo apostador.

UNIVERSIDADE CATOLICA DE PELOTAS - Escola de informatica
Algoritmos e Programacao - 040139

Escola de Engenharia e Arquitetura
Solução em linguagem C traduzida para Engenharia Computacional I

Solucao dos exercicios de 001 a 168

Exercicio 001

Algoritmo para preparar uma limonada

- 1.Cortar o limao.
- 2.Espremer o limao colocando o suco em um copo.
- 3.Acrescentar agua.
- 4.Colocar 3 colheres de acucar.
- 5.Mexer.

Exercicio 002

Algoritmo para atravessar o bode, a alfafa e o lobo para a outra margem.

- 1.Levar o BODE para a margem B.
- 2.Voltar VAZIO para a margem A.
- 3.Levar a ALFAFA para margem B.
- 4.Voltar com o BODE para a margem A.
- 5.Levar o LOBO para a margem B.
- 6.Voltar VAZIO para a margem A.
- 7.Levar o BODE para a margem B.

Exercicio 003

- 1.Calcular o resultado de 34×3 e anotar o resultado.
- 2.Calcular o resultado de 5×15 e anotar o resultado.
- 3.Calcular a soma do valor anotado no passo 1 com o valor anotado no passo 2

Exercicio 004

```
inicio
  Raio <-- 3
  Comp <-- 2 x Raio x 3,14
  Escreva Comp
fim
```

Exercicio 005

```

inicio
  A <-- 10
  B <-- 20
  C <-- A
  A <-- B
  B <-- C
fim

```

Exercicio 006

a) 20 b) 50 c) 10 10 10 d) 12 e) Valor indeterminado
 10 5 10 50 14
 30 10 40

Exercicio 007

Em português estruturado		/* em C */
		#include <stdio.h>
		void main(void)
inicio		{
real: n1,n2,m		float n1,n2,m;
n1 <-- 8;		n1 = 8.0;
n2 <-- 6;		n2 = 6.0;
m <-- (n1+n2)/2;		m = (n1+n2)/2;
escreva(m)		printf("%f",m);
fim		}

Exercicio 008

a) 23 d) 21.0 g) 3 j) 2
 b) 8.0 e) 9.0 h) 5
 c) 1 f) 15.5 i) 84

Exercicio 009

a) Sim b) Nao c) Nao

Exercicio 010

a = 6 * (3+2);	f = 6/3 + 8/2;
b = 2 + 6*(3+2);	g = (3+8/2)*4 + 3*2;
c = 2 + 3*6/(2+4);	h = 6*3*3 + 6 - 10;
d = 2 * 8/(3+1);	i = (10*8+3)*9;
e = 3+(13-2)/(2*(9-2));	j = -12 * -4 + 3 * -4

Exercicio 011

```
a = (B+C) / (D+E);
b = A/(sqr(C)+D);
c = sqr(A)/(B-1);
d = A/(B+E) - E/(F*G);
e = ( -B + sqrt( sqr(B)-4*A*C ) ) / (2*A);
f = (A-B)/(C+D) + (D+E)/(A-B);
g = r*r*r*i + y;
h = (A+B)/sqr(C) + (D-G)/(sqr(E)+sqr(F));
i = sin(PI/2);
j = sqr( sin(3*PI/2) );
k = E / sqrt( sqr(R) + sqr( 2*PI*F*L-1/(2*PI*F*C) ) );
```

obs.:

- 1) em C `sqr(x)` deve ser escrito `pow(x,2)`
- 2) em C `PI` deve ser escrito `M_PI`. O cabeçalho de funções ou constantes matemáticas está definido em `<math.h>`

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 012 - SALG-012.c
-----*/

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

void main(void)
{
    float r,area;

    printf("Raio: ");
    scanf("%f",&r);

    area = M_PI * sqrt(r);

    printf("\nArea: %6.2f",area);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 013 - SALG-013.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int valor,antec;

    printf("Digite um valor: ");
    scanf("%d",&valor);

    antec = valor--;

    printf("\nO antecessor de %d eh %d",antec, valor);
    getch();
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 014 - SALG-014.c  
-----*/
```

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

```
void main(void)  
{  
float x1,y1,x2,y2,d;  
  
printf("X1: "); scanf("%f",&x1);  
printf("\nY1: "); scanf("%f",&y1);  
printf("\nX2: "); scanf("%f",&x2);  
printf("\nY2: "); scanf("%f",&y2);  
  
d = sqrt( pow(x2-x1,2) + pow(y2-y1,2) );  
  
printf("\nDistancia: %6.2f",d);  
getch();  
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 015 - SALG-015.c  
-----*/
```

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

```
void main(void)  
{  
float fahr,celsius;  
  
printf("Entre com a temperatura em graus Fahrenheit: ");  
scanf("%f",&fahr);  
  
celsius = (fahr-32)/9 * 5;  
  
printf("\n%6.1f graus Fahrenheit correspondem a %6.1f graus Celsius.",  
fahr,celsius);  
getch();  
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 016 - SALG-016.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float fahr,celsius;

    printf("Entre com a temperatura em celsius: ");
    scanf("%f",&celsius);

    fahr = 9*celsius/5 + 32;

    printf("\n%6.1f graus Celsius correspondem a %6.1f graus Fahrenheit.",
celsius,fahr);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 017 - SALG-017.c
-----*/

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

void main(void)
{
float g,radi;

    printf("Entre com o arco em graus: ");
    scanf("%f",&g);

    radi = g * M_PI / 180;

    printf("\n%6.2f graus correspondem a %6.1f radianos",g,radi);
    getch();
}
```



```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 018 - SALG-018.c
-----*/

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

void main(void)
{
float base,altu,diag,area;

printf("Valor da base do retangulo: "); scanf("%f",&base);
printf("\nValor da altura do retangulo: "); scanf("%f",&altu);

area = base * altu;
diag = sqrt( pow(base,2) + pow(altu,2) );

printf("\nArea : %6.2f\n",area);
printf("Diagonal: %6.2f\n",diag);
getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 019 - SALG-019.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float comp,larg,altu,metr;

printf("Comprimento: "); scanf("%f",&comp);
printf("\nLargura: "); scanf("%f",&larg);
printf("\nAltura: "); scanf("%f",&altu);

metr = (comp+larg) * altu * 2;

printf("\nMetragem de azulejos: %6.2f",metr);
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 020 - SALG-020.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    NElei, NBran, Nnul, Nval;
    float  Pbran, Pnul, Pval;

    printf("Numero de eleitores: ");    scanf("%d",&NElei);
    printf("\nNumero de votos brancos: ");    scanf("%d",&NBran);
    printf("\nNumero de votos nulos: ");    scanf("%d",&Nnul);
    printf("\nNumero de votos validos: ");    scanf("%d",&Nval);

    Pbran = (float) NBran * 100 / NElei;
    Pnul  = (float) Nnul * 100 / NElei;
    Pval  = (float) Nval * 100 / NElei;

    printf("\nVotos brancos: %4.1f%%",Pbran);
    printf("\nVotos nulos  : %4.1f%%",Pnul);
    printf("\nVotos validos: %4.1f%%",Pval);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 021 - SALG-021.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    float Smens,perc,Snovo;

    printf("Valor do salario mensal: R$ ");    scanf("%f",&Smens);
    printf("\nValor do percentual de reajuste: ");    scanf("%f",&perc);

    Snovo = Smens + Smens * perc / 100;

    printf("\nValor do novo salario: R$%6.2f",Snovo);
    getch();
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 022 - SALG-022.c  
-----*/
```

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

```
void main(void)  
{  
float CFab,CCons;  
  
    printf("Valor do custo de fabrica: R$ ");  
    scanf("%f",&CFab);  
  
    CCons = CFab + CFab*28/100 + CFab*45/100;  
  
    printf("\nValor do custo ao consumidor: R$%6.2f",CCons);  
    getch();  
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 023 - SALG-023.c  
-----*/
```

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

```
void main(void)  
{  
float SFixo, Comi, Ncar, VV, Smens;  
  
    printf("Numero de carros vendidos: ");      scanf("%f",&Ncar);  
    printf("\nValor total das vendas: R$ ");      scanf("%f",&VV);  
    printf("\nValor do salario fixo: R$ ");      scanf("%f",&SFixo);  
    printf("\nComissao por carro vendido: R$ ");  scanf("%f",&Comi);  
  
    Smens = SFixo + Ncar * Comi + 5*VV/100;  
  
    printf("\nSalario mensal: R$%6.2f",Smens);  
    getch();  
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 024 - SALG-024.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float Pv,Pprom;

    printf("Preco de venda: R$ "); scanf("%f",&Pv);

    Pprom = Pv - Pv * 30 / 100;

    printf("\nPreco promocional: R$%.2f",Pprom);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 025 - SALG-025.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float Sb,Vinss,Vir,Pinss,Pir;

    printf("Valor do salario bruto: R$ ");          scanf("%f",&Sb);
    printf("\nValor descontado para o INSS: R$ ");  scanf("%f",&Vinss);
    printf("\nValor descontado para o IR: R$ ");    scanf("%f",&Vir);

    Pinss = Vinss * 100 / Sb;
    Pir   = Vir * 100 / Sb;

    printf("\nPercentual utilizado para o INSS: %4.1f %%",Pinss);
    printf("\nPercentual utilizado para o IR: %4.1f %%",Pir);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 026 - SALG-026.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
float prc, pad, prt, nrc, nrd;

printf("Percentual de alunos REPROVADOS na turma C: ");
scanf("%f",&prc);
printf("\nPercentual de alunos APROVADOS na turma D: ");
scanf("%f",&pad);

nrc = 60 * prc / 100;
nrd = 20 * (100 - pad) / 100;
prt = (nrc + nrd) * 100 / 80;

printf("\nNumero de alunos REPROVADOS na turma C : %2.0f",nrc);
printf("\nNumero de alunos REPROVADOS na turma D : %2.0f",nrd);
printf("\nPercentual de alunos reprovados em relacao ao TOTAL: %4.1f %%",
prt);
getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 027 - SALG-027.c
-----*/

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

void main(void)
{
float raio, altu, area, n_litros, n_latas,custo;

printf("Informe o raio do cilindro : "); scanf("%f",&raio);
printf("\nInforme a altura do cilindro: "); scanf("%f",&altu);

area = 2*M_PI*raio * altu + M_PI*pow(raio,2);
n_litros = area/3;
n_latas = n_litros/5;
custo = n_latas * 5;

printf("\nQuantidade de latas: %4.1f",n_latas);
printf("\nCusto R$ %6.2f",custo);
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 028 - SALG-028.c
-----*/
#include <stdio.h>
#include <conio.h>
#include <math.h>

void main(void)
{
float veloc,altu,dist;

printf("Velocidade em KM/H: "); scanf("%f",&veloc);
printf("\nAltura em Km: ");      scanf("%f",&altu);

dist = sqrt( pow(altu,2) + pow(veloc*0.01,2) );

printf("\nDistancia: %6.2f KM",dist);
getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 029 - SALG-029.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
int  inicio,final,nl;
float v_rec,dist,media,custo,lucro;

printf("Odometro no inicio do dia (em Km)      : ");
scanf("%d",&inicio);
printf("\nOdometro no fim do dia (em Km)        : ");
scanf("%d",&final);
printf("\nQuantidade de combustivel gasto (em L) : "); scanf("%d",&nl);
printf("\nValor recebido dos passageiros (em R$) : ");
scanf("%f",&v_rec);

dist  = final-inicio;
media = dist/nl;
custo = nl*0.53/dist;
lucro = v_rec - nl * 0.53;

printf("\nMedia do consumo (em Km/L)      : %5.2f",media);
printf("\nCusto medio do Km (em R$/Km)    : %5.2f",custo);
printf("\nLucro do dia (em R$)              : %8.2f",lucro);
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 030 - SALG-030.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    nempr, nbic;
    float totcom, sm, pcusto, vvendas, lucro, sfinal;

    printf("\nNumero de empregados.....: ");    scanf("%d",&nempr);
    printf("\nValor do salario minimo.....: ");    scanf("%f",&sm);
    printf("\nPreco de custo de cada bicicleta: ");    scanf("%f",&pcusto);
    printf("\nNumero de bicicletas vendidas....: ");    scanf("%d",&nbic);

    totcom  = 15 * pcusto * nbic / 100;
    sfinal  = 2 * sm + totcom / nempr;
    vvendas = ( pcusto + pcusto * 50/100) * nbic;
    lucro   = vvendas - pcusto * nbic - sfinal * nempr;

    printf("\n\nValor do salario final: %8.2f",sfinal);
    printf("\nValor do lucro.....: %8.2f",lucro);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 031 - SALG-031.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float  Pcusto_dz, Pdesc, Pvenda_dz;
float  Desc_dz, Pvenda_un, Valor_R;
float  Valor_CR, VPagoCR, TrocoCR, TrocoR;
int     Nlar;

printf("Preco de custo da duzia em Reais: ");      scanf("%f",&Pcusto_dz);
printf("\nPercentual de desconto.....: ");      scanf("%f",&Pdesc);
printf("\nNumero de laranjas.....: ");          scanf("%d",&Nlar);

Pvenda_dz = Pcusto_dz * 2;
Desc_dz    = Pvenda_dz * Pdesc / 100;
Pvenda_un  = (Pvenda_dz - Desc_dz) / 12;
Valor_R     = Pvenda_un * Nlar;
Valor_CR   = Valor_R * 2750.0;

printf("\n\nValor em Cruzeiros Reais : %8.2f",Valor_CR);
printf("\nEntre com o valor pago em Cruzeiros Reais: ");
scanf("%f", &VPagoCR);

TrocoCR = VPagoCR - Valor_CR;
TrocoR  = TrocoCR / 2750.0;

printf("\n\nValor do troco em Reais: %6.2f",TrocoR);
getch();
}
```



```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 032 - SALG-032.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float  ComprM, consumo ,total_percurso, dist_reab, n_litros;
int    Nvoltas, Nreab;

    printf("Comprimento da pista em metros: ");    scanf("%f",&ComprM);
    printf("\nNumero de voltas da corrida...: ");    scanf("%d",&Nvoltas);
    printf("\nNumero de reabastecimentos....: ");    scanf("%d",&Nreab);
    printf("\nConsumo de combustivel(Km/L)...: ");    scanf("%f",&consumo);

    total_percurso = ComprM / 1000 * Nvoltas;
    dist_reab      = total_percurso / (Nreab + 1);
    n_litros       = dist_reab / consumo;

    printf("\n\nNumero minimo de litros : %5.1f",n_litros);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 033 - SALG-033.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float n1,n2,media;

    printf("Informe a primeira nota: ");    scanf("%f",&n1);
    printf("\nInforme a segunda nota : ");    scanf("%f",&n2);

    media = (n1+n2)/2;

    if (media >=6)
        printf("\nParabens! Voce foi aprovado");
    printf("\nMedia: %5.2f",media);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 034 - SALG-034.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float n1,n2,media;

printf("Informe a primeira nota: "); scanf("%f",&n1);
printf("\nInforme a segunda nota : "); scanf("%f",&n2);

media = (n1+n2)/2;

if (media >=6)
printf("\nParabens! Voce foi aprovado");
else
printf("\nVoce foi reprovado! Estude mais");

printf("\nMedia: %5.2f",media);
getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 035 - SALG-035.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float n;

printf("Digite um valor : "); scanf("%f",&n);

if (n > 10)
printf("\nE maior que 10");
else
printf("\nNao e maior que 10");
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 036 - SALG-036.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
float n;

printf("Digite um valor : "); scanf("%f",&n);

if (n >= 0)
printf("POSITIVO");
else
printf("NEGATIVO");
getch();
}
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 037 - SALG-037.c
-----*/
#include <stdio.h>
#include <conio.h>
#include <string.h>

void main(void)
{
/* Em C nao existe o tipo cadeia de caracteres (string),
formalmente definido, entao, utiliza-se um vetor (array)
de caracteres. Este assunto sera estudado mais a frente no
curso mas... */

char senha[51];          /* cria uma string de 50 caracteres mais um */
                        /* caractere reservado para uso do C */
                        /* como marcacao do fim de texto */

printf("Digite a senha: ");
scanf("%s",senha); /* Nao existe o simbolo & antes de senha */

/* A comparacao de strings deve ser realizada por uma funcao
especial chamada strcmp. O resultado de strcmp e nulo (falso)
se as strings sao iguais. O resultado de strcmp e nao nulo
(verdadeiro) se as strings sao diferentes. A definicao de
strcmp esta em string.h */

if (!strcmp(senha,"UCPEL"))
printf("Acesso permitido");
else
printf("Acesso negado");
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 038 - SALG-038.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    float ano;

    printf("Informe o ano de nascimento: "); scanf("%f",&ano);

    if (ano <= 1987)
        printf("Permitido votar");
    else
        printf("Nao e permitido votar");
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 039 - SALG-039.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int a,b;

    printf("Informe um valor: ");      scanf("%d",&a);
    printf("\nInforme outro valor: "); scanf("%d",&b);

    if (a>b)
        printf("\n\nO maior e : %d",a);
    else
        printf("\n\nO maior e : %d",b);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 040 - SALG-040.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    float h,p;
    char  sexo;

    printf("Qual a sua altura? ");      scanf("%f",&h);
    printf("\nQual o seu sexo (f/m)? ");
    sexo = getch();

    if (sexo=='f')
        p = 62.1 * h - 44.7;
    else
        p = 72.7 * h - 58;

    printf("\nSeu peso ideal e : %4.1f",p);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 041 - SALG-041.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    n;
    float  t;

    printf("Informe o numero de macas adquiridas: "); scanf("%d",&n);

    if (n<12)
        t = n * 0.30;
    else
        t = n * 0.25;

    printf("\n\nValor total da compra: %5.2f",t);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 042 - SALG-042.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int n;

    printf("Digite um numero inteiro: "); scanf("%d",&n);

    if (n%2 == 1)
        printf("E impar");
    else
        printf("E par");
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 043 - SALG-043.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int a,b;

    printf("Digite um valor: ");    scanf("%d",&a);
    printf("\nDigite outro valor: "); scanf("%d",&b);

    if (a<b)
    {
        printf("\n%d",a);
        printf("\n%d",b);
    }
    else
    {
        printf("\n%d",b);
        printf("\n%d",a);
    };
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 044 - SALG-044.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
int n;

    printf("Digite um valor: ");    scanf("%d",&n);

    if (n>0)
        printf("\nPositivo");
    else
        if (n==0)
            printf("\nZero");
        else
            printf("\nNegativo");
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 045 - SALG-045.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
float n1,n2,media;

    printf("Informe a primeira nota: ");    scanf("%f",&n1);
    printf("\nInforme a segunda nota: ");    scanf("%f",&n2);

    media = (n1+n2)/2;
    printf("\nMedia: %5.2f",media);

    if (media >= 6)
        printf("\nAprovado");
    else
        if (media < 3)
            printf("\nReprovado");
        else
            printf("\nEm exame");
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 046 - SALG-046.c
-----*/
#include <stdio.h>
#include <conio.h>
#include <math.h>

void main(void)
{
float a,b,c,d,x1,x2;

printf("A="); scanf("%f",&a);
printf("\nB="); scanf("%f",&b);
printf("\nC="); scanf("%f",&c);

d = b*b - 4*a*c;
if (d<0)
printf("\nNao ha raizes reais");
else
{
x1 = (-b+sqrt(d))/(2*a);
x2 = (-b-sqrt(d))/(2*a);
printf("\nRaizes: %5.2f e %5.2f",x1,x2);
}
getch();
}
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 047 - SALG-047.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
char n1[51],n2[51]; /* o nome dos times nao pode ter mais de 50 letras */
int g1,g2;

printf("Informe o nome do primeiro time..: "); scanf("%s",n1);
printf("\nInforme o numero de gols marcados: "); scanf("%d",&g1);
printf("\nInforme o nome do segundo time...: "); scanf("%s",n2);
printf("\nInforme o numero de gols marcados: "); scanf("%d",&g2);

if (g1>g2)
printf("\nVitoria do time: %s",n1);
else
if (g2>g1)
printf("\nVitoria do time: %s",n2);
else
printf("Empate");
getch();
}
```



```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 048 - SALG-048.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int nl,m,p,a;

printf("Informe o numero de lados: "); scanf("%d",&nl);
printf("\nInforme a medida do lado: "); scanf("%d",&m);

if (nl==3)
{
p = nl*m;
printf("\nE um triangulo");
printf("\nPerimetro: %d",p);
}
else
if (nl==4)
{
a = m*m;
printf("\nE um quadrado.");
printf("\nArea: %d",a);
}
else
if (nl==5)
printf("\nE um pentagono");
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 049 - SALG-049.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int nl,m,p,a;

printf("Informe o numero de lados: "); scanf("%d",&nl);
printf("\nInforme a medida do lado: "); scanf("%d",&m);

if (nl<3)
printf("Nao e um poligono");
else
if (nl==3)
{
p = nl*m;
printf("\nE um triangulo");
printf("\nPerimetro: %d",p);
}
else
if (nl==4)
{
a = m*m;
printf("\nE um quadrado");
printf("\nArea: %d",a);
}
else
if (nl==5)
printf("\nE um pentagono");
else
printf("\nPoligono nao identificado");
getch();
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 050 - SALG-050.c  
-----*/  
  
#include <stdio.h>  
#include <conio.h>  
  
void main(void)  
{  
int a,b;  
  
    printf("A=");    scanf("%d",&a);  
    printf("\nB=");  scanf("%d",&b);  
  
    printf("\nAmarelo");  
    if (a==1)  
    {  
        printf("\nAzul");  
        printf("\nVerde");  
        printf("\nRoxo");  
    }  
    else  
    {  
        printf("\nPera");  
        if (b==1)  
            printf("\nMaca");  
        else  
            printf("\nMorango");  
    }  
    printf("\nUva");  
    getch();  
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 051 - SALG-051.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int a,b,c;

printf("A="); scanf("%d",&a);
printf("\nB="); scanf("%d",&b);
printf("\nC="); scanf("%d",&c);

if (a==1)
printf("\nVerde");
else
if (b==2)
if (c==3)
printf("\nUCPel");
else
{
printf("\nUva");
printf("\nPera");
}
else
printf("\nPreto");
printf("\nVioleta");
getch();
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 052 - SALG-052.c  
-----*/  
  
#include <stdio.h>  
#include <conio.h>  
  
void main(void)  
{  
int a,b,c;  
  
    printf("A=");    scanf("%d",&a);  
    printf("\nB=");  scanf("%d",&b);  
    printf("\nC=");  scanf("%d",&c);  
  
    if (a==1)  
    {  
        printf("\nMorango");  
        if (b==1)  
            if (c==1)  
                printf("\nPera");  
            else  
                printf("\nAbacate");  
    }  
    printf("\nLimao");  
    getch();  
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 053 - SALG-053.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int a,b,c;

printf("A="); scanf("%d",&a);
printf("\nB="); scanf("%d",&b);
printf("\nC="); scanf("%d",&c);

if (a==1)
    if (b==1)
        {
            printf("\nA");
            printf("\nB");
        }
printf("\nC");
if (c==1)
    {
        printf("\nD");
        printf("\nE");
    }
else
    printf("\nF");
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 054 - SALG-054.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int a,b,c,d,e;

printf("A="); scanf("%d",&a);
printf("\nB="); scanf("%d",&b);
printf("\nC="); scanf("%d",&c);
printf("\nD="); scanf("%d",&d);
printf("\nE="); scanf("%d",&e);

if (a==1)
{
printf("\nAmarelo");
if (b==1)
printf("\nAzul");
printf("\nPreto");
if (d==1)
printf("\nVerde");
else
printf("\nVioleta");
}
else
{
if (c==1)
printf("\nVermelho");
if (e==1)
{
printf("\nLaranja");
printf("\nCinza");
}
}
printf("\nBranco");
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 055 - SALG-055.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int a,b,c;

printf("Informe o primeiro valor: "); scanf("%d",&a);
printf("\nInforme o segundo valor : "); scanf("%d",&b);
printf("\nInforme o terceiro valor: "); scanf("%d",&c);

if (a>b)
    if (a>c)
        printf("\n%d e o maior",a);
    else
        printf("\n%d e o maior",c);
else
    if (b>c)
        printf("\n%d e o maior",b);
    else
        printf("\n%d e o maior",c);
getch();
}
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 056 - SALG-056.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int a,b,c,s;
printf("Informe o primeiro valor: "); scanf("%d",&a);
printf("\nInforme o segundo valor : "); scanf("%d",&b);
printf("\nInforme o terceiro valor: "); scanf("%d",&c);

if (a>b)
    if (b>c)
        s = a + b;
    else
        s = a + c;
else
    if (a>c)
        s = b + a;
    else
        s = b + c;

printf("\nSoma: %d",s);
getch();
}
```



```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 057 - SALG-057.c  
-----*/  
  
#include <stdio.h>  
#include <conio.h>  
  
void main(void)  
{  
int a,b,c;  
  
printf("Informe o primeiro valor: "); scanf("%d",&a);  
printf("\nInforme o segundo valor : "); scanf("%d",&b);  
printf("\nInforme o terceiro valor: "); scanf("%d",&c);  
  
if (a<b)  
    if (b<c)  
        printf("\n%d %d %d", a,b,c);  
    else  
        if (a<c)  
            printf("\n%d %d %d", a,c,b);  
        else  
            printf("\n%d %d %d", c,a,b);  
else  
    if (a<c)  
        printf("\n%d %d %d", b,a,c);  
    else  
        if (b<c)  
            printf("\n%d %d %d", b,c,a);  
        else  
            printf("\n%d %d %d", c,b,a);  
getch();  
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 058 - SALG-058.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int a,b,c;

printf("Informe a medida do primeiro lado: "); scanf("%d",&a);
printf("\nInforme a medida do segundo lado : "); scanf("%d",&b);
printf("\nInforme a medida do terceiro lado: "); scanf("%d",&c);

if (a < b+c)
    if (b < a+c)
        if (c < a+b)
            printf("\nFormam um triangulo");
        else
            printf("\nNAO formam um triangulo");
    else
        printf("\nNAO formam um triangulo");
else
    printf("\nNAO formam um triangulo");
getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 059 - SALG-059.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
float x,y;

printf("X: "); scanf("%f",&x);
printf("\nY: "); scanf("%f",&y);

if (x>0)
    if (y>0)
        printf("\n1o. Quadrante");
    else
        printf("\n4o. Quadrante");
else
    if (y>0)
        printf("\n2o. Quadrante");
    else
        printf("\n3o. quadrante");
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 060 - SALG-060.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
float x,y;
printf("X: "); scanf("%f",&x);
printf("\nY: "); scanf("%f",&y);

if (x==0)
printf("\nNAO esta em nenhum quadrante");
else
if (y==0)
printf("\nNAO esta em nenhum quadrante");
else
if (x>0)
if (y>0)
printf("\n1o. Quadrante");
else
printf("\n4o. Quadrante");
else
if (y>0)
printf("\n2o. Quadrante");
else
printf("\n3o. Quadrante");

getch();
}
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 063 - SALG-063.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int a,b,c;
printf("\nInforme o primeiro valor: "); scanf("%d",&a);
printf("\nInforme o segundo valor : "); scanf("%d",&b);
printf("\nInforme o terceiro valor: "); scanf("%d",&c);

if ((a>b) && (a>c))
printf("\n %d e o maior", a);
else
if ((b>a) && (b>c))
printf("\n %d e o maior", b);
else
printf("\n %d e o maior", c);
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 064 - SALG-064.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int a,b,c,s;
printf("\nInforme o primeiro valor: "); scanf("%d",&a);
printf("\nInforme o segundo valor : "); scanf("%d",&b);
printf("\nInforme o terceiro valor: "); scanf("%d",&c);
if ((a<b) && (a<c))
    s = b + c;
else
    if ((b<a) && (b<c))
        s = a + c;
    else
        s = a + b;

printf("\nSoma: %d", s);
getch();
}
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 065 - SALG-065.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int a,b,c;
printf("\nInforme o primeiro valor: "); scanf("%d",&a);
printf("\nInforme o segundo valor : "); scanf("%d",&b);
printf("\nInforme o terceiro valor: "); scanf("%d",&c);
if ((a<b) && (b<c))
    printf("\n %d %d %d", a,b,c);
else
    if ((a<c) && (c<b))
        printf("\n %d %d %d", a,c,b);
    else
        if ((b<a) && (a<c))
            printf("\n %d %d %d", b,a,c);
        else
            if ((b<c) && (c<a))
                printf("\n %d %d %d", b,c,a);
            else
                if ((c<a) && (a<b))
                    printf("\n %d %d %d", c,a,b);
                else
                    printf("\n %d %d %d", c,b,a);
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 066 - SALG-066.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int a,b,c;
printf("\nInforme a medida do primeiro lado: "); scanf("%d",&a);
printf("\nInforme a medida do segundo lado : "); scanf("%d",&b);
printf("\nInforme a medida do terceiro lado: "); scanf("%d",&c);

if ((a<b+c) && (b<a+c) && (c<a+b))
printf("\nFormam um triangulo");
else
printf("\nNAO formam um triangulo");
getch();
}
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 067 - SALG-067.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
float x,y;

printf("\nX: "); scanf("%f",&x);
printf("\nY: "); scanf("%f",&y);

if ((x==0) && (y==0))
printf("\nEsta na ORIGEM");
else
if ((x==0) && (y!=0))
printf("\nEsta no EIXO Y");
else
if ((x!=0) && (y==0))
printf("\nEsta no EIXO X");
else
if (x>0)
if (y>0)
printf("\n1o. Quadrante");
else
printf("\n4o. Quadrante");
else
if (y>0)
printf("\n2o. Quadrante");
else
printf("\n3o. Quadrante");
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 068 - SALG-068.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int hi,hf,dur;

    printf("\nDigite a hora inicial do jogo: "); scanf("%d",&hi);
    printf("\nDigite a hora final do jogo..: "); scanf("%d",&hf);

    if (hi<hf)
        dur = hf-hi;
    else
        dur = 24-hi+hf;

    printf("\nDuracao do jogo: %d", dur);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 069 - SALG-069.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int hi,mi,hf,mf,tiMin,tfMin,dur,durH,durM;

    printf("\nDigite a hora inicial do jogo (em horas e minutos) ");
    printf("\nHora: ");    scanf("%d",&hi);
    printf("\nMinuto: "); scanf("%d",&mi);

    printf("\nDigite a hora final do jogo (em horas e minutos) ");
    printf("\nHora: ");    scanf("%d",&hf);
    printf("\nMinuto: "); scanf("%d",&mf);

    tiMin = hi*60 + mi;
    tfMin = hf*60 + mf;

    if (tiMin < tfMin)
        dur = tfMin-tiMin;
    else
        dur = 24*60-tiMin+tfMin;

    durH = dur / 60;
    durM = dur % 60;
    printf("\nDuracao do jogo: %d horas %d minutos.",durH,durM);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 070 - SALG-070.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    ht,he;
    float  sh,sf;

    printf("\nNumero de horas trabalhadas no mes: "); scanf("%d",&ht);
    printf("\nSalario por hora: ");                 scanf("%f",&sh);

    if (ht > 160)
    {
        he = ht-160;
        sf = sh * 160 + he * sh * 1.5;
    }
    else
        sf = sh * ht;

    printf("\nSalario total do funcionario: %8.2f",sf);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 071 - SALG-071.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float n1,n2,n3,Mexerc,Maprov;
char C;

printf("\nNota 1: "); scanf("%f",&n1);
printf("\nNota 2: "); scanf("%f",&n2);
printf("\nNota 3: "); scanf("%f",&n3);
printf("\nMedia dos exercicios: "); scanf("%f",&Mexerc);

Maprov = (n1 + n2*2 + n3*3 + Mexerc)/7;

if (Maprov >= 9.0)
    C = 'A';
else
    if (Maprov >= 7.5)
        C = 'B';
    else
        if (Maprov >= 6.0)
            C = 'C';
        else
            C = 'D';

printf("\nMedia...: %5.2f", Maprov);
printf("\nConceito: %c", C);
getch();
}
```



```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 072 - SALG-072.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int idade;

printf("\nInforme a idade do nadador: ");
scanf("%d",&idade);

switch idade
{
case 5:
case 6:
case 7: printf("\nCategoria infantil A");
break;
case 8:
case 9:
case 10: printf("\nCategoria infantil B");
break;
case 11:
case 12:
case 13: printf("\nCategoria juvenil A");
break;
case 14:
case 15:
case 16:
case 17: printf("\nCategoria juvenil B");
break;
default:
if (idade>=18)
printf("\nCategoria senior");
}
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 073 - SALG-073.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float sf,vv,c,st;

printf("\nSalario fixo....: "); scanf("%f",&sf);
printf("\nValor das vas: ");    scanf("%f",&vv);

if (vv > 1500.0)
    c = 3*1500/100 + 5 * (vv-1500)/100;
else
    c = 3 * vv / 100;

st = sf + c;

printf("\nSalario total: %8.2f", st);
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 074 - SALG-074.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float nl, valorL, pd, valor, valorDesc, valorPg;
char tc;

printf("\nInforme o numero de litros...: "); scanf("%f", &nl);
printf("\nTipo de combustivel (a/g) ? ");
tc = getch();

if (tc=='a')
{
    valorL = 0.47;
    if (nl <= 20)
        pd = 3.0;
    else
        pd = 5.0;
}
else
{
    valorL = 0.53;
    if (nl <= 15)
        pd = 3.5;
    else
        pd = 6.0;
}

valor      = nl * valorL;
valorDesc  = valor * pd / 100;
valorPg    = valor - valorDesc;

printf("\nValor a pagar: R$ %6.2f", valorPg);
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 075 - SALG-075.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int a,b,c;

printf("\nA= "); scanf("%d",&a);
printf("\nB= "); scanf("%d",&b);
printf("\nC= "); scanf("%d",&c);

if ((a==b) && (b==c))
    printf("\nEquilatero");
else
    if ((a==b) || (b==c) || (a==c))
        printf("\nIsosceles");
    else
        printf("\nEscaleno");
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 076 - SALG-076.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int h1,h2,m1,m2,s,p;

printf("\nIdade do Homem 1: "); scanf("%d",&h1);
printf("\nIdade do Homem 2: "); scanf("%d",&h2);
printf("\nIdade da Mulher 1: "); scanf("%d",&m1);
printf("\nIdade da Mulher 2: "); scanf("%d",&m2);

if (h1 > h2)
    if (m1 < m2)
    {
        s = h1 + m1;
        p = h2 * m2;
    }
    else
    {
        s = h1 + m2;
        p = h2 * m1;
    }
else
    if (m1 < m2)
    {
        s = h2 + m1;
        p = h1 * m2;
    }
    else
    {
        s = h2 + m2;
        p = h1 * m1;
    }

printf("\nSoma...: %d", s);
printf("\nProduto: %d", p);
getch();
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 077 - SALG-077.c  
-----*/  
  
#include <stdio.h>  
#include <conio.h>  
  
void main(void)  
{  
int a,b,c;  
  
printf("\nA= "); scanf("%d",&a);  
printf("\nB= "); scanf("%d",&b);  
printf("\nC= "); scanf("%d",&c);  
  
if ((a<90) && (b<90) && (c<90))  
    printf("\nTriangulo acutangulo");  
else  
    if ((a=90) || (b=90) || (c=90))  
        printf("\nTriangulo retangulo");  
    else  
        printf("\nTriangulo obtusangulo");  
getch();  
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 078 - SALG-078.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float Qmor,Qmac,Pmor,Pmac,Qfrutas,Vfrutas,Vdesc,Vfinal;

printf("\nQuantidade de morangos: "); scanf("%f",&Qmor);
printf("\nQuantidade de macas...: "); scanf("%f",&Qmac);

if (Qmor <= 5)
    Pmor = 5.0;
else
    Pmor = 4.0;

if (Qmac <= 5)
    Pmac = 3.0;
else
    Pmac = 2.0;

Qfrutas = Qmor + Qmac;
Vfrutas = Qmor * Pmor + Qmac * Pmac;

if ((Vfrutas> 39.50) || (Qfrutas > 8.0))
    Vdesc = Vfrutas * 20 / 100;
else
    Vdesc = 0;

Vfinal = Vfrutas - Vdesc;
printf("\nValor final: R$ %6.2f",Vfinal);
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 079 - SALG-079.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
char m,q,c,h,b,o,f,v,a;

printf("\nE mamifero? ");
m = getche();

if (m=='s')
{
printf("\nE quadrupede? ");
q = getche();
if (q=='s')
{
printf("\nE carnivoros? ");
c = getche();
if (c=='s')
printf("\nLEAO");
else
{
printf("\nE herbivoro? ");
h = getche();
if (h=='s')
printf("\nCAVALO");
}
}
else
{
printf("\nE bipede? ");
b = getche();
if (b=='s')
{
printf("\nE onivoro? ");
o = getche();
if (o=='s')
printf("\nHOMEM");
else
{
printf("\nE frutifero? ");
f = getche();
if (f=='s')
printf("\nMACACO");
}
}
else
{
printf("\nE voador? ");
```



```
        v = getche();
        if (v=='s')
            printf("\nMORCEGO");
        else
        {
            printf("\nE aquatico? ");
            a = getche();
            if (a=='s')
                printf("\nBALEIA");
            }
        }
    }
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 080 - SALG-080.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float a,b,r;

    printf("\nDigite um valor: "); scanf("%f",&a);
    do
    {
        printf("\nDigite outro valor: ");
        scanf("%f",&b);
    } while (b==0);

    r = a/b;

    printf("\nResultado da divisao: %6.2f", r);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 081 - SALG-081.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float a,b,r;

    printf("\nDigite um valor: ");    scanf("%f",&a);
    printf("\nDigite outro valor: "); scanf("%f",&b);

    while (b==0)
    {
        printf("\nDigite outro valor: "); scanf("%f",&b);
    }

    r = a/b;

    printf("\nResultado da divisao: %6.2f", r);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 082 - SALG-082.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
float a,b,r;

printf("\nDigite um valor: "); scanf("%f",&a);
do
{
printf("\nDigite outro valor: "); scanf("%f",&b);
if (b==0)
printf("\nVALOR INVALIDO");
} while (b==0);

r = a/b;

printf("\nResultado da divisao: %6.2f", r);
getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 083 - SALG-083.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
float a,b,r;

printf("\nDigite um valor: "); scanf("%f",&a);
printf("\nDigite outro valor: "); scanf("%f",&b);
while (b==0)
{
printf("\nVALOR INVALIDO");
printf("\nDigite outro valor: ");
scanf("%f",&b);
}

r = a/b;

printf("\nResultado da divisao: %6.2f", r);
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 084 - SALG-084.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
float n1,n2,m;
do
{
printf("\nDigite a primeira nota: "); scanf("%f",&n1);
} while ((n1>10) || (n1<0));

do
{
printf("\nDigite a segunda nota: "); scanf("%f",&n2);
} while ((n2<0) || (n2>10));

m = (n1+n2)/2;

printf("\nMedia semestral: %6.2f", m);
getch();
}
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 085 - SALG-085.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
float a,b,r;
char x;
do
{
printf("\nDigite um valor: "); scanf("%f",&a);
do
{
printf("\nDigite outro valor: "); scanf("%f",&b);
if (b==0)
printf("\nVALOR INVALIDO");
} while (b==0);

r = a/b;

printf("\nResultado da divisao: %6.2f", r);
printf("\nDeseja um novo calculo(s/n)?");
x = getche();
} while (x!='n');
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 086 - SALG-086.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int i;

    for (i=1;i<=10;i++)
        printf("\n%d", i);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 087 - SALG-087.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int i;

    for (i=10;i>=1;i--)
        printf("\n%d", i);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 088 - SALG-088.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int a;

    for (a=101;a<=110;a++)
        printf("\n%d", a);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 089 - SALG-089.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
    int i;

    for (i=100;i<=200;i+=10)
        printf("\n%d", i);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 090 - SALG-090.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
    int n,i;
    printf("\nDigite um valor: "); scanf("%d",&n);

    for (i=1;i<=n;i++)
        printf("\n%d", i);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 091 - SALG-091.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
    int n,i;
    do {
        printf("\nDigite um valor: "); scanf("%d",&n);
    } while (n<=0);

    for (i=1;i<=n;i++)
        printf("\n%d", i);
    getch();
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 092 - SALG-092.c  
-----*/
```

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

```
void main(void)  
{  
    int r,i;  
  
    for (i=1;i<=10;i++)  
    {  
        r = i * 8;  
        printf("\n8 x %2d = %2d", i,r);  
    }  
    getch();  
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 093 - SALG-093.c  
-----*/
```

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

```
void main(void)  
{  
    int r,i,n;  
    do  
    {  
        printf("\nDigite um numero entre 1 e 10: ");  
        scanf("%d",&n);  
    } while ((n<1) || (n>10));  
  
    for (i=1;i<=10;i++)  
    {  
        r = i * n;  
        printf("\n %2d x %2d = %3d", n,i,r);  
    }  
    getch();  
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 094 - SALG-094.c  
-----*/
```

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

```
void main(void)  
{  
int c;  
    c = 0;  
    do  
    {  
        c=c+1;  
        printf("\n%d", c);  
    } while (c!=10);  
    getch();  
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 095 - SALG-095.c  
-----*/
```

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

```
void main(void)  
{  
int c;  
    c = 0;  
    while (c < 10)  
    {  
        c=c+1;  
        printf("\n%d", c);  
    }  
    getch();  
}
```



```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 096 - SALG-096.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int c,i,x;
    c = 0;
    for (i=1;i<=10;i++)
    {
        printf("\nDigite um numero: ");
        scanf("%d",&x);
        if (x<0)
            c = c+1;
    }
    printf("\nForam digitados %d valores negativos", c);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 097 - SALG-097.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int d,f,i,x;
    d = 0;
    f = 0;
    for (i=1;i<=10;i++)
    {
        printf("\nDigite um numero: ");
        scanf("%d",&x);
        if ((x>=10) && (x<=20))
            d = d+1;
        else
            f = f+1;
    }
    printf("\nForam digitados %d valores dentro do intervalo [10,20]", d);
    printf("\nForam digitados %d valores fora do intervalo [10,20]", f);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 098 - SALG-098.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    s,i,v;
    float m;
    s = 0;
    for (i=1;i<=10;i++)
    {
        printf("\nDigite o %d o. Valor: ", i);
        scanf("%d",&v);
        s = s + v;
    }
    m = (float) s/10;
    printf("\nMedia aritmetica: %6.2f", m);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 099 - SALG-099.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    na,i;
    float media,s,nota;
    s = 0;
    printf("\nEntre com o numero de alunos da turma: ");
    scanf("%d",&na);
    for (i=1;i<=na;i++)
    {
        printf("\nDigite a nota do %d o. Aluno: ", i);
        scanf("%f",&nota);
        s = s + nota;
    }
    media = s/na;
    printf("\nMedia aritmetica: %6.2f", media);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 100 - SALG-100.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int a,b,i,s;

printf("\nEntre com o primeiro valor: ");
scanf("%d",&a);
printf("\nEntre com o segundo valor: ");
scanf("%d",&b);

s = 0;
if (a<b)
    for (i=a;i<=b;i++)
        s = s + i;
else
    for (i=b;i<=a;i++)
        s = s+i;

printf("\nSoma: %d",s);
getch();
}
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 101 - SALG-101.c
-----*/
#include <stdio.h>
#include <conio.h>
#include <math.h>
void main(void)
{
int a,b,i,s;

printf("\nEntre com o primeiro valor: ");
scanf("%d",&a);
printf("\nEntre com o segundo valor: ");
scanf("%d",&b);

s = 0;
if (a<b)
    for (i=a;i<=b;i++)
        s = s + pow(i,2);
else
    for (i=b;i<=a;i++)
        s = s + pow(i,2);

printf("\nSoma: %d", s);
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 102 - SALG-102.c
-----*/

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

void main(void)
{
    int    i;
    float  s;

    s = 0;
    for (i=1;i<=100;i++)
        if (i%3 == 0)
            s = s + pow(i,2);
    getch();
    printf("\nSoma: %10.2f",s);
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 103 - SALG-103.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int a,b;

    b = 5;
    a = 0;
    while (a<b)
    {
        a = a + 1;
        b = b - 1;
        printf("%d %d", a,b);
    }
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 104 - SALG-104.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int a,b;

    b = 5;
    a = 0;
    do
    {
        printf("%d %d", a,b);
        a = a + 1;
        b = b - 1;
    } while (b<=a);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 105 - SALG-105.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int n,i;

    do
    {
        printf("\nEntre com um numero: ");
        scanf("%d",&n);
    } while (n<=0);

    printf("\nOs divisores de %d sao: " ,n);
    for (i=1;i<=n;i++)
        if (n%i == 0)
            printf("\n%d");
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 106 - SALG-106.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int n,i,c;

    do
    {
        printf("\nEntre com um numero: ");
        scanf("%d",&n);
    } while (n<=0);

    c = 0;
    for (i=1;i<=n;i++)
        if (n%i == 0)
            c = c+1;

    if (c <= 2)
        printf("\nE primo");
    else
        printf("\nNAO e primo");
    getch();
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 107 - SALG-107.c  
-----*/  
  
#include <stdio.h>  
#include <conio.h>  
  
void main(void)  
{  
int  a,b;  
  
    b = 0;  
    a = 0;  
    while (a<7)  
    {  
        if (a<3)  
            if (a<2)  
                a = a+2;  
            else  
                a = a+1;  
        else  
        {  
            b = b+1;  
            if (b==1)  
                b = b+2;  
            else  
                a = a+b;  
        }  
        printf("%d %d", a,b);  
    }  
    getch();  
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 108 - SALG-108.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int a,b;

    a = 0;
    b = 5;
    while (a<10)
    {
        if (b<6)
            while (a<3)
            {
                a = a+1;
                b = b+a;
            }
        else
            a = a+b;
        if (a<=3)
        {
            printf("%d %d", a,b);
            a = a+1;
        }
        else
            do
            {
                printf("\n%d", a);
                a = a+1;
            } while (a<=10);
    }
    getch();
}
```



```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 109 - SALG-109.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int  a,b;

    a = 0;
    b = 0;
    while (a==b)
    {
        do
        {
            if (a<2)
            {
                if (b<2)
                    printf("\n%d", a);
                else
                    a = a+5;
                b = b+2;
            }
            else
            {
                printf("%d %d", a,b);
                if (a>b)
                    b = b+a;
            }
        } while (b<=8);
        printf("\n%d", b);
    }
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 110 - SALG-110.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
int  a,b;

    a = 1;
    b = 2;
    do
    {
        if (a==b)
            if (a<0)
                printf("\n%d", a);
            else
                if (b>0)
                    printf("\n%d", b);
                else
                    printf("\n%d", a);
        else
            do
            {
                a = a+b;
                b = b+a;
                printf("%d %d", a,b);
            } while (a<=7);
        do
        {
            a = 0;
            while (a<15)
            {
                a = a+b+1;
                printf("\n%d",a);
            }
            } while (b<=10);
        b = b+a;
    } while (b<=20);
    printf("\n%d", b);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 111 - SALG-111.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    idade,s,n;
    float  m;

    s = 0;
    n = 0;
    printf("\nInforme uma idade (0 para fim): ");
    scanf("%d",&idade);
    while (idade != 0)
    {
        s = s + idade;
        n = n + 1;
        printf("\nInforme uma idade (0 para fim): ");
        scanf("%d",&idade);
    }
    m = (float) s/n;
    printf("\nMedia das idades: %6.2f ", m);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 112 - SALG-112.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int n,i,r;

    for (n=1;n<=10;n++)
    {
        for (i=1;i<=10;i++)
        {
            r = i * n;
            printf("\n%2d x %2d = %3d", i,n,r);
        }
        printf("\nAperte numa tecla para continuar");
        getch();
    }
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 113 - SALG-113.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int n,i;

    for (n=1;n<=20;n++)
    {
        printf("\n %2d: ", n);
        for (i=1;i<=n;i++)
            if (n%i == 0)
                printf("%d ",i);
    }
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 114 - SALG-114.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int n1,n2,i,c,n;

    printf("\nN1 = "); scanf("%d",&n1);
    printf("\nN2 = "); scanf("%d",&n2);

    for (n=n1;n<=n2;n++)
    {
        c = 0;
        for (i=1;i<=n;i++)
            if (n%i == 0)
                c = c+1;
        if (c<=2)
            printf("\n%d", n);
    }
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 115 - SALG-115.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
    int    a,b,i;
    float  r;

    r = 1;

    printf("\nA = ");    scanf("%d",&a);
    printf("\nB = ");    scanf("%d",&b);

    if (b>0)
        for (i=1;i<=b;i++)
            r = r * a;
    else
        if (b<0)
        {
            b = b * (-1);
            for (i=1;i<=b;i++)
                r = r * a;
            r = 1/r;
        }

    printf("\nResultado: %8.4f", r);
    getch();
}
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 116 - SALG-116.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
    int    n,i;
    float  fat;
    fat = 1;
    do {
        printf("\nEntre com um numero: ");
        scanf("%d",&n);
    } while(n<0);

    for (i=n;i>=1;i--)
        fat = fat * i;
    printf("\nFatorial: %8.0f ", fat);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 117 - SALG-117.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    n,i;
    float  v,maior,menor;

    printf("\nDigite o numero de valores: ");
    scanf("%d",&n);

    for (i=1;i<=n;i++)
    {
        printf("\nValor: ");
        scanf("%f",&v);
        if (i==1)
        {
            maior = v;
            menor = v;
        }
        if (v > maior)
            maior = v;
        if (v < menor)
            menor = v;
    }
    printf("\nMaior: %8.2f", maior);
    printf("\nMenor: %8.2f", menor);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 118 - SALG-118.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float nota,maior,menor;

    maior = 0;
    menor = 10;

    do {
        printf("\nNota: ");
        scanf("%f",&nota);
        if (nota!=99)
        {
            if (nota > maior)
                maior = nota;
            if (nota < menor)
                menor = nota;
        }
    } while (nota != 99);

    printf("\nMaior nota : %4.1f ", maior);
    printf("\nMenor nota : %4.1f ", menor);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 119 - SALG-119.c
-----*/

#include <stdio.h>
#include <conio.h>
#include <string.h>

void main(void)
{
float altura,maior;
int i;
char nome[81],mais_alto[81]; /* reserva 80 caracteres para os nomes */

maior = 0;

for (i=1;i<=10;i++)
{
printf("\nNome: "); scanf("%s",&nome);
printf("\nAltura: "); scanf("%f",&altura);
if (altura > maior)
{
maior = altura;
/*usar funcao especial para copiar strings*/
/*strcpy = string copy */
/*USO: strcpy(destino, origem) */
strcpy(mais_alto,nome);
}
}
printf("\nNome do mais alto : %s", mais_alto);
printf("\nAltura do mais alto: %4.2f", maior);
getch();
```



```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 120 - SALG-120.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float  sm,st,maior,menor,altu,mm,mt;
int    i,cm,codigo;

    sm = 0;
    cm = 0;
    st = 0;
    maior = 0;
    menor = 3;

    for (i=1;i<=10;i++)
    {
        printf("\nAltura: ");
        scanf("%f",&altu);
        printf("\nCodigo (1) masculino   (2) Feminino: ");
        scanf("%d",&codigo);

        if (altu > maior)
            maior = altu;

        if (altu < menor)
            menor = altu;

        if (codigo == 2)
        {
            sm = sm + altu;
            cm = cm + 1;
        }

        st = st + altu;
    }

    mm = sm/cm;
    mt = st/10;

    printf("\nMaior altura da turma.....: %4.2f", maior);
    printf("\nMenor altura da turma.....: %4.2f",menor);
    printf("\nMedia de altura das mulheres: %4.2f",mm);
    printf("\nMedia de altura da turma.....: %4.2f",mt);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 121 - SALG-121.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float s;
int i;

s = 0;

for (i=1;i<=100;i++)
s = s + (float) 1/i;

printf("\nSoma: %9.7f", s);
getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 122 - SALG-122.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float s;
int d,n;

s = 0;
n = 1;

for (d=1;d<=50;d++)
{
s = s + n/d;
n = n + 2;
}

printf("\nSoma: %10.7f", s);
getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 123 - SALG-123.c
-----*/

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

void main(void)
{
    float s;
    int i;

    s = 0;
    for (i=1;i<=10;i++)
        if (i%2 ==1)
            s = s + i/pow(i,2);
        else
            s = s - i/pow(i,2);
    printf("\nSoma: %9.7f", s);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 124 - SALG-124.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    float s;
    int i,n;

    s = 0;
    n = 1000;
    for (i=1;i<=30;i++)
    {
        if (i%2 == 1)
            s = s + n/i;
        else
            s = s - n/i;
        n = n-3;
    }
    printf("\nSoma: %11.7f", s);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 125 - SALG-125.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    mG,mM,mP,nf,cod,codG,codM,codP;
    char   porte;

    mG = 0;
    mM = 0;
    mP = 0;
    printf("\nInforme o codigo: ");
    scanf("%d",&cod);
    while (cod!=0)
    {
        printf("\nInforme o numero de funcionarios: ");
        scanf("%d",&nf);
        printf("\nInforme o porte (G-Grande M-Media P-Pequena): ");
        porte = getche();
        switch porte
        {
            case 'g' : if (nf > mG)
                        {
                            mG = nf;
                            codG = cod;
                        }
                        break;
            case 'm' : if (nf > mM)
                        {
                            mM = nf;
                            codM = cod;
                        }
                        break;
            case 'p' : if (nf > mP)
                        {
                            mP = nf;
                            codP = cod;
                        }
                        break;
        }
        printf("\nInforme o codigo: ");
        scanf("%d",&cod);
    }
    printf("\nEmpresa Grande   -Codigo: %d Funcionarios: %d ", codG, mG);
    printf("\nEmpresa Media     -Codigo: %d Funcionarios: %d ", codM, mM);
    printf("\nEmpresa Pequena  -Codigo: %d Funcionarios: %d ", codP, mP);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 126 - SALG-126.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    n,c,maior,idade;
    float  perc;
    char   sexo,olhos,cab;

    n = 0;
    c = 0;
    maior = 0;
    printf("\nInforme a idade: ");
    scanf("%d",&idade);
    while (idade != -1)
    {
        printf("\nInforme o sexo (M-Masculino F-Feminino): ");
        sexo = getche();
        printf("\nInforme a cor dos olhos (A-Azul V-Verde C-Castanho): ");
        olhos = getche();
        printf("\nInforme a cor dos cabelos (L-Louros C-Castanhos P-Pretos): ");
        cab = getche();

        n = n + 1;
        if (idade > maior)
            maior = idade;
        if ((sexo=='f') && (idade>=18) && (idade<=35) &&
            (olhos=='v') && (cab=='l'))
            c = c+1;
        printf("\nInforme a idade: ");
        scanf("%d",&idade);
    }
    perc = (float) c * 100 / n;
    printf("\nMaior idade: %d", maior);
    printf("\nPercentagem solicitada no exercicio: %5.2f %%", perc);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 127 - SALG-127.c
|-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int    ca,cb,cc,cm,cv,cn,i,n,MAlto,andar;
char   elev,per;
    MAlto = 0;   ca = 0;   cb = 0;   cc = 0;
    cm = 0;   cv = 0;   cn = 0;
    printf("\nInforme o numero de usuarios: ");   scanf("%d",&n);
    for (i=1;i<=n;i++)
    {
        printf("\nInforme o elevador utilizado (A,B,C): ");
        elev = getche();
        printf("\nInforme o andar: ");   scanf("%d",&andar);
        printf("Informe o periodo (M-matutino, V-Vespertino, N-Noturno): ");
        per = getche();
        if (andar > MAlto)
            MAlto = andar;
        switch (elev) {
            case 'a' : ca = ca+1; break;
            case 'b' : cb = cb+1; break;
            case 'c' : cc = cc+1; break;
        }
        switch (per) {
            case 'm' : cm = cm+1; break;
            case 'v' : cv = cv+1; break;
            case 'n' : cn = cn+1; break;
        }
    }
    printf("\nAndar mais alto: %d", MAlto);
    printf("\nElevador mais utilizado: ");
    if ((ca>cb) && (ca>cc))
        printf("A");
    else
        if ((cb>ca) && (cb>cc))
            printf("B");
        else
            printf("C");
    printf("\nPeriodo mais utilizado: ");
    if ((cm>cv) && (cm>cn))
        printf("Matutino");
    else
        if ((cv>cm) && (cv>cn))
            printf("Vespertino");
        else
            printf("Nortuno");
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 128 - SALG-128.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int  s,n,i;

    s = 0;

    printf("\nN= ");
    scanf("%d",&n);

    for (i=1;i<=n;i++)
        if (i%2 == 1)
            s = s + 5;
        else
            s = s + 1;

    printf("\nO %d o. termo da serie e: %d ", n,s);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 129 - SALG-129.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int a,b,c,i;

    a = 0;
    b = 1;
    printf("\n%d %d ", a,b);

    for (i=1;i<=18;i++)
    {
        c = a + b;
        printf("\n%d", c);
        a = b;
        b = c;
    }
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 130 - SALG-130.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int  s,c;

    s = 0;
    c = 0;

    do {
        c = c + 1;
        s = s + c;
    } while (s<=100);

    printf("\nDevemos somar %d valores ", c);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 131 - SALG-131.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int      ndias;
    float    supri;

    ndias = 0;
    printf("\nEntre com o suprimento de comida: ");
    scanf("%f",&supri);

    do {
        supri = supri / 2;
        ndias = ndias + 1;
    } while (supri > 1);

    printf("\nPassarao %d dias.", ndias);
    getch();
}
```



```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 132 - SALG-132.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    a,b,c;
    float  menor,r;

    menor = (1+1-2)/(1*1*1);
    for (a=1;a<=9;a++)
        for (b=1;b<=9;b++)
            for (c=1;c<=9;c++)
                {
                    r = (float) (a+b-2*c)/(a*b*c);
                    if (r<menor)
                        menor = r;
                }

    printf("\nO menor valor e: %8.6f", menor);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 133 - SALG-133.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int  a,b,c,d,i;

    a = 1;
    b = 2;
    c = 3;
    for (i=4;i<=15;i++)
        {
            d = c + 2*b - a;
            a = b;
            b = c;
            c = d;
        }
    printf("\nO 15o da sequencia e: %d", d);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 134 - SALG-134.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    ca,cr;
    float  sa,sr,nota,ma,mr;

    sa = 0;
    ca = 0;
    sr = 0;
    cr = 0;
    ma = 0;
    mr = 0;

    printf("\nNota: ");    scanf("%f",&nota);
    while (nota != 99)
    {
        if (nota >= 7)
        {
            sa = sa + nota;
            ca = ca + 1;
        }
        else
            if (nota >= 3)
            {
                sr = sr + nota;
                cr = cr + 1;
            }
        printf("\nNota: ");    scanf("%f",&nota);
    }

    if (ca!=0) ma = sa/ca;
    if (cr!=0) mr = sr/cr;
    printf("\nMedia dos alunos aprovados.....: %5.2f", ma);
    printf("\nMedia dos alunos em recuperacao: %5.2f", mr);
    getch();
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 135 - SALG-135.c  
-----*/  
  
#include <stdio.h>  
#include <conio.h>  
  
void main(void)  
{  
int s,a,n,i;  
  
s = 0;  
  
printf("\nA= "); scanf("%d",&a);  
  
do {  
printf("\nN= ");  
scanf("%d",&n);  
} while (n<=0);  
  
for (i=1;i<=n;i++)  
{  
s = s + a;  
a = a + 1;  
}  
  
printf("\nSoma: %d", s);  
getch();  
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 136 - SALG-136.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    a,b,i,c,s;
    float m;

    s = 0;
    c = 0;

    do {
        printf("\nA= ");
        scanf("%d",&a);
        printf("\nB= ");
        scanf("%d",&b);
    } while (a==b);

    if (a < b)
    {
        for (i=a;i<=b;i++)
            if (i%2 == 1)
                s = s + i;
        printf("\nSoma dos impares: %d", s);
    }
    else
    {
        for (i=b; i<=a; i++)
            if (i%3 == 0)
            {
                s = s + i;
                c = c + 1;
            }
        m = (float) s/c;
        printf("\nMedia dos multiplos de 3: %.2f", m);
    }
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 137 - SALG-137.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
float  s,peso,maior,m;
int    c;

    s      = 0;
    c      = 0;
    maior  = 0;

    printf("\nPeso: ");
    scanf("%f",&peso);

    while (peso >= 0)
    {
        if (peso > 60)
        {
            s = s + peso;
            c = c + 1;
        }
        else
            if ((peso<60) && (peso>maior))
                maior = peso;
        printf("\nPeso: ");
        scanf("%f",&peso);
    }

    m = s/c;
    printf("\nMedia de peso das pessoas com mais de 60 Kg: %6.2f", m);
    printf("\nO mais pesado entre as pessoas com menos de 60 Kg: %6.2f",
maior);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 138 - SALG-138.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
int  c,s,x,z;

    s      = 0;
    c      = 0;

    printf("\nX= "); scanf("%d",&x);

    do {
        printf("\nZ= ");
        scanf("%d",&z);
    } while (z<x);

    do {
        s = s + x;
        c = c + 1;
        x = x + 1;
    } while (s<=z);

    printf("\nDevemos somar %d numeros", c);
    getch();
}
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 139 - SALG-139.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
int  c;
float a,b;

    c = 0;    a = 100000;    b = 210000;

    do {
        a = a + a*3.1/100;
        b = b + b*1.5/100;
        c = c + 1;
    } while (a<b);

    printf("\nSerao necessarios %d anos ", c);
    getch();
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 140 - SALG-140.c  
-----*/
```

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

```
void main(void)  
{  
    int i;  
    int x[10];  
  
    for (i=0;i<10;i++)  
        x[i] = 30;  
  
    for (i=0;i<10;i++)  
        printf("\nx[ %d ] = %d", i,x[i]);  
    getch();  
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 141 - SALG-141.c  
-----*/
```

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

```
void main(void)  
{  
    int i;  
    int a[10];  
  
    for (i=0;i<10;i++)  
        a[i] = i+1;  
  
    for (i=0;i<10;i++)  
        printf("\na[ %d ] = %d " ,i,a[i]);  
    getch();  
}
```

```

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 142 - SALG-142.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
    int i;
    int b[10];

    for (i=0;i<10;i++)
        if (i%2 == 1)
            b[i] = 1;
        else
            b[i] = 0;

    for (i=0;i<10;i++)
        printf("\nb[ %d ] = %d ", i,b[i]);
    getch();
}
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 143 - SALG-143
-----*/

```

v[1]	v[2]	v[3]	v[4]	v[5]	v[6]	v[7]	v[8]
6	3	6	7	2	6	1	5

```

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 144 - SALG-144.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
    int i;
    int c[10];
    for (i=0;i<10;i++)
    {
        printf("\nc[ %d ] = ", i);
        scanf("%d",&c[i]);
    }
    for (i=0;i<10;i++)
        if (c[i] < 0)
            c[i] = 0;
    printf("\n");
    for (i=0;i<10;i++)
        printf("\nc[ %d ] = %d", i,c[i]);
    getch();
}

```



```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 145 - SALG-145.c  
-----*/  
  
#include <stdio.h>  
#include <conio.h>  
  
void main(void)  
{  
    int i;  
    int d[10],e[10];  
  
    for (i=0;i<10;i++)  
    {  
        printf("\nd[ %d ] = ", i);  
        scanf("%d",&d[i]);  
    }  
  
    for (i=0;i<10;i++)  
        e[i] = d[i];  
  
    printf("\n");  
    for (i=0;i<10;i++)  
        printf("\nd[ %d ] = %d", i,d[i]);  
  
    printf("\n");  
    for (i=0;i<10;i++)  
        printf("\ne[ %d ] = %d", i,e[i]);  
    getch();  
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 146 - SALG-146.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
    int i;
    int x[10];
    float y[10];

    for (i=0;i<10;i++)
    {
        printf("\nx[ %d ] = ", i);
        scanf("%d",&x[i]);
    }

    for (i=0;i<10;i++)
        if (i%2 == 0)
            y[i] = (float) x[i]/2;
        else
            y[i] = (float) x[i] * 3;

    printf("\n");
    for (i=0;i<10;i++)
        printf("\ny[%d] = %5.1f", i,y[i]);
    getch();
}
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 147 - SALG-147.c
-----*/
#include <stdio.h>
#include <conio.h>

void main(void)
{
    int i, z[10],w[10];
    for (i=0;i<10;i++)
    {
        printf("\nz[ %d ] = ", i);
        scanf("%d",&z[i]);
    }
    for (i=0;i<10;i++)
        w[i] = z[9-i];
    printf("\n");
    for (i=0;i<10;i++)
        printf("\nw[ %d ] = %d ", i,w[i]);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 148 - SALG-148.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
    int v,i,c,a[10];

    for (i=0;i<10;i++)
    {
        printf("\na[ %d ] = ", i);
        scanf("%d",&a[i]);
    }

    printf("\nV= ");
    scanf("%d",&v);

    c = 0;
    for (i=0;i<10;i++)
        if (v == a[i])
            c = c + 1;

    printf("\n %d aparece %d vezes no vetor A", v,c);
    getch();
}

/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 149 - SALG-149.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
    int x,i,b[10];

    for (i=0;i<10;i++)
    {
        printf("\nb[ %d ] = ", i);
        scanf("%d",&b[i]);
    }

    printf("\nX= ");
    scanf("%d",&x);

    for (i=0;i<10;i++)
        if (x == b[i])
            printf("\n%d", i);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 150 - SALG-150.c
-----*/

#include <stdio.h>
#include <conio.h>
#include <string.h>

void main(void)
{
    int  achei,i;
    char c[10][81];
    char a[81];

    for (i=0;i<10;i++)
    {
        printf("\nc[%d ] = ", i);
        scanf("%s",&c[i]);
    }

    printf("\nA= ");
    scanf("%s",&a);

    achei = 0;
    for (i=0;i<10;i++)
        if (!strcmp(a,c[i]))
            achei = 1;

    if (achei == 1)
        printf("\nACHEI");
    else
        printf("\nNAO ACHEI");
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 151 - SALG-151.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    i;
    float  nota[10];
    float  s,m;
    int    c;

    s = 0;
    for (i=0;i<10;i++)
    {
        printf("\nnota[ %d ] = ", i);
        scanf("%f",&nota[i]);
        s = s + nota[i];
    }
    m = s / 10;

    c = 0;
    for (i=0;i<10;i++)
        if (nota[i] > m)
            c = c + 1;

    printf("\nMedia: %6.2f", m);
    printf("\nExistem %d notas acima da media",c);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 152 - SALG-152.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int  c,i;
    int  x[10],r[10];

    for (i=0;i<10;i++)
    {
        printf("\nx[ %d ] =", i);
        scanf("%d",&x[i]);
    }

    c = 0;
    for (i=0;i<10;i++)
        if (x[i] < 0)
        {
            r[c] = x[i];
            c = c + 1;
        }

    printf("\n");
    for (i=0;i<c;i++)
        printf("\nR[ %d ] = %d ",i,r[i]);
    getch();
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 153 - SALG-153.c  
-----*/  
  
#include <stdio.h>  
#include <conio.h>  
  
void main(void)  
{  
    int i;  
    int x[15],s[10];  
    int r[5];  
  
    for (i=0;i<5;i++)  
    {  
        printf("\nr[ %d ] =", i);  
        scanf("%d",&r[i]);  
    }  
  
    for (i=0;i<10;i++)  
    {  
        printf("\ns[ %d ] = ", i);  
        scanf("%d",&s[i]);  
    }  
  
    for (i=0;i<5;i++)  
        x[i] = r[i];  
  
    for (i=0;i<10;i++)  
        x[i+5] = s[i];  
  
    printf("\n");  
    for (i=0;i<15;i++)  
        printf("\nx[ %d ] = %d ", i,x[i]);  
    getch();  
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 154 - SALG-154.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int i, maior, posi;
    int q[10];

    for (i=0; i<10; i++)
        do {
            printf("\nq[ %d ] = ", i);
            scanf("%d", &q[i]);
        } while ( q[i] <= 0);

    maior = 0;
    for (i=0; i<10; i++)
        if (q[i] > maior)
        {
            maior = q[i];
            posi = i;
        }

    printf("\n\nO maior valor e: %d", maior);
    printf("\nFoi encontrado na posicao: %d", posi);
    getch();
}
```



```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 155 - SALG-155.c  
-----*/  
  
#include <stdio.h>  
#include <conio.h>  
  
void main(void)  
{  
    int  i,aux,u[10];  
  
    for (i=0;i<10;i++)  
    {  
        printf("\nu[ %d ] = ", i);  
        scanf("%d",&u[i]);  
    }  
  
    for (i=0;i<5;i++)  
    {  
        aux = u[i];  
        u[i] = u[9-i];  
        u[9-i] = aux;  
    }  
  
    printf("\n");  
    for (i=0;i<10;i++)  
        printf("\nu[ %d ] = %d", i,u[i]);  
    getch();  
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 156 - SALG-156.c  
-----*/  
  
#include <stdio.h>  
#include <conio.h>  
  
void main(void)  
{  
    int i,aux;  
    int m[10];  
  
    for (i=0;i<10;i++)  
    {  
        printf("\nm[ %d ] = ", i);  
        scanf("%d",&m[i]);  
    }  
  
    for (i=0;i<5;i++)  
    {  
        aux = m[i];  
        m[i] = m[i+5];  
        m[i+5] = aux;  
    }  
  
    printf("\n");  
    for (i=0;i<10;i++)  
        printf("\nm[ %d ] = %d", i,m[i]);  
    getch();  
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 157 - SALG-157.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    i,j,c;
    int    s[10];
    int    r[5],x[5];

    for (i=0;i<5;i++)
    {
        printf("\nr[ %d ] = ", i);
        scanf("%d",&r[i]);
    }

    for (i=0;i<10;i++)
    {
        printf("\ns[ %d ] = ", i);
        scanf("%d",&s[i]);
    }

    c = 0;
    for (i=0;i<5;i++)
        for (j=0;j<10;j++)
            if (r[i] == s[j])
            {
                x[c] = r[i];
                c = c + 1;
            }

    printf("\n");
    for (i=0;i<c;i++)
        printf("\nx[ %d ] = %d", i,x[i]);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 158 - SALG-158.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    i,j,c;
    int    a[10];
    int    r[5];

    for (i=0;i<5;i++)
    {
        printf("\nr[ %d ] = ", i);
        scanf("%d",&r[i]);
    }

    printf("\n");
    for (i=0;i<10;i++)
    {
        printf("\na[ %d ] = ", i);
        scanf("%d",&a[i]);
    }

    c = 0;
    for (i=0;i<5;i++)
        for (j=0;j<10;j++)
            if (r[i] == a[j])
                c = c + 1;

    printf("\n\nO apostador fez %d pontos.", c);
    getch();
}
```

```
/*-----  
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139  
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.  
| Exercicio 159 - SALG-159.c  
-----*/  
  
#include <stdio.h>  
#include <conio.h>  
  
void main(void)  
{  
    int    i,c;  
    int    d[20];  
  
    for (i=0;i<20;i++)  
    {  
        printf("\nd[ %d ] = ", i);  
        scanf("%d",&d[i]);  
    }  
  
    c = 0;  
    for (i=0;i<20;i++)  
        if (d[i] > 0)  
        {  
            d[c] = d[i];  
            c = c + 1;  
        }  
  
    printf("\n");  
    for (i=0;i<c;i++)  
        printf("\nd[ %d ] = %d", i,d[i]);  
    getch();  
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 160 - SALG-160.c
|-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int    i,c;
    int    p[10];
    float  n[10];
    float  s,m;

    s=0;
    c=0;
    for (i=0;i<10;i++)
    {
        printf("\nn[ %d ] = ", i);
        scanf("%f",&n[i]);
        if (n[i] >= 6)
        {
            s = s + n[i];
            c = c + 1;
        }
    }
    m = s/c;

    c = 0;
    for (i=0;i<10;i++)
        if (n[i] > m)
        {
            p[c] = i;
            c = c + 1;
        }

    printf("\n\nMedia: %5.2f", m);
    printf("\n");
    for (i=0;i<c;i++)
        printf("\np[ %d ] = %d", i,p[i]);
    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 161 - SALG-161.c
-----*/

#include <stdio.h>
#include <conio.h>

void main(void)
{
    int i,c,d;
    int x[10],y[10];
    int r[20];

    c=0;
    d=0;
    for (i=0;i<10;i++)
    {
        printf("\nx[ %d ] = ", i);
        scanf("%d",&x[i]);
    }

    printf("\n");
    for (i=0;i<10;i++)
    {
        printf("\ny[ %d ] = ", i);
        scanf("%d",&y[i]);
    }

    for (i=0;i<20;i++)
        if (i%2 == 1)
        {
            r[i] = x[c];
            c = c + 1;
        }
        else
        {
            r[i] = y[d];
            d = d + 1;
        }

    printf("\n");
    for (i=0;i<20;i++)
        printf("\nr[ %d ] = %d", i,r[i]);

    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 162 - SALG-162.c
|-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int M[5][5], i, j, soma;
for(i=0;i<=4;i++)
    for(j=0;j<=4;j++)
    {
        printf("Digite o elemento M[%d][%d]:",i,j);
        scanf("%d",&M[i][j]);
    }
soma = 0;

for(i=0;i<=4;i++)
{
    soma = soma + M[3][i]; /*considerando a linha 1 com indice 0 */
}
printf("\na) Soma dos elementos da linha 4: %d",soma);

soma = 0;
for(i=0;i<=4;i++)
{
    soma = soma + M[i][1]; /*considerando a coluna 1 com indice 0 */
}
printf("\nb) Soma dos elementos da coluna 2: %d",soma);

soma = 0;
for(i=0;i<=4;i++)
{
    soma = soma + M[i][i]; /*considerando a linha 1 com indice 0 */
}
printf("\nc) Soma dos elementos da diagonal principal: %d",soma);

soma = 0;
for(i=0;i<=4;i++)
{
    soma = soma + M[i][4-i]; /*considerando a linha 1 com indice 0 */
}
printf("\nd) Soma dos elementos da diagonal secundaria: %d",soma);

soma = 0;
for(i=0;i<=4;i++)
    for(j=0;j<=4;j++)
    {
        soma = soma + M[i][j]; /*considerando a linha 1 com indice 0 */
    }
printf("\ne) Soma de todos os elementos da matriz: %d",soma);
getch();
}
```



```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 163 - SALG-163.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
    int A[4][4], i, j, soma;

    for(i=0;i<=3;i++)
        for(j=0;j<=3;j++)
        {
            printf("Digite o elemento A[%d][%d]:",i,j);
            scanf("%d",&A[i][j]);
        }

    soma = 0;
    for(i=0;i<=1;i++)
        for(j=0;j<=1;j++)
        {
            soma = soma + A[i][j]; /*considerando a linha 1 com indice 0 */
        }
    printf("\na) Soma elementos das duas primeiras linhas e colunas: %d",soma);

    soma = 0;
    for(i=2;i<=3;i++)
        for(j=2;j<=3;j++)
        {
            soma = soma + A[i][j]; /*considerando a linha 1 com indice 0 */
        }
    printf("\nb) Soma elementos das ultimas duas linhas e colunas: %d",soma);

    soma = 0;
    for(i=1;i<=3;i++)
        for(j=0;j<=i-1;j++)
        {
            soma = soma + A[i][j]; /*considerando a linha 1 com indice 0 */
        }
    printf("\nc) Soma elementos matriz triangular inferior: %d",soma);

    soma = 0;
    for(i=0;i<=2;i++)
        for(j=i+1;j<=3;j++)
        {
            soma = soma + A[i][j]; /*considerando a linha 1 com indice 0 */
        }
    printf("\nd) Soma elementos da matriz triangular superior: %d",soma);

    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 164 - SALG-164.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
    int A[4][6], B[4][6], i, j, S[4][6], D[4][6];

    for(i=0;i<=3;i++)
        for(j=0;j<=5;j++)
        {
            printf("Digite o elemento A[%d][%d]:",i,j);
            scanf("%d",&A[i][j]);
        }

    for(i=0;i<=3;i++)
        for(j=0;j<=5;j++)
        {
            printf("Digite o elemento B[%d][%d]:",i,j);
            scanf("%d",&B[i][j]);
        }

    for(i=0;i<=3;i++)
        for(j=0;j<=5;j++)
        {
            S[i][j] = A[i][j] + B[i][j];
            D[i][j] = A[i][j] - B[i][j];
        }

    for(i=0;i<=3;i++)
    {
        for(j=0;j<=5;j++)
        {
            printf("S[%d][%d]=%3d ",i,j,S[i][j]);
        }
        printf("\n");
    }

    printf("\n");

    for(i=0;i<=3;i++)
    {
        for(j=0;j<=5;j++)
        {
            printf("D[%d][%d]=%3d ",i,j,D[i][j]);
        }
        printf("\n");
    }

    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 165 - SALG-165.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
    int G[5][5], SL[5], SC[5], i, j;

    for(i=0;i<=4;i++)
    {
        for(j=0;j<=4;j++)
        {
            printf("Digite o elemento G[%d][%d]:",i,j);
            scanf("%d",&G[i][j]);
        }
        SL[i] = 0;
        SC[i] = 0;
    }

    for(i=0;i<=4;i++)
        for(j=0;j<=4;j++)
        {
            SL[i] = SL[i] + G[i][j];
            SC[i] = SC[i] + G[j][i];
        }

    printf("\n");
    for(i=0;i<=4;i++)
    {
        printf("SL[%d]=%3d ",i,SL[i]);
    }
    printf("\n");

    printf("\n");
    for(i=0;i<=4;i++)
    {
        printf("SC[%d]=%3d ",i,SC[i]);
    }

    getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 166 - SALG-166.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
int D[5][5], i, j, X, existe;

for(i=0;i<=4;i++)
    for(j=0;j<=4;j++)
    {
        printf("Digite o elemento D[%d][%d]:",i,j);
        scanf("%d",&D[i][j]);
    }

printf("Digite um valor X: ");
scanf("%d",&X);

existe = 0;
for(i=0;i<=4;i++)
    for(j=0;j<=4;j++)
    {
        if (D[i][j]==X)
            existe = 1;
    }

if (existe==1)
    printf("O valor de X existe na matriz.");
else
    printf("O valor de X nao existe na matriz.");

getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 167 - SALG-167.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
float A[12][13];
int i, j, maior;

for(i=0;i<=11;i++)
    for(j=0;j<=12;j++)
    {
        printf("Digite o elemento A[%d][%d]:",i,j);
        scanf("%f",&A[i][j]);
    }

for(i=0;i<=11;i++)
{
    maior = A[i][0];
    for(j=1;j<=12;j++)
    {
        if (maior < A[i][j])
        {
            maior = A[i][j];
        }
    }
    printf("A[linha %d]/Maior: ",i);
    for(j=0;j<=12;j++)
    {
        printf("%6.2f ",A[i][j]/maior);
    }
    printf("\n");
}

getch();
}
```

```
/*-----
| UCPEL - Escola de Informatica - Algoritmos e Programacao - 040139
| Solucao em C - Engenharia Computacional I - Escola de Engenharia e Arq.
| Exercicio 168 - SALG-168.c
-----*/
#include <stdio.h>
#include <conio.h>
void main(void)
{
    int G[13], A[13][3];
    int i, j, acerto, simples, duplo, triplo;

    for(i=0;i<=12;i++)
    {
        printf("Digite o vetor gabarito G[%d]:",i);
        scanf("%d",&G[i]);
    }

    for(i=0;i<=12;i++)
        for(j=0;j<=2;j++)
        {
            printf("Digite a matriz de apostas A[%d][%d]:",i,j);
            scanf("%d",&A[i][j]);
        }

    acerto = simples = duplo = triplo = 0;
    for(i=0;i<=12;i++)
    {
        if (A[i][G[i]-1]==1) acerto++;
        switch(A[i][1] + A[i][2] + A[i][3])
        {
            case 1: simples++;
                     break;
            case 2: duplo++;
                     break;
            case 3: triplo++;
                     break;
        }
    }

    printf("Existem %d acertos \n",acerto);
    printf("Foram realizadas %d apostas simples \n",simples);
    printf("Foram realizadas %d apostas duplas \n",duplo);
    printf("Foram realizadas %d apostas triplas \n",triplo);

    getch();
}
```

Exercícios Resolvidos - Primeira parte da Matéria

1) Escrever um programa que encontra e escreve na tela os n primeiros números perfeitos. O número n é determinado pelo usuário. Um número perfeito é aquele que é igual a soma de seus divisores (exemplo: “ $6 = 1 + 2 + 3$ ” e “ $28 = 1 + 2 + 4 + 7 + 14$ ”).

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

/* O algoritmo apresentado abaixo nao esta' otimizado. Voce pode melhora-lo? */
/* Para encontrar os tres primeiros numeros perfeitos a busca pode ser rapida */
/* para os demais numeros a busca pode ser bem demorada. */
/* Se o programa estiver demorando muito use CTRL + C para abortar */

void main(void)
{
    long int numero, cont, N, soma, i;

    printf("Quantos numeros perfeitos: ");
    scanf("%d", &N);

    numero = 1;    /* numero em teste */
    cont = 0;      /* quantidade de numeros perfeitos encontrados */

    while (cont!=N)
    {
        /* inicializa um somador para cada numero em teste */
        soma = 0;

        /* encontra todos os divisores do numero */
        /* e soma cada divisor */
        for (i=1; i<numero; i++)
        {
            if ((numero%i)==0)
                soma = soma + i;
        }

        /* se a soma dos divisores e' igual ao numero */
        /* imprime o numero perfeito e */
        /* atualiza a quantidade de numeros perfeitos encontrados */
        if (soma==numero)
        {
            printf("Numero %d\n", numero);
            cont++;
        }

        /* passa para o proximo numero a ser testado */
        numero++;
    }

    getch();
}
```

2) Escreva um programa em C para calcular uma raiz real de uma equação do terceiro grau. O programa deve ler os 4 termos da equação, o intervalo de valores onde está a raiz, e uma constante associada ao erro na determinação da raiz. Use o método de bisecção (Descobrir o sinal da função em cada extremo e no ponto central do intervalo definido para a raiz. Considerar como novo intervalo os dois pontos mais próximos e com sinais diferentes para a função. Repetir até que o valor da função, em módulo, seja menor que o erro admitido.).

```
#include <math.h>
#include <stdio.h>
#include <conio.h>
void main(void)
{
    float A,B,C,D, i1, i2, erro;
    float meio, smeio, sl;

    /* ler os coeficientes da equacao */
    printf("X^0: "); scanf("%f", &A);
    printf("X^1: "); scanf("%f", &B);
    printf("X^2: "); scanf("%f", &C);
    printf("X^3: "); scanf("%f", &D);

    /* ler o intervalo onde esta a raiz e o erro aceitavel para a raiz */
    printf("Intervalo da esquerda: "); scanf("%f", &i1);
    printf("Intervalo da direita : "); scanf("%f", &i2);
    printf("Erro maximo aceitavel: "); scanf("%f", &erro);

    /* calcula a equacao nos limites do intervalo */
    do {

        /* calcula o ponto central do intervalo e o valor da funcao neste ponto */
        meio = (i2+i1)/2.0;
        smeio = A + B*meio + C*pow(meio,2) + D*pow(meio,3);
        sl = A + B*i1 + C*pow(i1,2) + D*pow(i1,3);

        /* se a funcao tem o mesmo sinal de i1 entao o novo intervalo e' formado*/
        /* pelo valor central e por i2, senao, o novo intervalo e' formado */
        /* por i1 e o ponto central */
        if (((smeio>0) && (sl>0)) || ((smeio<0)&&(sl<0)))
        {
            i1 = meio;
        }
        else
        {
            i2 = meio;
        }
    } while ((i2-i1)>fabs(erro));

    /*termina quando o valor da funcao e' menor que o erro aceitavel para a raiz*/
    /* imprime o resultado */
    printf("Uma raiz real da equacao e' %f ", meio);
    getch();
}
```


3) A média elíptica (ou aritmético-geométrica) de dois números positivos a e b [$a < b$], é calculada do seguinte modo: Chamando $a_{n+1} = \sqrt{a_n b_n}$ e $b_{n+1} = (a_n + b_n) / 2$ respectivamente as médias geométrica e aritmética desses números obtemos uma seqüência de números a_0, a_1, a_2, \dots e b_0, b_1, b_2, \dots tal que $a_0 < a_1 < a_2 < \dots < b_2 < b_1 < b_0$. O limite desta seqüência é $m = a_\infty = b_\infty$. Por exemplo: a média elíptica de 4 e 7 é 5.5932... Escrever ma função *double elip(double a, double b)* que calcula a média elíptica entre a e b .

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

double elip(double A, double B);

/* programa para teste */
void main(void)
{
    double A, B;

    /* le dois numeros para o calculo da media */
    printf("Primeiro numero: ");
    scanf("%lf", &A);
    printf("Segundo numero: ");
    scanf("%lf", &B);

    /* chama a funcao e imprime o resultado */
    printf("Media eliptica: %lf ", elip(A,B));

    getch();
}

/* resposta do problema */

double elip(double A, double B)
{
    double A1, B1;

    /* OBSERVACOES PARA O ALGORITMO ABAIXO */
    /* se o laço "while" estiver demorando muito para terminar, isto pode estar */
    /* acontecendo por problemas numericos (erros de arredondamento nas operacoes */
    /* com "double") mas isto pode ser minimizado substituindo "while (A<B)" por */
    /* "while ((B-A)<0.000001)" onde 0.000001 e' o erro admitido no calculo */

    while (A<B)
    {
        A1 = pow(A*B,0.5);
        B1 = (A+B)/2;
        A = A1;
        B = B1;
    }
    return A1;
}
```

4) Escrever um algoritmo e o programa em C para simular uma calculadora com as 4 operações, raiz quadrada e percentual. O programa só termina se for selecionada uma opção para fim.

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

void main (void)
{
    float Numerol, Numero2;
    char operacao;

    do {
        printf("\nPrimeiro Numero: ");
        scanf("%f", &Numerol);
        printf("Escolha uma operacao\n");
        printf("* / + - (P)Percentual (Z)Raiz (F)Fim: ");
        operacao = getche();
        switch (operacao) {
            case '*':
                printf("\nSegundo Numero: ");
                scanf("%f", &Numero2);
                printf("Resultado: %f", Numerol*Numero2);
                break;

            case '/':
                printf("\nSegundo Numero: ");
                scanf("%f", &Numero2);
                printf("Resultado: %f", Numerol/Numero2);
                break;

            case '+':
                printf("\nSegundo Numero: ");
                scanf("%f", &Numero2);
                printf("Resultado: %f", Numerol+Numero2);
                break;

            case '-':
                printf("\nSegundo Numero: ");
                scanf("%f", &Numero2);
                printf("Resultado: %f", Numerol-Numero2);
                break;

            case 'P':
            case 'p':
                printf("\nSegundo Numero: ");
                scanf("%f", &Numero2);
                printf("Resultado: %f", Numerol*Numero2/100);
                break;

            case 'R':
            case 'r':
                printf("\nResultado: %f", sqrt(Numerol));
                break;

        }
    } while (operacao!='F' && operacao!='f');
}
```

5) Escrever um algoritmo e o programa em C para calcular a soma dos N (definido pelo usuário) primeiros termos da série: $\left[1 - \frac{3}{4} + \frac{5}{9} - \frac{7}{16} + \dots\right]$.

```
#include <stdio.h>
#include <conio.h>
#include <math.h>
void main(void)
{
    int N, i;
    float total;

    /* ler a quantidade de parcelas da serie */
    printf("Qual o numero de parcelas? ");
    scanf("%d", &N);

    /* inicializa o totalizador da soma */
    total = 0;

    /* calcula cada um dos N termos da serie e soma ao valor acumulado anterior */
    for(i=1;i<=N;i++)
    {
        total = total + (pow(-1,i-1)*(2.0*i-1)/(i*i));
    }

    /* imprime o resultado */
    printf("A soma das %d primeiras parcelas da serie e' %f", N, total);
    getch();
}
```

6) Qual é o resultado do programa ?

```
#include <stdio.h>

void main ( void )
{
    int k, y;
    float a, b;

    k = 6;
    y = k * k + 6.5 / 2 - 6;

    a = (float) y / 2;
    b = y + 3.854;

    printf ( "a = %5.2f b = %7.4f\n",  a, b );
}
```

7) O resultado esperado na tela deveria ser 47.325193 mas isso não acontece. Determine o erro e explique como corrigir o programa.

```
#include <stdio.h>

int soma(float, float);

void main(void)
{
    float a, b;
    a = 28.232323;
    b = 19.09287;
    printf( "%f",  soma(a,b) );
}

int  soma(float x, float y)
{
    return x + y;
}
```

8) Para o programa abaixo, diga o que será impresso na tela após a sua execução.

```
#include <stdlib.h>
#include <conio.h>
#include <stdio.h>
void a(void);
void b(void);
void c(void);
int x = 1;
int main()
{
    int x = 5;
    printf("x eh %d\n",x);
    {
        int x = 7;
        printf("x eh %d\n",x);
    }
    printf("x eh %d\n",x);
    a();
    b();
    c();
    a();
    b();
    c();
    printf("x eh %d\n",x);
    getch();
    return 0;
}

void a(void)
{
    int x = 25;
    printf("\nx eh %d\n",x);
    ++x;
    printf("x eh %d\n",x);
}

void b(void)
{
    static int x = 50;
    printf("\nx eh %d\n", x);
    ++x;
    printf("x eh %d\n",x);
}

void c(void)
{
    printf("\nx eh %d\n",x);
    x *= 10;
    printf("x eh %d\n",x);
}
```

9) Escrever os algoritmos e as funções em linguagem C para avaliar o valor da expressão $y = e^x$, onde x é um número inteiro positivo. Para tanto, utilize a seguinte aproximação:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \dots$$

Os parâmetros de entrada da função principal são o valor de “ x ” e o número de termos da série (inteiro maior ou igual a 1). A função deve retornar um número no formato de ponto flutuante para o programa principal e fazer uso apenas de variáveis locais. Se o valor de “ x ” ou o número de termos da série estiverem fora da faixa permitida a função deve retornar um número menor que zero. Os fatoriais devem ser calculados em uma função separada cuja definição deve ser “long int fatorial (int *numero*)”.

```
#include <math.h>
#include <conio.h>
#include <stdio.h>
double exp(int x, int N);
long int fatorial(int x);
/***** PROGRAMA PARA TESTE *****/
void main(void)
{
    int x, N;

    printf("\nX=");
    scanf("%d", &x);
    printf("\nN=");
    scanf("%d", &N);
    printf("\nExp(x,N)=%f", exp(x,N));
    printf("\n\nDigite algo para retornar ao windows.");
    getch();
}
/***** RESPOSTA DO PROBLEMA *****/
double exp(int x, int N)
{
    double expo;
    int i;
    if (x<0 || N<1)
        return -1;
    expo = 1;
    for(i=2;i<N;i++)
        expo = expo + pow(x,i-1)/fatorial(i-1);
    return expo;
}

long int fatorial(int x)
{
    int i;
    long int resultado;
    resultado = 1;
    for (i=1;i<=x;i++)
        resultado = resultado * i;
    return resultado;
}
```

10) Deseja-se que um sistema microcontrolado seja capaz de calcular a raiz quadrada de um número. Como algoritmo, optou-se pelo uso do método das secantes para encontrar a raiz da equação $f(x)=x^2-A=0$. A raiz da equação $f(x)$ pode ser obtida pela seguinte fórmula:

$$x_{k+1} = \frac{x_{k-1} \cdot f(x_k) - x_k \cdot f(x_{k-1})}{f(x_k) - f(x_{k-1})}$$

Utilize como estimativas iniciais: $x_{k-1}=0$ e $x_k=1$. Não é necessário escrever o programa principal. Escreva apenas os módulos (funções) que calculam a raiz e a $f(x)$. A função que calcula a raiz deve receber “A” e “Erro” como parâmetros de entrada e devolver a raiz como saída. Se não existir uma raiz real a função devolve -1. Utilize apenas variáveis locais. A função matemática $f(x)$ deve ser implementada como um módulo aparte. O parâmetro “Erro” corresponde ao erro máximo aceito para X e para Y.

```
#include <stdio.h>
#include <conio.h>
#include <math.h>
float Raiz(float A, float Erro);
float f(float x, float A);

/***** Chamada principal para exemplo *****/
void main(void)
{
    printf("X = %f", Raiz(2,0.0000001));
    getch();
}

/*****Resposta do problema *****/
float Raiz(float A, float Erro)
{
    float x, xk, xk1;
    xk = 1;
    xk1 = 0;

    if (A<0) return(-1);
    else {
        while ( (fabs(xk-xk1)>Erro) || (fabs(f(xk,A))>Erro) )
        {
            x = (xk1*f(xk,A)-xk*f(xk1,A)) / (f(xk,A)-f(xk1,A));
            xk1 = xk;
            xk = x;
        }
        return(x);
    }
}

float f(float x, float A)
{
    return(x*x-A);
}
```

11) Deseja-se o algoritmo e o programa em linguagem C para calcular as soluções de equações de primeiro e segundo grau. Utilize módulos e um menu de opções conforme mostrado abaixo.

- (1) Digitar coeficientes da equação
- (2) Ver equação atual
- (3) Mostrar resultado
- (4) Terminar

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

float a, b, c;

void DigitaCoef();
void VerEq();
void Mostra();

/*****/
void main(void)
{
    char opcao;

    do {
        clrscr();
        printf("*****");
        printf("\n(1) Digitar coeficientes da equacao");
        printf("\n(2) Ver equacao atual");
        printf("\n(3) Mostrar resultado");
        printf("\n(4) Terminar");
        printf("\n*****");
        opcao = getche();
        switch (opcao)
        {
            case '1': DigitaCoef(); break;
            case '2': VerEq(); break;
            case '3': Mostra(); break;
        }
    } while (opcao!='4');
}

/*****/
void DigitaCoef()
{
    clrscr();
    printf("Digite o valor do coeficiente A: ");
    scanf("%f", &a);
    printf("Digite o valor do coeficiente B: ");
    scanf("%f", &b);
    printf("Digite o valor do coeficiente C: ");
    scanf("%f", &c);
}
```



```

/*****
void VerEq()
{
    char ch;
    clrscr();
    printf("*** Equacao Atual ***");
    if (a==0) printf("\nEquacao do primeiro grau");
    else printf("\nEquacao do segundo grau");
    printf("\nEquacao: %f * x^2 + %f * x + %f = 0",a,b,c);
    printf("\nPressione uma tecla para continuar");
    getch();
}

/*****
void Mostra()
{
    float aux;
    char ch;
    clrscr();
    printf("*** Equacao Atual ***");
    if (a==0) printf("\nEquacao com uma raiz");
    else printf("\nEquacao com duas raizes");
    if (a==0) printf("\nX = %f",-c/b);
    else
    {
        aux = b*b - 4*a*c;
        if (aux>=0)
            printf("\nX1=%f, X2=%f",(-b+sqrt(aux))/(2*a),(-b-sqrt(aux))/(2*a));
        else printf("\nX12=%f +- %f i",-b/(2*a),sqrt(-aux)/(2*a));
    }
    printf("\nPressione uma tecla para continuar");
    getch();
}

```

12) Escrever um programa que leia um número entre 2 e 20 e escreva na tela a seguinte sequência:

Primeira linha: 1 2 3 ... N

Segunda linha: 2 3 ... (N-1)

...

*Antes de terminar o programa apresentar a mensagem: "Digite 'T' para terminar". Terminar o programa apenas quando a letra 'T' for pressionada.

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

void main(void)
{
    char ch;
    int  n, x, y;

    do {
        clrscr();
        printf("Escolha um numero inteiro entre 2 e 20: ");
        scanf("%d", &n);
    } while( (n<2) && (n>20) );
    for(x=1; x<=n/2; x++)
    {
        for (y=x; y<=(n-x+1); y++)
            printf("  %d ", y);
        printf("\n");
    }
    printf("Digite 'T' para terminar");
    do ch= getch(); while ( (ch!='T') && (ch!='t') );
}
```

13) Um sistema de aquisição de sinais está sendo utilizado para medir capacitância. O sistema injeta uma corrente constante sobre o capacitor e monitora a tensão sobre ele. Durante o período em que o capacitor está sendo carregado são realizadas 5 medidas de corrente e tensão, nos instantes de tempo $t=0$, $t=0,2$, $t=0,4$, $t=0,6$ e $t=0,8$. Nestas condições $v_C(t) = v_C(0) + \frac{1}{C} \cdot i(t) \cdot t$. Utilize o método de mínimos quadrados, cujas fórmulas são apresentadas abaixo, para determinar o valor do capacitor e sua tensão inicial.

$$y(x) = a + b \cdot x$$

$$a = \frac{\sum_i^n (y_i)}{n} - \frac{b \cdot \sum_i^n (x_i)}{n} \quad b = \frac{\sum_i^n (x_i \cdot y_i) - \frac{1}{n} \cdot \sum_i^n (y_i) \cdot \sum_i^n (x_i)}{\sum_i^n (x_i^2) - \frac{1}{n} \cdot \left(\sum_i^n (x_i) \right)^2}$$

Os dados referentes a cada medida devem ser fornecidos manualmente. Ao final, os valores do capacitor e de sua tensão inicial são apresentadas na tela. Depois de realizar a primeira estimativa o programa deve perguntar ao usuário se ele deseja realizar outra e, em caso afirmativo, repetir todo o procedimento.

```
#include <conio.h>
#include <stdio.h>
void main(void)
{
    float Sx, Sy, Sxx, Sxy, a, b;
    float v, i, Cap;
    float t;
    char ch;
    do {
        Sx = 0; Sy = 0; Sxx = 0; Sxy = 0;
        for(t=0; t<=0.8; t=t+0.2)
        {
            printf("\n Qual a medida de tensao em t=%f ? " ,t);
            scanf("%f",&v);
            printf("\n Qual a medida de corrente em t=%f ? " ,t);
            scanf("%f",&i);
            printf("\n");
            Sx = Sx + i * t;
            Sy = Sy + v;
            Sxx = Sxx + v*v;
            Sxy = Sxy + i * t * v;
        }
        b = (Sxy-0.2 * Sy * Sx) / (Sxx- 0.2 * Sx * Sx);
        a = 0.2 * Sy-0.2 * b * Sx;
        Cap = 1/b;
        printf("\n A tensao inicial no capacitor e: %f", a);
        printf("\n O valor do capacitor e: %f", Cap);
        printf("\n \n \n Voce deseja fazer outra estimativa? (s/n) ");
        ch = getch();
    } while ( (ch=='S') || (ch=='s') );
}
```

14) A função $f(x) = 0,9887 \cdot x - 0,1450 \cdot x^3$ é uma boa aproximação para $\sin(x)$ com $0 \leq x \leq \pi/2$. Escreva uma função que devolva o valor de $\sin(x)$, utilizando a aproximação proposta, para qualquer valor de x .

```
/* ***** PROGRAMA TESTE ***** */
void main(void)
{
    float x;

    for(x=0;x<=2*M_PI;x=x+0.3)
        printf("x=%f,sen(x)=%f,sin(x)=%f,erro=%f\n",x,sen(x),sin(x),sen(x)-sin(x));
    getch();
}

/* ***** SOLUCAO DO PROBLEMA ***** */
float sen(float x)
{
    float ang;
    int    quadrante,sinal;

    sinal = 1;

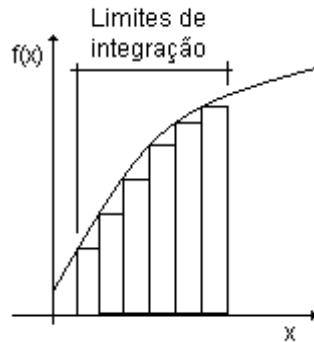
    /* faz a reducao do angulo ao primeiro quadrante */
    ang = (int) (x/(2*M_PI));
    ang = x-ang*2*M_PI;

    /* determina o quadrante correto */
    quadrante = (int)ang/(M_PI/2);

    /* determina o sinal apropriado */
    switch (abs(quadrante)) {
        case 0: if (x<0) {sinal=-1; ang = -ang; }
                break;
        case 1: if (x>0) {sinal = 1; ang = M_PI-ang; }
                else    {sinal =-1; ang = M_PI+ang; }
                break;
        case 2: if (x>0) {sinal =-1; ang = ang-M_PI; }
                else    {sinal = 1; ang = -ang-M_PI;}
                break;
        case 3: if (x>0) {sinal=-1; ang = 2*M_PI-ang;}
                else    {sinal= 1; ang = 2*M_PI+ang;}
    }

    return (sinal*(0.9887*ang-0.1450*ang*ang*ang));
}
```

15) Implementar uma função que calcule a integral de um polinômio de segundo grau. A integral pode ser realizada calculando área de pequenos retângulos cuja amplitude corresponde ao valor da função e a base corresponde a um tamanho conhecido. A função deve receber os intervalos inferior e superior da integral e os três coeficientes do polinômio. Os limites superior e inferior devem ser utilizados para o cálculo e a base dos retângulos não pode ser maior do que 0,01.



```
/* ***** PROGRAMA TESTE ***** */
void main(void)
{
    printf("\nintegral = %f \n", integral(1,2,1,2,3));
    getch();
}

/* ***** SOLUCAO DO PROBLEMA ***** */
float integral(float xi, float xf, float a, float b, float c)
{
    float i, passo, integral;
    int iteracoes;

    integral=0;
    iteracoes = (int) ((xf-xi)/0.01) + 1;
    passo = (xf-xi)/iteracoes;

    /* O uso de variaveis reais no laço para não é muito recomendado */
    /* mas esta é a solução mais curta. O melhor é utilizar uma variável */
    /* inteira para o controle do laço e multiplica-la pelo passo. */

    for(i=xi;i<=xf;i=i+passo)
        integral = integral + (a + b*i + c*i*i);
    return (passo*integral);
}
```

Exercícios Resolvidos - Segunda parte da Matéria

1) Escrever um algoritmo e um programa em linguagem C que leia e multiplique um vetor coluna de dimensão (Nx1) por um vetor linha de dimensão (1xP). As dimensões máximas são 30x1 e 1x30. N e P são conhecidos em tempo de execução. A multiplicação de vetores deste tipo deve ser realizada de acordo com o exemplo abaixo. Refazer o exercício utilizando ponteiros e alocação dinâmica de memória.

$$\begin{bmatrix} 10 \\ 11 \end{bmatrix} \cdot \begin{bmatrix} 13 & 14 \end{bmatrix} = \begin{bmatrix} 10 \cdot 13 & 10 \cdot 14 \\ 11 \cdot 13 & 11 \cdot 14 \end{bmatrix}$$

```
/* Este programa foi escrito para resolver o produto de um vetor coluna
   por um vetor linha. */
```

```
/* ALGORITMO SIMPLIFICADO
-inicio
| ler o numero de colunas (P)
| ler o numero de linhas (N)
|
| repetir da primeira linha ate' a ultima
|   -inicio
|   | ler elemento do vetor
|   -fim
|
| repetir da primeira coluna ate' a ultima
|   -inicio
|   | ler elemento do vetor
|   -fim
|
| repetir da primeira linha ate' a ultima
|   -inicio
|   | repetir da primeira coluna ate' a ultima
|   |   -inicio
|   |   | Matriz[linha][coluna] = vetor[linha]*vetor[coluna]
|   |   -fim
|   -fim
|
-fim
/*
```

```
/* SOLUCAO UTILIZANDO NOTACAO DE VETORES E MATRIZES */

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

void main(void)
{
    float VetorLinha[30], VetorColuna[30], MatrizResultado[30][30];
    int N, P, linha, coluna;

    do {
        printf("Qual a dimensao P? ");
        scanf("%d",&P);
    } while ( (P>30) || (P<0));

    do {
        printf("Qual a dimensao N? ");
        scanf("%d",&N);
    } while ( (N>30) || (N<0));

    for(linha=0;linha<N;linha++)
    {
        printf("Elemento %d: ", linha);
        scanf("%f", &VetorLinha[linha]);    /* uso com vetores */
    }

    for(coluna=0;coluna<P;coluna++)
    {
        printf("Elemento %d: ", coluna);
        scanf("%f", &VetorColuna[coluna]);    /* uso com vetores */
    }

    for (linha=0;linha<N;linha++)
    {
        for (coluna=0;coluna<P;coluna++)
        {
            MatrizResultado[linha][coluna] = VetorLinha[linha] *
                                                VetorColuna[coluna];
        }
    }

    for (linha=0;linha<N;linha++)
    {
        for (coluna=0;coluna<P;coluna++)
        {
            printf("%f  ", MatrizResultado[linha][coluna]);    /* matriz */
        }
        printf("\n");
    }

    getch();
}
```

```
/* SOLUCAO UTILIZANDO NOTACAO DE PONTEIROS E ALOCACAO DINAMICA DE MEMORIA */
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
void main(void)
{
    float *VetorLinha, *VetorColuna, *MatrizResultado;
    int N, P, linha, coluna;

    do {
        printf("Qual a dimensao P? ");
        scanf("%d", &P);
    } while ( (P>30) || (P<0));

    VetorLinha = (float *)calloc(P, sizeof(float)); /* alocação dinamica */

    do {
        printf("Qual a dimensao N? ");
        scanf("%d", &N);
    } while ( (N>30) || (N<0));

    VetorColuna = (float *)calloc(N, sizeof(float)); /* alocação dinamica */
    MatrizResultado = (float *)calloc(N*P, sizeof(float)); /* alocação dinamica */

    /* teste de alocação dinamica */
    if (!MatrizResultado || !VetorLinha || !VetorColuna)
    {
        printf("Erro de alocação de memoria. Programa terminado");
        getch();
        exit;
    }

    for(linha=0;linha<N;linha++) {
        printf("Elemento %d: ", linha);
        scanf("%f", (VetorLinha + linha)); /* uso com ponteiros */
    }

    for(coluna=0;coluna<P;coluna++) {
        printf("Elemento %d: ", coluna);
        scanf("%f", (VetorColuna + coluna)); /* uso com ponteiros */
    }

    for (linha=0;linha<N;linha++) {
        for (coluna=0;coluna<P;coluna++) {
            /* produto utilizando ponteiros */
            *(MatrizResultado + linha*P +coluna) = *(VetorLinha+linha) *
                                                    *(VetorColuna+coluna);
        }
    }

    for (linha=0;linha<N;linha++) {
        for (coluna=0;coluna<P;coluna++) {
            printf("%f ", *(MatrizResultado + linha*P +coluna));/* ponteiros */
        }
        printf("\n");
    }
    getch();
    free(VetorLinha);
    free(VetorColuna);
    free(MatrizResultado);
}
```


2) Escrever um algoritmo e um programa em linguagem C que leia e ordene um vetor de estruturas contendo N valores reais e N nomes (N máximo é 1000. Os nomes não ultrapassam 50 caracteres). O vetor resultante deve ser ordenado pelos valores numéricos, de forma crescente, a partir da primeira posição do vetor. Para ordenar o vetor compare cada elemento com o seguinte e troque-os de posição se necessário. Faça isto da primeira a última posição do vetor. Se, nesta varredura, houver ao menos uma troca de posições, será necessário repetir o procedimento. Quando realizar uma troca, não esqueça de trocar números e nomes.

```
/* Este programa ordena um vetor de estrutura contendo um campo NOME com
   um vetor de 50 caracteres e um campo numerico chamado VALOR. O Vetor e'
   ordenado em ordem crescente de valores. */

/* ALGORITMO SIMPLIFICADO
- inicio
| repetir da primeira posicao ate a ultima posicao do vetor
|   -inicio
|   |   ler os campos NOME e VALOR
|   -fim
| -fazer
| |   -inicio
| | |   numero de trocas = 0
| | |   repetir da primeira posicao ate' a penultima posicao do vetor
| | |   |   -inicio
| | |   |   |   se o Valor da posicao atual e' maior que o da proxima posicao
| | |   |   |   |   entao troca os valores, troca os nomes e faz troca = 1
| | |   |   -fim
| | |   -fim
| | -fim
| -enquanto troca=1
|
| repetir da primeira posicao ate a ultima posicao do vetor
|   -inicio
|   |   escrever campos NOME e VALOR
|   -fim
*/
```

```
/* SOLUCAO UTILIZANDO NOTACAO DE VETORES E MATRIZES */
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <string.h>

/* A diretiva de compilacao abaixo define a constante TAM_MAX como 1000.
   Este valor so' pode ser alterado antes da compilacao do programa.
   Para testar o programa altere TAM_MAX para um valor pequeno e compile o
   programa com este valor. Uma vez funcionando, retorne ao valor original
   e compile novamente o programa. */
#define TAM_MAX 1000

struct Prova {
    float Valor;
    char Nome[51];
};

void main(void)
{
    struct Prova Vetor[TAM_MAX];
    int i, troca;
    float aux_valor;
    char aux_nome[51];

    for (i=0;i<TAM_MAX;i++)
    {
        printf("Digite o Nome[%d]: ", i);
        scanf("%s", Vetor[i].Nome); /* vetores */

        printf("Digite o Valor[%d]: ", i);
        scanf("%f", &Vetor[i].Valor);
    }
    do {
        troca = 0;
        for (i=0;i<TAM_MAX-1;i++)
        {
            if (Vetor[i].Valor>Vetor[i+1].Valor) /* vetores */
            {
                troca = 1;

                aux_valor = Vetor[i].Valor;
                Vetor[i].Valor = Vetor[i+1].Valor;
                Vetor[i+1].Valor = aux_valor; /* vetores */

                strcpy(aux_nome, Vetor[i].Nome);
                strcpy(Vetor[i].Nome, Vetor[i+1].Nome);
                strcpy(Vetor[i+1].Nome, aux_nome); /* vetores */
            }
        }
    } while (troca==1);

    for (i=0;i<TAM_MAX;i++)
    {
        printf("%s \t %f \n", Vetor[i].Nome, Vetor[i].Valor); /* vetores */
    }
    getch();
}
```

```

/* SOLUCAO UTILIZANDO NOTACAO DE PONTEIROS E ALOCACAO DINAMICA DE MEMORIA */
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <string.h>
/* A diretiva de compilacao abaixo define a constante TAM_MAX como 1000.
   Este valor so' pode ser alterado antes da compilacao do programa.
   Para testar o programa altere TAM_MAX para um valor pequeno e compile o
   programa com este valor. Uma vez funcionando, retorne ao valor original
   e compile novamente o programa. */
#define TAM_MAX 1000
struct Prova {
    float Valor;
    char Nome[51];
};

void main(void)
{
    struct Prova *Vetor;
    int i, troca;
    float aux_valor;
    char aux_nome[51];

    Vetor = (struct Prova *) calloc(TAM_MAX, sizeof(struct Prova));
    if (!Vetor) {
        printf("Erro na alocao de memoria.");
        getch();
        exit;
    }
    for (i=0;i<TAM_MAX;i++) {
        printf("Digite o Nome[%d]: ", i);
        scanf("%s", (Vetor+i)->Nome);
        printf("Digite o Valor[%d]: ", i);
        scanf("%f", &(Vetor+i)->Valor);
    }
    do {
        troca = 0;
        for (i=0;i<TAM_MAX-1;i++) {
            if ((*Vetor+i).Valor > (*(Vetor+i+1)).Valor) {
                troca = 1;
                /* aux_valor = (*(Vetor+i)).Valor; */ /* opcao 1*/
                /* (*(Vetor+i)).Valor = (*(Vetor+i+1)).Valor; */ /* opcao 1*/
                /* (*(Vetor+i+1)).Valor = aux_valor; */ /* opcao 1*/
                aux_valor = (Vetor+i)->Valor; /* opcao 2*/
                (Vetor+i)->Valor = (Vetor+i+1)->Valor; /* opcao 2*/
                (Vetor+i+1)->Valor = aux_valor; /* opcao 2*/

                strcpy(aux_nome, (Vetor+i)->Nome);
                strcpy((Vetor+i)->Nome, (Vetor+i+1)->Nome);
                strcpy((Vetor+i+1)->Nome, aux_nome);
            }
        }
    } while (troca==1);
    for (i=0;i<TAM_MAX;i++) {
        printf(" %s \t %f \n", (Vetor+i)->Nome, (Vetor+i)->Valor); /* ponteiros */
    }
    getch();
    free(Vetor);
}

```

3) Escrever o algoritmo e o programa, em linguagem C, que solicite a entrada de um número ímpar entre 5 e 29 (com proteção para números fora dessa faixa) e apresente o seu quadrado mágico. O quadrado mágico de ordem n (número ímpar) é um arranjo de números variando de 1 a n^2 em uma matriz quadrada, de tal modo que a soma de cada linha, coluna e diagonal é a mesma. A figura a seguir mostra um quadrado mágico de ordem 5. A regra para gerá-lo é a seguinte: Comece com 1 no meio da primeira linha, então siga para cima e para a esquerda diagonalmente (quando sair do quadrado suponha que os lados superior e inferior estão unidos, o mesmo ocorre para os lados esquerdo e direito), ao passar por cada quadrado coloque o valor do anterior mais 1. Se for atingir um quadrado já ocupado desça uma casa e continue seguindo a diagonal até ter preenchido todos os espaços.

15	8	1	24	17
16	14	7	5	23
22	20	13	6	4
3	21	19	12	10
9	2	25	18	11

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

void main (void)
{
    int i, j;
    int Tam, Matriz[29][29];
    int Linha, Coluna, LinhaAtual, ColunaAtual;

    /* inicializa matriz maxima */
    for (i=0;i<29;i++)
        for (j=0;j<29;j++)
            Matriz[i][j] = 0;

    /* tamanho da matriz */
    do {
        printf("\nTamanho da Matriz (nro. impar entre 5 e 29): ");
        scanf("%d",&Tam);
    } while ((Tam%2==0) || (Tam>29) || (Tam<5));

    Linha = 0;
    Coluna = Tam/2;
    Matriz[Linha][Coluna] = 1;
    for (i=2;i<=Tam*Tam;i++)
    {
        LinhaAtual = Linha;
        ColunaAtual= Coluna;

        /* no canto superior */
        if (((Linha-1)<0) && ((Coluna-1)<0))
        {
            Linha = Tam-1;
            Coluna = Tam-1;
        }
        else
```

```
/* na linha superior */
if (((Linha-1)<0) && ((Coluna-1)>=0))
{
    Linha = Tam-1;
    Coluna = Coluna - 1;
}
else

/* na coluna da esquerda */
if (((Linha-1)>=0) && ((Coluna-1)<0))
{
    Linha = Linha -1;
    Coluna = Tam-1;
}
else

/* normal */
if (((Linha-1)>=0) && ((Coluna-1)>=0))
{
    Linha = Linha - 1;
    Coluna = Coluna -1;
}

/* encontrada a posicao tenta escrever */
/* se nao for possivel escreva abaixo da posicao atual */
if (Matriz[Linha][Coluna] == 0)
{
    Matriz[Linha][Coluna] = i;
}
else
{
    Linha = (LinhaAtual+1)>=Tam ? 0 : LinhaAtual+1;
    Coluna = ColunaAtual;
    Matriz[Linha][Coluna] = i;
}

}

/* Escreve o resultado
for (i=0;i<Tam;i++)
{
    for (j=0;j<Tam;j++)
        printf("%3d ", Matriz[i][j]);
    printf("\n");
}
getch();
}
```

4) Escrever um algoritmo e o programa em C para criptografar frases da seguinte forma:

- a) Escrever a mensagem em uma matriz com 4 colunas;
- b) Substituir todos os espaços por X;
- c) Formar palavras lendo o texto verticalmente na matriz;
- d) Não aceitar frases com mais de 254 caracteres.

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

void main (void)
{
    /* Matriz de 4 colunas para guardar texto de 254 caracteres: 254/4=64 linhas */
    char Texto[255], Criptografado[255], Matriz[64][4];
    int linha, coluna, i;

    /* para ler uma "string" com espacos use gets(char *) */
    /* scanf("%s", ) nao le "strings" com espaco */
    printf("Digite a mensagem\n");
    gets(Texto);

    /* inicializa a matriz com zero */
    for (linha=0;linha<64;linha++)
        for (coluna=0;coluna<4;coluna++)
            Matriz[linha][coluna] = ' ';

    /* inicializa variaveis */
    i = 0; /* posicao do caractere na "string" */
    linha = 0; /* linha da matriz onde o caractere sera' escrito */
    coluna = 0; /* coluna da matriz onde o caractere sera' escrito */

    /* varre a "string" ate' o fim, copiando cada caractere para a matriz */
    while(Texto[i]!='\0')
    {
        /* se nao e' espaco copia direto, se e' espaco substitui por X */
        if (Texto[i]!=' ')
        {
            Matriz[linha][coluna] = Texto[i];
        }
        else
        {
            Matriz[linha][coluna] = 'X';
        }

        /* se esta na ultima coluna volta para a primeira */
        /* se esta na ultima linha indica que o limite de caracteres foi obtido */
        if (++coluna > 3)
        {
            coluna = 0;
            if (++linha > 63)
            {
                printf("Limite de caracteres ultrapassado");
                getch();
                exit;
            }
        }
        i++;
    }
}
```

```
    }

    /* varre a matriz na vertical para criar a nova sentenca criptografada */
    i = 0;
    for (coluna=0;coluna<4;coluna++)
    {
        for (linha=0;linha<65;linha++)
        {
            if (Matriz[linha][coluna]==' ')
            {
                Criptografado[i] = ' ';
                i++;
                break;
            }
            Criptografado[i] = Matriz[linha][coluna];
            i++;
        }
    }

    /* adiciona o caractere '\0' (ele nao foi copiado para a matriz) */
    Criptografado[i] = '\0';

    /* imprime a mensagem criptografada */
    printf("\n%s",Criptografado);
    getch();
}
```

5) Escrever um algoritmo e o programa em C para ler registros contendo número de matrícula, sexo, curso (elétrica ou eletrônica), nome e nota de alunos. Percorra a lista de registros e imprima os dados do aluno que tenha obtido a melhor nota em cada curso. Em caso de empate, considere o aluno de maior número de matrícula. Para este programa utilize vetores e estruturas.

```
#include <stdio.h>
#include <conio.h>
#define MAX 50
#define NCURSOS 2
struct Prova {
    int Matricula;
    char Sexo;
    int Curso;
    char Nome[50];
    float Nota;
};
void main (void)
{
    float nota;
    int i, j, indice;
    struct Prova Registros[MAX];
    for (i=0;i<MAX;i++) {
        printf("Aluno %d: Nome: ", i);
        scanf("%s",Registros[i].Nome);
        printf("Aluno %d: Sexo (M-F): ", i);
        Registros[i].Sexo = getche();
        printf("\nAluno %d: Matricula: ", i);
        scanf("%d",&Registros[i].Matricula);
        printf("Aluno %d: Curso (1)Eletrica ou (2)Eletronica: ", i);
        scanf("%d",&Registros[i].Curso);
        printf("Aluno %d: Nota: ", i);
        scanf("%f",&Registros[i].Nota);
    }
    for (j=1;j<=NCURSOS;j++) {
        nota = -1; /* inicializa a maior nota dentro de um curso */
        indice = 0; /* inicializa o indice do registro com a maior nota no curso */
        for (i=0;i<MAX;i++) {
            if (Registros[i].Curso==j) {
                if (nota==Registros[i].Nota)
                    if (Registros[i].Matricula > Registros[indice].Matricula) {
                        nota = Registros[i].Nota;
                        indice = i;
                    }
                if (nota<Registros[i].Nota) {
                    nota = Registros[i].Nota;
                    indice = i;
                }
            }
        }
        printf("\nAluno %d: Matricula: %d \n",indice,Registros[indice].Matricula);
        printf("Aluno %d: Sexo : %c \n",indice,Registros[indice].Sexo);
        printf("Aluno %d: Curso : %d \n",indice,Registros[indice].Curso);
        printf("Aluno %d: Nome : %s \n",indice,Registros[indice].Nome);
        printf("Aluno %d: Nota : %f \n",indice,Registros[indice].Nota);
    }
    getch();
}
```


6) *Reverse* é o nome de um antigo jogo de tabuleiro, que pode ser facilmente implementado em um computador: consiste de um tabuleiro de 3x3 casas, com um disco branco ou preto (valor 0 ou 1) dispostos, inicialmente, de modo aleatório em cada casa.

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

Ao selecionar uma das casas o jogador reverte a cor (valor) do disco daquela casa e de algumas casas vizinhas conforme o esquema abaixo. O objetivo do jogo é reverter todas as casas para uma mesma cor. Escreva o algoritmo e o programa em linguagem C que simule o jogo *Reverse*.

Ao pressionar	Reverte:
[1]	[1],[2] e [4]
[2]	[2],[1] e [3]
[3]	[3],[2] e [6]
[4]	[4],[1] e [7]
[5]	[5],[2],[4],[6] e [8]
[6]	[6],[3] e [9]
[7]	[7],[4] e [8]
[8]	[8],[7] e [9]
[9]	[9],[6] e [8]

DICAS: Utilize funções para: inicializar o tabuleiro; desenhar o tabuleiro; ler o lance do jogador; atualizar o estado do tabuleiro; testar o fim do jogo. Mostre o algoritmo principal utilizando estas tarefas e depois detalhe cada função em um algoritmo separado. Para sortear um valor inicial à cada uma das casas, utilize: `variavel = rand() % 2;` /* biblioteca `stdlib.h` */

```
#include <stdlib.h>
#include <stdio.h>
#include <conio.h>
void inicializa_tabuleiro(void);
void desenha_tabuleiro(void);
int le_lance(void);
void atualiza_tabuleiro(int lance);
int testa_fim(void);
int Matriz[3][3];

void main (void)
{
    int lance;
    int resultado;
    char opcao;
    do {
        inicializa_tabuleiro();
        do {
            desenha_tabuleiro();
            lance = le_lance();
            atualiza_tabuleiro(lance);
            resultado = testa_fim();
        } while (resultado);
        printf("\nParabens voce ganhou!\n");
        printf("Deseja jogar novamente (s/n)? ");
        opcao = getche();
    } while (opcao=='s' || opcao=='S');
}
```

```

/*****/
void inicializa_tabuleiro(void)
{
    int linha, coluna;
    for(linha=0;linha<3;linha++)
        for(coluna=0;coluna<3;coluna++)
            Matriz[linha][coluna] = rand()%2;
}
/*****/
void desenha_tabuleiro(void)
{
    int linha, coluna, posicao;
    printf("\nTabuleiro Atual Codigo de Posicoes \n");
    for(linha=0;linha<3;linha++)
    {
        for(coluna=0;coluna<3;coluna++)
            printf(" %2d ", Matriz[linha][coluna]);
        /* so para ficar bonito na tela */
        posicao = linha*3+1;
        printf(" [%d][%d][%d]\n",posicao,posicao+1,posicao+2);
    }
}
/*****/
int le_lance(void)
{
    int i;
    printf("\nQual o seu lance (-1 finaliza)? ");
    scanf("%d",&i);
    return i;
}
/*****/
void atualiza_tabuleiro(int lance)
{
    switch (lance) {
        case 1:
            /* [1],[2] e [4] */
            Matriz[0][0] = Matriz[0][0]?0:1;
            Matriz[0][1] = Matriz[0][1]?0:1;
            Matriz[1][0] = Matriz[1][0]?0:1;
            break;
        case 2:
            /* [2],[1] e [3] */
            Matriz[0][0] = Matriz[0][0]?0:1;
            Matriz[0][1] = Matriz[0][1]?0:1;
            Matriz[0][2] = Matriz[0][2]?0:1;
            break;
        case 3:
            /* [3],[2] e [6] */
            Matriz[0][2] = Matriz[0][2]?0:1;
            Matriz[0][1] = Matriz[0][1]?0:1;
            Matriz[1][2] = Matriz[1][2]?0:1;
            break;
        case 4:
            /* [4],[1] e [7] */
            Matriz[1][0] = Matriz[1][0]?0:1;
            Matriz[0][0] = Matriz[0][0]?0:1;
            Matriz[2][0] = Matriz[2][0]?0:1;
            break;
    }
}
```

```
case 5:
    /* [5],[2],[4],[6] e [8] */
    Matriz[1][1] = Matriz[1][1]?0:1;
    Matriz[0][1] = Matriz[0][1]?0:1;
    Matriz[1][0] = Matriz[1][0]?0:1;
    Matriz[1][2] = Matriz[1][2]?0:1;
    Matriz[2][1] = Matriz[2][1]?0:1;
    break;
case 6:
    /* [6],[3] e [9] */
    Matriz[1][2] = Matriz[1][2]?0:1;
    Matriz[0][2] = Matriz[0][2]?0:1;
    Matriz[2][2] = Matriz[2][2]?0:1;
    break;
case 7:
    /* [7],[4] e [8] */
    Matriz[2][0] = Matriz[2][0]?0:1;
    Matriz[1][0] = Matriz[1][0]?0:1;
    Matriz[2][1] = Matriz[2][1]?0:1;
    break;
case 8:
    /* [8],[7] e [9] */
    Matriz[2][1] = Matriz[2][1]?0:1;
    Matriz[2][0] = Matriz[2][0]?0:1;
    Matriz[2][2] = Matriz[2][2]?0:1;
    break;
case 9:
    /* [9],[6] e [8] */
    Matriz[2][2] = Matriz[2][2]?0:1;
    Matriz[1][2] = Matriz[1][2]?0:1;
    Matriz[2][1] = Matriz[2][1]?0:1;
    break;
case -1:
    printf("\nFoi bom ter jogado com voce. Tente de novo mais tarde.");
    printf("\nPressione uma tecla para retornar ao windows.");
    getch();
    exit(0);
    break;
}
}
/*****/
int testa_fim(void)
{
    int linha, coluna, aux, fim;
    fim = 0;
    aux = Matriz[0][0];
    for(linha=0;linha<3;linha++)
        for(coluna=0;coluna<3;coluna++)
            fim = fim + (Matriz[linha][coluna]==Matriz[0][0]);
    return fim;
}
```

7) No programa que segue, complete as linhas que faltam abaixo dos comentários 2, 3 e 4.

```
main()
{
/* 1 - criar e inicializar variaveis */
  int *zptr;
  int z[5] = { 10, 20, 30, 40, 50 };
  int numero, i;

/* 2 - usar o ponteiro para obter o endereço de início do vetor
*/

printf("O endereço inicial do vetor é ' %u",
);

/* 3 - com o ponteiro, atribuir o segundo elemento do array à
variável numero */

printf("A segunda posição do vetor contém: %d",
);

/* 4 - imprimir todo o array usando aritmética de ponteiros */

}
```

8) Escreva o algoritmo e o programa em linguagem C para ler e armazenar em um vetor, inteiros maiores ou iguais a zero. Armazene os valores de tal forma que o vetor esteja sempre ordenado (com o maior valor na posição 0 e o menor valor na última posição do vetor). O vetor pode armazenar um máximo de 1024 elementos mas seu preenchimento pode se encerrar precocemente se um valor negativo for informado.

```
#include <stdio.h>
#include <conio.h>
#define MAX 1024

void main(void)
{
    int vetor[MAX], numero;
    int i,j,total;

    total = 0;
    for(i=0;i<MAX;i++)
        vetor[i] = -1;

    do {
        printf("Digite um numero: ");
        scanf("%d", &numero);
        if (numero<0) break;
        for(i=0;i<MAX;i++)
        {
            if (vetor[i]<numero)
            {
                for(j=MAX-1;j>i;j--)
                    vetor[j] = vetor[j-1];
                vetor[i] = numero;
                break;
            }
        }
        total++;
    } while (total<MAX);

    for(i=0;i<MAX;i++)
        printf("\nVetor ordenado: V[%d]=%d",i,vetor[i]);

    printf("\nDigite algo para retornar ao windows");
    getch();
}
```

9) Em um só programa resolver os 6 problemas listados abaixo:

- a) Multiplicar um vetor coluna por um vetor linha;
- b) Multiplicar uma matriz quadrada por uma matriz quadrada com as mesmas dimensões;
- c) Multiplicar um vetor linha por um vetor coluna utilizando notação de ponteiros;
- d) Simular os comandos PUSH e POP em um vetor;
- e) Copiar uma string que contém apenas letras para outra string, convertendo todas as letras para maiúsculas. Não utilizar as funções especiais do C;
- f) Copiar uma string que contém apenas letras para outra string trocando as maiúsculas por minúsculas e vice versa.

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

void ColunaLinha(void);
void MatrizMatriz(void);
void LinhaColuna(void);
void PushPop(void);
void CopiarString(void);
void CopiarStringTrocada(void);

void main(void)
{
    char opcao;

    while(1) {
        clrscr();
        printf("Escolha a opcao: \n");
        printf("(1) Multiplica Coluna por Linha \n");
        printf("(2) Multiplica Matriz por Matriz \n");
        printf("(3) Multiplica Linha por Coluna (com ponteiros) \n");
        printf("(4) Push-Pop \n");
        printf("(5) Copiar String com Maiusculas\n");
        printf("(6) Copiar String Trocando Maiusculas e Minusculas\n");
        printf("(0) Sair \n");
        opcao = getch();

        switch(opcao)
        {
            case '1': ColunaLinha();
                       break;
            case '2': MatrizMatriz();
                       break;
            case '3': LinhaColuna();
                       break;
            case '4': PushPop();
                       break;
            case '5': CopiarString();
                       break;
            case '6': CopiarStringTrocada();
                       break;
            case '0': exit(0);
                       break;
        }
    }
}
```

```
/****** ColunaLinha *****/
void ColunaLinha(void)
{
    int C[50],L[50],R[50][50];
    int tam, i,j;

    printf("\n");
    do {
        printf("Qual o tamanho dos vetores (max=49)? ");
        scanf("%d", &tam);
    } while (tam<1 || tam>49);

    for(i=0;i<tam;i++)
    {
        printf("C[%d]=",i);
        scanf("%d",&C[i]);
        printf("L[%d]=",i);
        scanf("%d",&L[i]);
    }

    for(i=0;i<tam;i++)
    {
        for(j=0;j<tam;j++)
        {
            R[i][j] = L[i]*C[j];
            printf("%d  ", R[i][j]);
        }
        printf("\n");
    }
    printf("\nPressione uma tecla para retornar ao menu de opcoes \n");
    getch();
}
```

```
/****** MatrizMatriz *****/
void MatrizMatriz(void)
{
    int A[50][50],B[50][50],R[50][50];
    int tam, i,j,k;

    printf("\n");
    do {
        printf("Qual o tamanho dos vetores (max=49)? ");
        scanf("%d", &tam);
    } while (tam<1 || tam>49);

    for(i=0;i<tam;i++)
        for(j=0;j<tam;j++)
        {
            printf("A[%d][%d]=",i,j);
            scanf("%d",&A[i][j]);

            R[i][j] = 0;
        }

    for(i=0;i<tam;i++)
        for(j=0;j<tam;j++)
        {
            printf("B[%d][%d]=",i,j);
            scanf("%d",&B[i][j]);
        }

    for(i=0;i<tam;i++)
        for(j=0;j<tam;j++)
            for(k=0;k<tam;k++)
                R[i][j] = R[i][j] + A[i][k]*B[k][j];

    for(i=0;i<tam;i++)
    {
        for(j=0;j<tam;j++)
            printf("%d ", R[i][j]);
        printf("\n");
    }

    printf("\nPressione uma tecla para retornar ao menu de opcoes \n");
    getch();
}
```



```
/****** LinhaColuna *****/
void LinhaColuna(void)
{
    int C[50],L[50],R;
    int *pc, *pl, *pr;
    int tam, i,j;

    pc = C;
    pl = L;
    pr = &R;

    *r = 0;
    printf("\n");
    do {
        printf("Qual o tamanho dos vetores (max=49)? ");
        scanf("%d", &tam);
    } while (tam<1 || tam>49);

    for(i=0;i<tam;i++)
    {
        printf("C[%d]=",i);
        scanf("%d", (pc+i)); /*scanf("%d",&C[i]);*/
        printf("L[%d]=",i);
        scanf("%d", (pl+i)); /*scanf("%d",&L[i]);*/
    }

    for(i=0;i<tam;i++)
    {
        *pr = *pr + *(pl+i) * *(pc+i); /*R = R + L[i]*C[i];*/
    }
    printf("Resultado = %d \n", *pr); /*printf("Resultado = %d \n", R);*/

    printf("\nPressione uma tecla para retornar ao menu de opcoes \n");
    getch();
}
```

```
/****** PushPop *****/
int *tos, *pl, stack[50];

/******/
int push(int i)
{
    if(pl==(tos+5))
    {
        printf("Estouro da pilha!\n");
        return(1);
    }
    pl++;
    *pl = i;
}

/******/
int pop(void)
{
    if(pl==tos)
    {
        printf("Pilha vazia!\n");
        return(-1);
    }
    pl--;
    return(*(pl+1));
}
/******/
void PushPop(void)
{
    int value;
    tos = stack;
    pl = stack;
    do {
        printf("Entre com o valor: ");
        scanf("%d",&value);
        if(value!=0)
            push(value);
        else
            printf("Valor do topo e': %d\n",pop());
    } while (value!=-1);
}
```

```

/***** CopiarString *****/
void CopiarString(void)
{
    char A[51],B[51];
    int i;

    printf("\nDigite uma String que contenha apenas letras (Max=50
caracteres):\n");
    gets(A);
    for(i=0;A[i];i++)
        if (A[i]>='a')
            A[i]=A[i]-'a'+'A';
    A[i] = '\0';
    printf("%s",A);

    printf("\nPressione uma tecla para retornar ao menu de opcoes \n");
    getch();
}

/***** CopiarStringTrocada *****/
void CopiarStringTrocada(void)
{
    char A[51],B[51];
    int i;

    printf("\nDigite uma String que contenha apenas letras (Max=50
caracteres):\n");
    gets(A);
    for(i=0;A[i];i++)
    {
        if (A[i]>='a')
            A[i]=A[i]-'a'+'A';
        else if (A[i]< 'A')
            A[i]=A[i]-'A'+'a';
    }
    A[i] = '\0';
    printf("%s",A);

    printf("\nPressione uma tecla para retornar ao menu de opcoes \n");
    getch();
}

```

10) O código abaixo é um algoritmo ou um programa em C? Ele compila? O que ele faz?

```
/****** DEFINICOES *****/

#include <stdio.h>
#include <conio.h>

typedef char  caracter;
typedef int   inteiro;
typedef float real;

#define tipo struct
#define registro {

#define se if
#define entao
#define senao else
#define escolha switch
#define caso case

#define inicio {
#define fim }

#define leia scanf
#define escreva printf

#define repita do
#define enquanto while
#define INICIO void main(void) {
#define FIM printf("\nDigite algo para terminar" ); getch(); }
#define para for(
#define ate ;
#define passo ;
#define faca )

tipo professor registro
    caracter nome[50],dep[50];
    caracter cat;
    inteiro  pront;
fim;

/****** PROGRAMA PRINCIPAL *****/
```

INICIO

```
professor p[2000];
inteiro i,x,contA,contM,contD,maior;

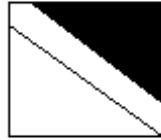
contA = 0; contM = 0; contD = 0;
i = 1; maior = 0;
repita
  inicio
    escreva("Numero de prontuario: ");
    leia("%d",&p[ i ].pront);
    se (p[i].pront != -1) entao
      inicio
        escreva("Nome: ");
        leia("%s",p[i].nome);
        repita
          inicio
            escreva("Categoria: ");
            leia("%c",&p[i].cat);
          fim
        enquanto (!(p[i].cat=='A' || p[i].cat=='M' || p[i].cat=='D'));
        escreva("Departamento: ");
        leia("%s",p[ i ].dep);
        escolha (p[i].cat)
        inicio
          caso 'A': contA = contA + 1;
                    break;
          caso 'M': contM = contM + 1;
                    break;
          caso 'D': contD = contD + 1;
                    break;
        fim
        i = i + 1;
      fim
    fim
  enquanto ((p[i].pront != -1) && (i <= 1999));

se (contA > contM) entao maior = contA;
senao maior = contM;

se (contD > maior) entao maior = contD;
escreva("O maior: %d",maior);

para x=0 ate x<=i-1 passo x=x+1 faca
  inicio
    se (p[ x ].cat == 'D') entao
      escreva("Nome: %s, Departamento: %s \n",p[ x ].nome,p[ x ].dep);
  fim
FIM
```

10) Ler uma matriz **A** de dimensões 4 x 4, calcular e escrever as somas das partes hachuradas. Utilizar estruturas de repetição.



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

void main(void)
{
    int i, j, A[4][4], Soma=0;

    for(i=0;i<4;i++)
        for(j=0;j<4;j++)
        {
            printf("A[%d][%d]= ", i, j);
            scanf("%d", &A[i][j]);
            if (j>i) Soma=Soma + A[i][j];
        }
    printf("A soma dos elementos acima da diagonal principal e' %d", Soma);
    getch();
}
```

11) Usando uma estrutura capaz de armazenar Nome, Idade e Telefone, criar uma agenda de contatos que possa ser ordenada por nome. Organizar todos os dados em um vetor de estruturas.

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
#define MAX 5
struct TAgenda {
    char Nome[50], Telefone[20], Idade[5];
};
void main(void)
{
    TAgenda aux_agenda, agenda[MAX];
    char aux_char[50];
    int i,j;
    for(i=0;i<MAX;i++)
    {
        printf("Posicao de memoria %d\n", i);
        printf("Nome: ");
        gets(agenda[i].Nome);
        printf("Telefone: ");
        gets(agenda[i].Telefone);
        printf("Idade: ");
        gets(agenda[i].Idade);
    }
    for(i=0;i<MAX-1;i++)
        for(j=i;j<MAX;j++)
            if (strcmp(agenda[i].Nome,agenda[j].Nome)>0)
            {
                aux_agenda = agenda[i];
                agenda[i] = agenda[j];
                agenda[j] = aux_agenda;
                /* alternativa: trocar cada campo individualmente */
                /*strcpy(aux_char,agenda[i].Nome);
                strcpy(agenda[i].Nome,agenda[j].Nome);
                strcpy(agenda[j].Nome,aux_char);
                strcpy(aux_char,agenda[i].Telefone);
                strcpy(agenda[i].Telefone,agenda[j].Telefone);
                strcpy(agenda[j].Telefone,aux_char);
                strcpy(aux_char,agenda[i].Idade);
                strcpy(agenda[i].Idade,agenda[j].Idade);
                strcpy(agenda[j].Idade,aux_char);
                */
            }
    for(i=0;i<MAX;i++)
    {
        printf("\nPosicao de memoria %d\n", i);
        printf("Nome: %s\n", agenda[i].Nome);
        printf("Telefone: %s\n", agenda[i].Telefone);
        printf("Idade: %s\n", agenda[i].Idade);
    }
    getch();
}
```

12) Ler o nome de uma pessoa e separá-lo em três variáveis distintas contendo cada uma delas o 1º o 2º e o 3º nome da pessoa. Usar ponteiros.

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

void main(void)
{
    char nome[50],primeiro[50],segundo[50],terceiro[50];
    char *p,*s;

    p = nome;
    printf("Nome da pessoa: ");
    gets(nome);

    s = primeiro;
    while(*p != ' ') *s++ = *p++;
    *s = '\\0';
    p++;

    s = segundo;
    while(*p != ' ') *s++ = *p++;
    *s = '\\0';
    p++;

    s = terceiro;
    while(*p != '\\0') *s++ = *p++;
    *s = '\\0';

    printf("N1= %s, N2= %s, N3= %s",primeiro, segundo, terceiro);
    getch();
}
```


13) Escrever uma função que receba um ponteiro para um vetor de inteiros e o número de elementos do vetor, inverta a ordem dos elementos do vetor e retorne ao programa principal. Usar ponteiros.

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

void troca (int *p, int n)
{
    int i, aux;

    for(i=0;i<(n/2);i++)
    {
        aux = *(p+i);
        *(p+i) = *(p+n-i-1);
        *(p+n-i-1) = aux;
    }
}

void main(void)
{
    int i,x[5];

    for(i=0;i<5;i++)
        x[i]=i;
    troca(x,5);
    for(i=0;i<5;i++)
        printf("%d ",x[i]);
    getch();
}
```

14) Leia uma string e copie seus caracteres de forma que a primeira metade da string resultante contenha apenas os caracteres com código ASCII par e a segunda metade da string contenha apenas os caracteres com código ASCII ímpar. Utilize ponteiros.

```
/* ***** */
void main(void)
{
    char string1[50], string2[50];
    char *p1,*p2;

    p2=string2;

    printf("Digite o texto:");
    gets(string1);

    p1=string1;
    while(*p1!='\0')
        if ((*p1%2)==0) *p2++=*p1++;
        else p1++;

    p1=string1;
    while(*p1!='\0')
        if ((*p1%2)==1) *p2++=*p1++;
        else p1++;

    *p2 = '\0';

    printf("A string misturada e': %s",string2);
}
```