

Algoritmos e Linguagem - LAB 01 / 22

- 1) Escreva programa que desenhe na tela as iniciais de seu nome formadas com o caracter **#** usando printf() e caracteres de controle, com cada caracter em tamanho 6 linhas por 6 colunas.
- 2) Teste o código dos programas INTASIGN.C, LOTYPES.C e MORETYPES. Acrescente comentários explicativos ao código e adicione **instruções de saída** para verificar a correção dos resultados.
- 3) Escreva um programa para imprimir os nomes e valores das variáveis abaixo nos formatos: decimal (%d), hexadecimal (%x), unsigned int (%u), character output (%c), ponto flutuante de precisão simples (%f), ponto flutuante de precisão dupla (%lf). Dados:

```
int i;           /* simple integer type           */
char c;          /* character type                                */
double d;        /* floating point type                           */
```

Teste com: a=-123; c='A'; d=3.1415926535898;

Verifique a correção das saídas. Explique possíveis divergências com os valores atribuídos às variáveis. Como podemos converter os tipos das variáveis e corrigir as discrepâncias? Quantos dígitos são significativos nas saídas de float e de double?

- 4) Assuma que todas as variáveis das expressões a seguir são do **tipo inteiro**. Encontre o valor de cada expressão e escreva um programa que as imprima para verificar os resultados.
i) $x = (2+3)*6$ ii) $y = 5/(2*3)$ iii) $i = j = (3+2)/4$ iv) $a = 3+2*(b=7/2)$
v) $c = 5+10\%2/3$ vi) $d = -(3\%5)*3/2$ vii) $x = -3+4*5-6$ viii) $x = (7+6)\%5/2$
- 5) Imprima a tabela ASCII, com valores e "desenho" de cada caracter. Experimente com valores entre 0 e 127. Obs.: Alguns caracteres não aparecem no monitor. Comente. Dica: %d imprime valor, %c imprime "desenho". Em C caracteres são representados, internamente, como números inteiros
- 6) Faça um gráfico com caracteres para representar $y=x$ e $y=2x$ e $y=0.5x$. Use a rotina w_gotoxy(x,y) baseada na **API do Windows**. A coordenada no monitor são x = colunas e y = linhas, com origem (0,0) no canto superior esquerdo. Ver exemplo.

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

```
int w_gotoxy( const WORD x, const WORD y )
{
    COORD xy;
    xy.X = x;
    xy.Y = y;
    return SetConsoleCursorPosition( GetStdHandle( STD_OUTPUT_HANDLE ), xy );
}
```

```
int main(void)
{
    system("cls");
    for(x=2; x<80; x++)
    {
        w_gotoxy(x, 0.25*x);
        putchar(178);
    }
    for(x=2; x<25; x++)
```

```

{
    w_gotoxy(x, 1.0*x);
    putchar(178);
}
for(x=2; x<25; x++)
{
    w_gotoxy(x, 1.5*x);
    putchar(178);
}
}

```

- a. Use rotina do exemplo, baseada na API do Windows.
- b. Instale no DevCpp as funções de conio.h para usar gotoxy().

Seu programa deve inverter o eixo y, para que a origem esteja no canto inferior esquerdo.

Observe que você está usando modo caracter. Comente sobre a precisão nos plots. Observe o efeito de serrilhamento ... degraus em cada plot. Em qual dos plots o efeito é mais pronunciado? Há como melhorar, ainda em modo caracter?

- 7) As funções em conio.h incluem getch() para leitura de caracter. Escreva programa para comparar funcionamento de getchar() e getch().
- 8) Sabendo que o valor ZERO (0) é interpretado como FALSO e qualquer valor diferente de zero é interpretado como verdadeiro, escreva um programa para testar se as expressões a seguir são falsas(valor zero) ou verdadeiras (valor diferente de zero).
 - a) $1 > 2$ b) $'a' > 'b'$ c) $2 == 0$ d) $'1' == '1'$ e) $3 >= 2$ f) $'A' != 'A'$ g) $a = 0$

Insira cada expressão dentro de uma instrução if().. else de modo a imprimir mensagem com a expressão dada e a mensagem FALSO ou VERDADEIRO.

Exemplo: `if(3<2)`
 `printf("3<2 --- %d --- verdadeiro/n" 3<2);`
 `else`
 `printf("3<2 --- %d --- falso/n", 3<2);`

- 9) Escreva programa que leia o número de horas semanais trabalhadas e o valor da hora de trabalho e calcule o salário semanal. **Inclua as horas extras pagando 50% a mais que o valor das horas normais.** Imprima o salário do empregado. Por simplicidade, considere controle feito por semana, de modo que até 40 horas trabalhadas na semana são consideradas horas normais; acima disso considere a diferença como horas extras.
- 10) Leia os valores a, b e c dos coeficientes de uma equação de segundo grau ($a*x*x + b*x + c = 0$), calcule e imprima as suas raízes. Leia os valores de a, b e c. Seu programa deve tratar casos de raízes complexas, raízes reais e iguais, e raízes reais e diferentes. Trate também o caso em que $a=0$. Compare com as soluções teóricas. Dica.: Monte as equações a serem testadas a partir das raízes. Exemplo: $(x-2)*(x-1) = x*x-3*x+2 \Leftrightarrow a=1, b=-3, c=2$.
- 11) Escreva programa para implementar uma calculadora de operações com números inteiros. Operações *, /, %, +, -. Use **switch...case**. você deve ler dois números inteiros separados por operador. Dica: ler ... número_1 operador número_2 com uma única instrução scanf().
- 12) Leia dois números inteiros positivos, n_1 e n_2, e calcule a soma dos múltiplos de 3 entre eles (inclua os extremos).