



ANO XV - Nº 167 - R\$ 5,00

# **Micro Sistemas**

A PRIMEIRA REVISTA BRASILEIRA DE MICROCOMPUTADORES

## **Redes Neurais Artificiais**

**Micro Processadores  
e Placas de CPU**

**Proteção Máxima  
para o seu  
Disco Rígido**

**Informática na  
Medicina**

**Convertendo Dígitos  
em Valores por  
Extenso em  
Object Pascal**

**Matemática  
Computacional  
Empregada  
na Análise  
Estratégica  
das Empresas**



# TYPON

## AG fUR Fotografische Industrie

### QUALIDADE A TODA PROVA

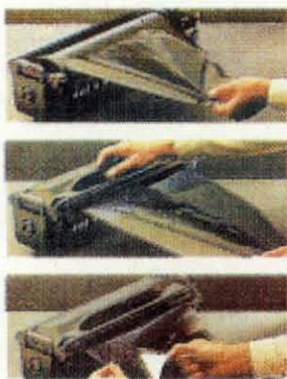
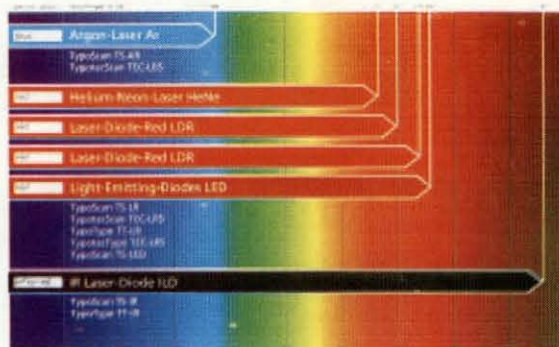
**TYPON**, é uma empresa suíça que fabrica filmes para a Indústria de artes gráficas com reputação internacional há mais de 50 anos, além de toda a qualidade e produtividade efetivamente comprovada por consumidores europeus, os filmes para Scanner e imagesetter **TYPON** foram, ainda, mais aperfeiçoados em suas propriedades mecânicas. Hoje, os clientes **TYPON** passaram a beneficiar-se com:

- Melhorada e mais prolongada proteção eletrostática
- Melhor e mais fácil manuseio nos equipamentos de Imagesetter
- Significativo aumento de resistência a pressões e danos ao filme



### FILMES TYPOSCAN E TYPOTYPE

#### Dimensão e aplicações



A **TYPON** detém a mais moderna tecnologia "Day Light Load". Esta tecnologia de ponta permite um carregamento seguro dos filmes em ambiente normal de luz do dia, dispensando totalmente o uso de câmara escura.

#### ISO QUALITY CERTIFICATE

Recebido em março de 1994.

Qualidade é prioridade para a **TYPON**. O Certificado de Qualidade ISO 9001 faz com que nossos clientes não precisem apenas acreditar na palavra da **TYPON**.

Para adquirir os produtos **TYPON** no Brasil e obter maiores informações, contate:



**DISTRIBUIDOR EXCLUSIVO**  
COMERCIAL IMPORTADORA  
**SÉPIA** DA

#### COMERCIAL IMPORTADORA SÉPIA LTDA.

Rua Alves Montes, nº 21, São Cristóvão Rio de Janeiro - CEP 20920-430

Tel.: (021) 589-5581 - FAX: (021) 589-3356



## ARTIGO CURSO

### PROTEÇÃO MÁXIMA PARA O SEU HARD DISK

Ruben Nascimento da Rocha 14

### REDES NEURAIS ARTIFICIAIS Parte 1

Prof. Carlos A. Thompson Ph.D 20

### JOGO INFANTIL EM VISUAL BASIC

Mauri Resende Faria 24

### Convertendo Dígitos em Valores por Extenso em Object Pascal (Delphi)

Ricardo Flores 6

### Matemática Computacional Empregada na Análise Estratégica das Empresas - Parte 2

Carlos A. Thompson Ph.D.

Marly M. Friaes \*

Eduardo G. Porto \* 8

### INFORMÁTICA NA MEDICINA

Renato Francisco G. Amaral 41

### WORDSTAR NOW WINDOWS (WS2WIN)

Carlos Roberto Macedonio 44

### FiveWin e a Migração de Aplicações Clliper para o Windows Parte 3 de 5

Vilian F. Arraes 35

### MENU FACIL PARA PASCAL

Gilberto Vincentin 38

## COLUNA DO LAÉRCIO

### Microprocessadores e Placas de CPU

Laércio Vasconcelos 28

**Diretor:** Marcelo Zóchio

**Editor:** Ricardo Flores

**Redação:** Marcia Corrêa

**Arte:** Marcelo Zóchio e Júlio M. F. Santos

**Colaboradores:** Carlos A. Thompson, Ph.D.,  
Ruben Nascimento da Rocha, Renato Degiovani,  
Laércio Vasconcelos e Vilian F. Arraes.

**Distribuição:** Fernando Chinaglia Dist. Ltda.  
Tel.: (021) 575-7725 / 575-7766

**Assinaturas:** 1 ano - R\$60,00 / 2 anos -  
R\$120,00

Venda de Assinaturas

**Rio de Janeiro:** Alexandre Rodrigues Nogueira

Tel/Fax: (021) 532-0770 cod.4016779

(2ª a 6ª de 09 às 16 horas)

Os artigos publicados são de responsabilidade única e exclusiva dos autores. Todos os direitos do conteúdo da revista estão reservados e qualquer reprodução, com finalidade comercial ou não, só poderá ser feita mediante autorização prévia. Transcrições parciais de textos para comentários ou referências podem ser feitas, desde que sejam mencionados os dados bibliográficos de Micro Sistemas.

**REVISTA MICRO SISTEMAS**

Caixa Postal 18347

Rio de Janeiro - RJ - CEP 20722-970



# BITS & bytes

NOVIDADES

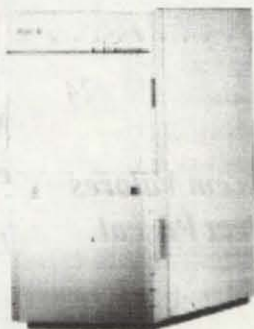
## ACER LANÇA SERVIDOR VOLTADO PARA REDES DE PEQUENO E MÉDIO PORTE

Servidor Altos Uno supre todas as necessidades desses segmentos com preços próximos ao de uma estação de trabalho.

A **Acer do Brasil** participou da Condex 97 apresentando ao mercado toda sua linha de produtos, composta por servidores, micros PC multimídia, desktops corporativos e notebooks.

Além disso, durante a feira, a Acer do Brasil esteve lançando um novo servidor voltado para o mercado low-end. Denominado **AcerAltos UNO**, o novo equipamento foi desenvolvido especialmente para atender às necessidades das empresas, oferecendo a melhor combinação de rendimento, confiabilidade e preço.

Destinado às redes de pequeno e médio porte, **AcerAltos UNO** reúne características que visam facilitar o dia-a-dia destas companhias. Com design compacto e no formato de mini-torre, o novo servidor é fácil de ser instalado, possui alta capacidade de expansão, recursos para configura-



ção e monitoramento do sistema, contando ainda com dispositivos de alarmes, que detectam e informam o usuário sobre possíveis problemas antes de parar.

Segundo Marco Zapparoli, gerente do departamento de servidores da Acer do Brasil, o AcerAltos UNO além de ser uma excelente solução para as pequenas e médias redes é também indicado como servidor departamental.

O preço é outro importante diferencial do novo equipamento. "Por pouco mais do que custaria uma estação de trabalho, adaptada para operar como servidor de rede, o usuário pode contar com todos os recursos de um verdadeiro servidor corporativo", afirma Zapparoli, lembrando que, atualmente as empresas de menor porte utilizam micros desktop como servidores de rede, o que pode ocasionar problemas sérios com lentidão na troca de dados e até travamento do sistema.

## YANCO LANÇA MOUSE COM TECNOLOGIA EXCLUSIVA E PREÇO COMPETITIVO

A **Yanco Tecnologia**, empresa nacional fabricante de acessórios para informática e para automação comercial está lançando no mercado mais um mouse inovador. Batizado de **Yanco Toppo** o equipamento substituirá com vantagens o Yanco Optic, atual carro-chefe da linha de mouses da empresa.

Também com tecnologia semi-ótica o **Yanco Toppo** possui características exclusivas e inovadoras. Os principais destaques do novo mouse são o design ergonômico, o tempo de rolagem, estimado em 480 quilômetros, e a vida útil de seus botões, que foram projetados para mais de um milhão de acionamentos. Estas características garantem uma média de 100 mil horas de uso sem apresentação de nenhum defeito.

A tecnologia semi-ótica permite transcodificar os movimentos da esfera, localizada na parte inferior do mouse, que fica em contato da superfície através de um sensor infravermelho. Nos mouses tradicionais essa tarefa é feita de forma mecânica, o que compromete a vida útil do equipamento. Já, os dois botões do

**Yanco Toppo** possuem resolução real de 400 dpi e dinâmica entre 4 e 40.000 dpi. Além disso, o novo mouse conta com uma incrível velocidade de operação máxima na faixa de 500mm/s.

Além disso, sua embalagem foi totalmente reformulada apresentando dimensões maiores, suporte para ser colocada em gôndolas de loja e uma janela transparente que permite maior visualização do produto.

Outra grande vantagem do Yanco Toppo é o preço. Compatível com os ambientes DOS, Windows 3.1 e Windows 95. O Yanco Toppo possui garantia de um ano para defeitos de fabricação e possui certificação FCC e CE.

Com o lançamento, a Yanco pretende conquistar ainda mais espaço no mercado brasileiro de acessórios para informática. Atualmente, a empresa comercializa 9.000 mouses por mês. Para o Yanco Toppo a expectativa é comercializar mensalmente 15.000 unidades, afirma Cássio Pedrão, diretor de marketing da Yanco Tecnologia.





**disquete=ASMwin+Bitmap+ASM86+TILTonline**

# A SUPER NOVIDADE DA PRO KIT

A PRO KIT juntou todas as suas produções editoriais numa única série: a TILT disquete. Agora ficou muito mais fácil aprender os truques e segredos da programação for Windows e da criação e arte no computador. Veja só o resultado:

## ProgramaçãoDELPHI

Dicas, programas, funções, procedures e muito mais. Tudo escrito de uma forma simples e fácil de compreender. Aqui você aprenderá como criar animação e jogos.

## Artes&Efeitos

As mais quentes matérias da série Bitmap: efeitos especiais, criação artística, dicas de animação e edição gráfica.

## AssemblerforWindows

A TILT disquete é a única revista onde você pode realmente aprender os segredos da programação Assembler, começando desde os conceitos básicos do computador indo até programas completos.

## Jogos&Aventuras

Nenhuma outra revista apresenta fontes de jogos, criados aqui mesmo no Brasil. São jogos de todos os tipos, desde os mais simples jogos de reflexo até os sofisticados Adventures.

**OBS:** Se você tem algum disco da série ASMwin, ao fazer um pedido, envie-nos o disquete que ele valerá uma TILT disquete inteiramente grátis.

Já estão  
disponíveis:

disco 0...R\$ 5,70 [ ]  
disco 1...R\$ 5,70 [ ]  
disco 2...R\$ 5,70 [ ]  
disco 3...R\$ 5,70 [ ]  
disco 4...R\$ 5,70 [ ]

em agosto:  
disco 5...R\$ 5,70 [ ]

Visite a TILT Online: <http://www.orldatec.com.br/prokit/tilt.htm>

## OUTROS PRODUTOS DA PRO KIT

- [ ] Sistema EDITOR de Livro de Aventuras (win)...R\$14,70
- [ ] Amazônia (dos/cdrom)...R\$25,00
- [ ] Amazônia (win/disquetes 31/2)...R\$14,70
- [ ] Curso completo de Assembler (dos/disquete 31/2)...R\$98,00
- [ ] Graphos III + Topview (dos/disquete 31/2)...R\$35,00

## PRO KIT Informática

Caixa Postal 76 - Orlandia / SP  
cep 14620-000 fax (016) 726 6820

## COMO FAZER O PEDIDO

#Envie cheque nominal à PROKIT Informática, por carta. Proteja bem o cheque, envolvendo-o em papel opaco. Nos pedidos acima de R\$50,00 use carta registrada para maior segurança.

#Depósito bancário em conta corrente BRADESCO, agência 3176-3 conta 077270-4, nominal a Renato Degiovani. Envie o comprovante do depósito por carta ou fax.

nome: \_\_\_\_\_

endereço: \_\_\_\_\_

cidade: \_\_\_\_\_ Uf \_\_\_\_\_

cep: \_\_\_\_\_ data: \_\_\_\_\_

Cheque Número: \_\_\_\_\_ • Valor R\$ \_\_\_\_\_



# ARTIGO

## Convertendo Dígitos em Valores por Extenso em Object Pascal (Delphi)

Por: Ricardo Flores

*Estava navegando na Web a procura de divertimento, clk clk clk ... quando cheguei ao site do Delphi Brasil (<http://www.geocities.com/SiliconValley/8314/delphi.htm>) que aliás, meu prezado leitor programador em Object Pascal, é um local virtual onde você deveria bater seu ponto (magnético é claro), pelo menos uma vez por mês. Lá encontrei um código fonte muito bem elaborado que também traduz dígitos em valores por extenso. Entretanto, em minha opinião, somente programadores avançados em lógica e na sintaxe Object Pascal poderiam acompanhar tal código. Foi aí que resolvi fazer este artigo, tirando meus bytes da gaveta e das memórias: minha e do micro. Após sua compreensão, você poderá incluir as 2 funções - ExtensoDeGrana( ) e AnalisaParte( ) - em seus aplicativos que necessitem escrever valores por extenso.*

### Usando a Função ExtensoDeGrana( ) em Outro Formulário

Só continue se ExtensoDeGrana( ) estiver funcionando perfeitamente.

### Adicionando Um Novo Formulário ao Projeto

1. Menu File | New Form | Ative Blank form | OK.
2. Mude a propriedade Name de Form1 para \_NovaUnit (Iniciando o nome do form com sublinhado, ele sempre aparecerá como o primeiro item no Inspetor de Objeto. Isto facilita muito quando o form tem muitos objetos.)
3. Clique o botão Save project | Escolha uma pasta para guardar seus fontes PAS, DFM e DPR.
4. Digite: NovaUnit (Para nome desse novo formulário.) | OK.

### Criando a Interface

5. Adicione ao formulário \_NovaUnit (figura 5) os seguintes objetos:
- | Objeto  | Tipo    | Name |
|---------|---------|------|
| TEdit   | Edit1   |      |
| TButton | Button1 |      |
| TLabel  | Label1  |      |

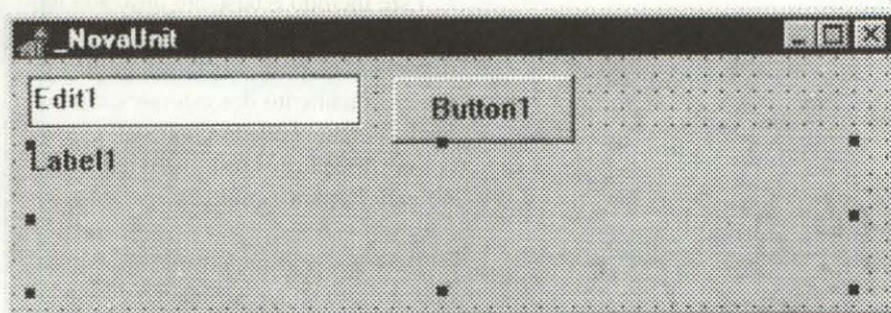
Figura 5

### Escrevendo o Código para Button1 do form \_NovaUnit, Evento Ao Clicar

1. Duplo clique em Button1.
2. Complete o manipulador de evento como segue:

```
procedure T_NovaUnit.Button1Click(Sender: TObject);
begin
    Label1.caption := ExtensoDeGrana(Edit1.Text);
    caption := FormatFloat('000,000,000,000,000,000',
    StrToFloat(Edit1.Text));
end;
```





## Informando ao form \_NovaUnit Onde Se Encontra a Função ExtensoDeGrana( )

3. Para que o compilador consiga "enxergar" onde se encontra a função ExtensoDeGrana( ) é preciso que você informe isto no código do form \_NovaUnit, digitando, ou seja, criando a cláusula **uses** logo após a diretiva de compilação da seção **implementation**:

```
implementation
{SR*.DFM;
uses VrtExt; {Você tem que informá-la;} {3}

procedure T_NovaUnit.Button1Click(Sender:
TObject);
begin
Label1.caption:=ExtensoDeGrana(Edit1.Text);
caption:=FormatFloat('000,000,000,000,00',
StrToFloat(Edit1.Text));
```

end;

O form NovaUnit está pronto.

## Preparando o form \_Teste para Enxergar e Trabalhar com o form \_NovaUnit

4. Ative a guia Vrect do form \_Teste.

5. Digite, ou seja, crie a cláusula **uses** logo após a diretiva de compilação da seção **implementation**:

```
implementation
{SR*.DFM;
uses NovaUnit; {Você tem que informá-la;}
...
```

6. Tecle F12 para visualizar o form \_Teste.

*Ricardo Flores é Editor de Micro Sistemas*

# COSMO

COLLEÇÃO

A MELHOR COLEÇÃO ESOTÉRICA EM CD-ROM

ICHING

RUNAS

MAPA ASTRAL

INTERPRETAÇÃO ASTROLÓGICA

HORÓSCOPO ASTECA

HORÓSCOPO CHINÊS

RELATÓRIO ASTROLÓGICO

NÚMEROS DA SORTE

ANJOS

VIDAS PASSADAS

FLORAS DE BACH

LIVRO DO DESTINO

ASTROLOGIA ELETIVA

BÚZIOS

CORTES DE CABELO PELA LUA

TAROT

**GNP**

(011) 9999-1757

CADASTRAMOS  
RE VENDAS

WINDOWS

PARA ADQUIRIR LIGUE PARA (011)9999-1757



## Matemática Computacional Empregada na Análise Estratégica das Empresas Parte 2

Por: Carlos A Thompson, Ph.D  
Marly Friaes (\*)

Eduardo Gaspar Porto(\*)  
(\*) Alunos do curso de Administração

*Dando prosseguimento ao artigo iniciado na Micro Sistemas 166 sobre análise e controle de estoques, alguns conceitos fundamentais são introduzidos e aplicados neste trabalho.*

*O Professor Thompson tem recebido inúmeras correspondências e gentilmente respondido.*

*Entretanto, solicitamos aos leitores interessados que, para adquirirem os artigos anteriores, dirijam-se à página com o anúncio "O Melhor da Informática Nacional".*

Este método é bastante utilizado na gestão de estoques, sendo um fator indispensável para o planejamento e dimensionamento dos estoques, com o qual o gerente pode orientar seus esforços em direção aos resultados mais significativos para a empresa.

Baseia-se nas conclusões de Pareto que, ao estudar a distribuição de renda entre a população do sistema econômico em que vivia, estabeleceu um princípio, no qual o maior segmento de renda nacional concentrava-se em uma pequena parcela da população, enquanto a maioria desta absorvia a menor parte da mesma renda.

Aproximadamente, na década de 50 em diante, com esforços iniciais da General Electric, é que o princípio de Pareto, com algumas adaptações, tem sido utilizado no universo de materiais, mais especificamente à gerência de estoques, com o nome de Classificação ABC.

O fundamento do método é aplicável a muitas situações onde seja possível estabelecer prioridades: uma tarefa a cumprir é mais importante que a outra; uma obrigação é mais significativa que a outra, de modo que a soma de algumas partes dessas tarefas ou obrigações de importância elevada representam, provavelmente, uma grande parcela das obrigações totais.

A Classificação ABC pode ser feita por: peso, volume, tempo de reposição, valor de demanda, inventário, custo unitário, aquisições realizadas e assim por diante.

Levando em consideração o Método ABC aplicado aos estoques, podemos afirmar que, na maioria dos casos, na maior parte das organizações, os estoques apresentam, em média, a seguinte distribuição:

Classe	Itens	Valor Total dos Estoques
A	10%	75%
B	25%	20%
C	65%	5%
Total	100%	100%

Tab. 2.1 - Decisão sobre os itens e valores do Método ABC



A distribuição apresentada pode variar de organização para organização. Esta estratificação (A, B e C), em qualquer empresa, com suas características de distribuição são sempre correlacionadas entre "quantidade x valor".

Os dados relativos às classes A, B e C podem ser representados por uma curva da seguinte forma:

**Erro! Argumento de opção desconhecido.** Figura 2.3 - Curva ABC, obtida dos Dados da Tabela 2.1

Assim, baseando-se neste método (tabela 2.1 e figura 2.3), verificamos que uma pequena economia obtida em relação aos materiais da classe A deverá ser mais significativa que uma grande redução de custos em materiais da classe C. Desta forma, as decisões do gerente de estoques deverão, sempre que possível, ser dirigidas aos benefícios decorrentes da boa gestão para os materiais da classe principal.

Desta forma, concluímos que, quanto maior o valor agregado aos itens e menor a sua quantidade em estoque, deveremos dar uma atenção especial a esta parcela do estoque (classe A); será nesta parcela onde haverá resultados mais significativos, em decorrência dos seus valores elevados.

Como exemplo, consideremos uma determinada concessionária onde se vendem-se carros e autopeças:

Classe A	Classe B	Classe C
carro	motor	volante

Tab. 2.2 - Classes do Método ABC

**Erro! Argumento de opção desconhecido.**

Figura 2.4 - Curva ABC correspondente às Classes ABC

Desta forma, verificamos que, se não houver uma atenção especial com a classe A (maiores valores e menores quantidades - carro), deixaremos de ter um "suposto - maior lucro" e/ou deixaremos de ganhar por falta de carros na agência.

Raciocínio (onde a perda ou lucro é mais significativa):

Onde teremos mais lucro ??? Vendendo um carro ou um volante e/ou motor ???

Onde deixaremos de ganhar mais ??? Não ter carros para vender para o cliente na agência ou não ter motores e/ou volantes ???

Para obtermos a curva ABC para estoques, deveremos fazê-la através de dois fatos geradores: inventário dos materiais existentes ou previsão de demanda para determinado exercício.

Nos dois casos, a metodologia a ser utilizada deverá ser a mesma, conforme o seguinte roteiro:

## 1. Obtenção de relação dos itens com seus valores totais correspondentes;

Pode ser feito a partir do inventário físico ou da previsão de demanda. Multiplicamos a quantidade de cada item por seu custo unitário e obtemos o valor da existência ou do consumo provável. Este processo poderá eventualmente ser empregado para grupos de materiais, em vez de itens, na medida em que aqueles representem materiais com grande afinidade de consumo e valor, e o relacionamento item a item se torne respectivo e oneroso.

Porém, o procedimento usual é voltado para o levantamento por item de material.

## 2. Tabulação;

Com os dados da relação colocados em ordem decrescente de valor, elaboramos uma tabela que fornecerá os percentuais de itens e valores correspondentes para a obtenção da curva.

## 3. Elaboração da curva; e

Os pares de dados referentes à percentagem acumulada em quantidade de itens e valores são plotados um a um, ou em intervalos representativos da curva formada. Os pontos assim encontrados permitem o traçado da curva ABC, conforme figura 2.4, p. 37 (exemplo carro, motor e volante).

## 4. Separação em classes.

Será feita com base no princípio de que a classe A deverá abranger o menor número de itens correspondente ao maior valor possível. A relação média de 10 % dos itens para 75 % em valor é apenas um ponto de referência (vide tabela 2.1, p. 36), pois caberá ao gerente de estoques definir os intervalos de sua conveniência.

De um modo geral, podemos identificar na curva ABC dois pontos de "inflexão" bem nítidos, que deverão corresponder aos limites de A para B e de B para C.

Na realidade, não há uma regra única e definitiva para se conseguir a melhor separação de classes. O importante é não esquecer que materiais sujeitos a estoque distribuem-se nitidamente quanto à relação item x valor de consumo, em parcelas (A, B e C) com características muito peculiares. Cabe ao gerente explorar todas as possibilidades dessa estratificação, para aumentar a razão custo x benefício em sua empresa.

## Determinação dos Pontos de Separação entre as Classes A, B e C

Em estoques de muitos itens, as percentagens estão tão próximas umas das outras, que se torna difícil, sem incorrer em boa margem de erro, a separação das classes.

O processo gráfico, ilustrado na figura abaixo, tem por base o seguinte roteiro, como sugestão:

**Erro! Argumento de opção desconhecido.** Figura 2.5 - Determinação dos Pontos sobre a Curva ABC

1. Traçado de dois eixos de coordenadas com escalas iguais representando as percentagens acumuladas em itens e valores;

2. Traçado da curva, dadas percentagens acumuladas;

3. União por um reta dos pontos inicial e final da curva (O e Z);



4. Traçado de uma tangente à curva, paralela à reta OZ, obtendo-se então dois pontos X e Y;

5. Traçado das bissetrizes dos ângulos OXY e XYZ; e

6. Onde as bissetrizes cortarem a curva, estarão os pontos de separação das classes A, B e C.

**Erro! Argumento de opção desconhecido.**  
Figura 2.6 - Método para se obter com mais Precisão as Classes ABC

## Análise Probabilística

### Análise das Séries Temporais

#### a) Média Móvel

A média móvel é uma aritmética calculada período a período, de uma série de dados, substituindo-se, a cada período, o dado mais antigo da série pelo mais recente, mantendo-se constante o número de dados da série. O conceito pode ser representado matematicamente da seguinte forma:

$$x = \frac{r_t + r_{t-1} + r_{t-2} + \dots + r_{t-n} + 2 + r_{t-n+1}}{n}$$

$$x = \frac{\sum_{i=t-n+1}^{t-n+1} r_i}{n}$$

onde:

$r_t$  - demanda ( ou consumo ) mais recente  
 $r_{t-1}$  - demanda imediatamente anterior  
 $r_{t-n+1}$  - demanda mais antiga para o "n" estabelecido  
n - número de dados da série, estabelecido previamente

O principal problema quando da adoção do método da média móvel reside na escolha do valor de "n" mais adequado. Quanto maior for seu valor, menor será a reação da inclusão/exclusão de dados sobre a nova média. Conseqüentemente acontecerá o inverso, se considerarmos um valor pequeno para "n".

#### b) Média Móvel Ponderada Exponencialmente (MMPE)

O método da média móvel ponderada exponencialmente é baseado no ajustamento de período a período da última média prevista ( $r_{t-1}$ ) por adição ou subtração de uma fração a da diferença entre a demanda atual no período presente ( $r_t$ ) e a última média prevista. Essa fração a representa o quanto dessa diferença o administrador reconhece como tendência e o quanto ele considera desvio ocasional. A experiência diz que esse  $\alpha$  é mais adequado entre 0,1 e 0,4, entretanto esse valor será determinado pela análise histórica das demandas, comparando os valores que seriam previstos para cada  $\alpha$  e o real comportamento da demanda. Enfim o resultado vai representar a nova média prevista no período atual ( $r^*$ ), que é válida como previsão para o próximo período.

$$r^* = \alpha r_t + (1 - \alpha)r_{t-1}$$

O que não é explícito nesta representação matemática é o fato de que, assim como a nova média prevista é baseada no mês anterior, a média prevista no mês anterior também baseada no seu antecessor, e assim sucessivamente. Agora nos resta não apenas apreciar a equação geral desse procedimento, mais também demonstrar como o mecanismo pode funcionar se não há previsão antecedente que sirva de base para o cálculo da primeira previsão.

Assim:

$$r^* = \alpha r_t + (1 - \alpha)r_{t-1}$$

induz

$$r_{t-1}^* = \alpha r_{t-1} + (1 - \alpha)r_{t-2}^*$$

substituindo

$$r^* = \alpha r_t + (1 - \alpha)\alpha r_{t-1} + (1 - \alpha)^2 r_{t-2}^*$$

por outro lado,

$$r_{t-2}^* = \alpha r_{t-2} + (1 - \alpha)r_{t-3}^*$$

Substituindo

$$r^* = \alpha r_t + \alpha(1 - \alpha)r_{t-1} + (1 - \alpha)[\alpha r_{t-2} + (1 - \alpha)r_{t-3}^*]$$

$$r^* = \alpha r_t + \alpha(1 - \alpha)r_{t-1} + \alpha(1 - \alpha)^2 r_{t-2}^* + (1 - \alpha)^3 r_{t-3}^*$$

Assim, para "n" valores da demanda, pode-se inferir que,

$$r^* = \alpha r_t + \alpha(1 - \alpha)r_{t-1} + \alpha(1 - \alpha)^2 r_{t-2}^* + \dots + (1 - \alpha)^{n-1} r_{t-n+1}^*$$

Como  $0 \leq \alpha \leq 1$  e para "n" um valor elevado, tem-se:

$$(1 - \alpha)^{n+1} r_{t-(n+1)}^* \approx 0$$

Logo

$$r^* = \alpha r_t + \alpha(1 - \alpha)r_{t-1} + \alpha(1 - \alpha)^2 r_{t-2}^* + \dots + \alpha(1 - \alpha)^{n-1} r_{t-n+1}^*$$

Dados da tabela 2.3 e a respectiva figura 2.7 são mostrados na figura 2.9, p. 46 obtida pelo emprego do método dos mínimos quadrados. Esses mesmos dados de frequência de semanas onde a demanda ocorre são, também aproximados pelo métodos da média móvel e da média móvel ponderada exponencialmente para  $\alpha$  igual a 0,5, conforme a figura 2.7 A figura mostra o efeito do parâmetro  $\alpha$  sobre os valores estimados, calculados pela média móvel ponderada exponencialmente.

$$\alpha = 0,5 \quad ; \quad r^* = 12$$

t	r	r*
1	10	5
2	5	5
3	8	6,5
4	20	13,25

Tab. 2.3 - Dados para Cálculo da MMPE

$$r^*(4) = \alpha r(4) + \alpha(1 - \alpha)r(3) + \alpha(1 - \alpha)^2 r(2) + \alpha(1 - \alpha)^3 r(1)$$

$$= 0,5 (20) + (0,5) (1 - 0,5)(8) + (0,5) (1 - 0,5)^2(5) + (0,5) (1 - 0,5)^3 (10)$$

$$= 10 + 2 + 0,625 + 0,625 = 13,25$$

$$r^*(4) = \alpha r(4) + (1 - \alpha)r^*(3) \therefore r^*(3) = \frac{r^*(4) - \alpha r(4)}{(1 - \alpha)}$$

$$= \frac{13,25 - (0,5) (20)}{1 - 0,5} = 6,5$$

$$r^*(2) = \frac{r^*(3) - \alpha r(3)}{(1 - \alpha)} = \frac{6,5 - (0,5) (8)}{1 - 0,5} = 5$$



$$r^*(1) = \frac{r^*(2) - \alpha r(2)}{(1 - \alpha)} = \frac{5 - (0,5)(5)}{1 - 0,5} = 5$$

$$r^*(0) = \frac{r^*(1) - \alpha r(1)}{(1 - \alpha)} = \frac{5 - (0,5)(10)}{1 - 0,5} = 0$$

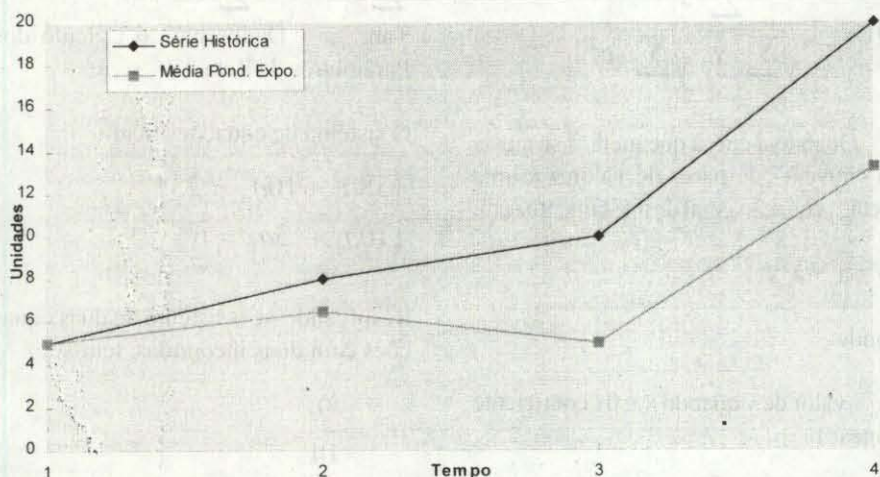


Figura 2.7 - Distribuição da Demanda Inicial e da MMPE

Podemos agora concluir que para  $n$  períodos a nossa expressão seria assim:

$$r_t^* = \mu r_1 + \mu(1 - \mu)r_{t-1} + \mu(1 - \mu)^2 r_{t-2} + \dots + \mu(1 - \mu)^{n-1} r_{t-(n-1)}$$

onde a última parcela da relação pode ser abandonada. Observe que o fator

$r_{t-(n-1)}^*$  é multiplicado por  $\mu(1 - \mu)^{n-1}$ , que, para  $n$  elevado, pode ser descartado. Assim,

$$r_t^* = \mu r_1 + \mu(1 - \mu)r_{t-1} + \mu(1 - \mu)^2 r_{t-2} + \dots +$$

Obtido  $r_t^*$ , os valores de  $r_{t-1}$ ,  $r_{t-2}$ , ... podem ser calculados recursivamente.

### c) Média Móvel Ponderada Exponencialmente: Dupla Filtragem (MDF)

A filtragem dos dados com altas frequências pode ser mais precisa empregando-se um duplo ajustamento como é desenvolvido a seguir:

Obtida a série temporal pela primeira filtragem, empregando-se a média móvel ponderada exponencialmente, defi-

ne-se tendência aparente  $Dx_t$  por,  $Dx_t = x(t) - x(t-1)$

Onde a primeira filtragem  $r_t^*$  (calculado pela média móvel ponderada exponencialmente) é representada por  $x(t)$ . Por outro lado, por recorrência, escreve-se

$$Dx_{t-1} = x(t-1) - x(t-2)$$

etc.

Analogamente ao que foi feito para uma filtragem, tem-se como valor estimado da tendência aparente no instante  $t$  o parâmetro  $Dx_t^*$  dado por,  $Dx_t^* = a Dx_t + (1 - a) Dx_{t-1}$

Fazendo-se um segundo ajuste para a variação da demanda, tem-se,

$$\Delta^{**}_t x = \left( \frac{1 - \alpha}{\alpha} \right) \Delta^*_t$$

Portanto, a demanda esperada  $d$  no instante  $t$ , após duas filtrações, é dada por

$$d(t) = x(t) + \left( \frac{1 - \alpha}{\alpha} \right) [a(x(t) - x(t-1)) + (1 - a)(x(t-1) - x(t-2))]$$

Dados:

$$\alpha = 0,5 \quad \text{e} \quad \frac{1 - \alpha}{\alpha} = 1$$

t	r	r*	TA	T <sub>t</sub>	d_e
0		0		0	
1	10	5	5	$a(5) + (1-a)0 = 2,5$	$5 + (1)(2,5) = 7,5$
2	5	5	0	$a(0) + (1-a)(5) = 2,5$	$5 + 2,5 = 7,5$
3	8	6,5	1,5	$a(1,5) + (1-a)(0) = 0,75$	$6,5 + 0,75 = 7,25$
4	20	13,25	6,75	$a(6,75) + (1-a)(1,5) = 4,125$	$13,25 + 4,125 = 17,375$

Tab. 2.4 - Dados para o Cálculo do Duplo Ajuste ou Dupla Filtragem

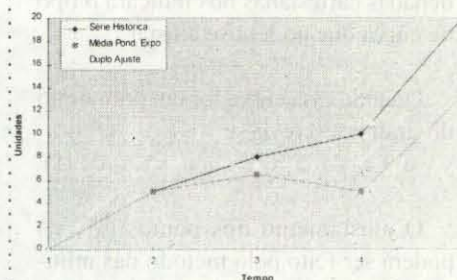


Figura 2.8 - Distribuição da Demanda Inicial, da MMPE e da Dupla Filtragem.

### d) Mínimos Quadrados [apêndice B]

Este método já tem um campo específico de emprego, diferente do da média móvel ou da média móvel ponderada exponencialmente. É mais voltado para as projeções a médio e longo prazos, onde se pode admitir um nível maior de erro. Além disso, há necessidade de um mínimo de correlação entre os dados da série histórica para que este método apresente resultados razoáveis. Nas demandas afetadas por variáveis aleatórias ou por sazonalidade, seu emprego é pouco eficaz.

O método dos mínimos quadrados é largamente utilizado quando se deseja estimar uma tendência em função de duas variáveis:  $x$  (ano, semestre, trimestre, mês, semana, dia etc.) e  $y$  (demanda ou ocorrência correspondente).

As variáveis  $x$  e  $y$  são também conhecidas, respectivamente, como variável independente e variável dependente.

O método de ajustamento tem por objetivo determinar uma curva chamada linha de regressão, capaz de representar o valor médio provável de uma das variáveis em função dos valores da outra.



Inicialmente, a distribuição dos pares de valores  $x_i$  e  $y_i$  nos eixos de coordenadas cartesianas nos indicará o tipo de curva que poderá se ajustar.

Quando essa curva for um polinômio do grau  $n$ , escreve-se

$$y = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n \quad (1)$$

O ajustamento dos pontos  $(x_i, y_i)$  poderá ser feito pelo método dos mínimos quadrados, que é baseado no seguinte princípio:

"A soma dos quadrados dos desvios deve ser um mínimo para o tipo de curva ajustada, isto é, deve ser menor para a curva ajustada do que para qualquer outra curva de mesma equação geral."

**Erro! Argumento de opção desconhecido.**  
Figura 2.9 - Ajuste da Reta

De todas as curvas que se ajustam a um conjunto de pontos  $(y_i, x_i)$  até  $(y_n, x_n)$  a que apresenta o mínimo valor de  $d_1^2 + d_2^2 + \dots + d_n^2$  é considerada a melhor curva de ajustamento.

Para a previsão de demandas (consumo), o emprego dos mínimos quadrados é adequado, principalmente, quando se deseja projetar um consumo provável para um período além do imediato (extrapolação). Assim, à medida que a projeção for calculada para períodos mais distantes, maior será o erro de previsão que se incorrerá. O método também é adequado para interpolações, ou seja, para se obter valores contínuos da variável dependente  $y$  para abscissas entre  $x_i$  e  $x_n$ .

Na prática, para escolhermos o tipo de curva que se ajusta a uma certa distribuição (reta, parábola de 2º grau, de 3º, etc.), registramos os pares de valores  $x$  e  $y$  em papel milimetrado e, de acordo com o tipo de "nuvem" obtido, decidimos pelo tipo mais adequado de curva. Hoje em dia, com os avanços na área computacional, ajusta-se vários polinômios e decide-se pelo menor dos mínimos  $d^2$ . Assim, obtém-se o quadrado dos desvios para diferentes tipos de curva e decide-se por aquela que apresente o menor desvio médio absoluto.

O pré-requisito para utilização dos mínimos quadrados é a existência de um coeficiente de correlação razoável entre  $x$  e  $y$  (acima de 60 por cento). Define-se coeficiente de correlação entre dois conjuntos de variáveis,

$$\rho_{xy} = \frac{\sum x_i y_i}{\sqrt{\sum x_i^2} \sqrt{\sum y_i^2}}$$

Quando a curva que melhor se ajusta à "nuvem" de pares de valores é uma reta, a equação geral de ajustamento será:

$$y = a_0 + a_1 x$$

onde:

$a_0$  = valor de  $y$  quando  $x = 0$  (coeficiente linear)

$a_1$  = coeficiente angular

A equação  $y = a_0 + a_1 x$ , que representa a tendência de  $n$  pares de valores, é obtida determinando-se os parâmetros  $a_0$  e  $a_1$  através de suas equações normais.

$$d^2 = \sum_{i=1}^n (y_i - y)^2 = \sum_{i=1}^n y_i^2 - 2y_i y + y^2$$

Substituindo-se a função  $y$ , obtém-se:

$$\begin{aligned} d^2 &= \sum_{i=1}^n y_i^2 - 2 \sum_{i=1}^n y_i (a_0 + a_1 x_i) + \sum_{i=1}^n (a_0 + a_1 x_i)^2 \\ \frac{\partial d^2}{\partial a_0} &= -2 \sum_{i=1}^n y_i + \sum_{i=1}^n 2(a_0 + a_1 x_i) = 0 \\ \frac{\partial d^2}{\partial a_1} &= -2 \sum_{i=1}^n x_i y_i + \sum_{i=1}^n 2x_i (a_0 + a_1 x_i) = 0 \end{aligned}$$

Reagrupando convenientemente os termos, tem-se, para o caso da reta,

$$\begin{aligned} \sum y &= na_0 + a_1 \sum x \\ \sum xy &= a_0 \sum x + a_1 \sum x^2 \end{aligned}$$

Como os coeficientes  $a_0$  e  $a_1$  são as incógnitas, resolve-se o sistema de duas equações onde são conhecidos, o número de pontos  $n$  e os somatórios

$$\sum x, \sum y, \sum x^2, \sum xy$$

Como exemplo, a melhor reta que se ajusta aos pontos (2,1), (3,2) e (5,3), é obtida conforme a tabela.

x	y	x <sup>2</sup>	xy
2	1	4	2
3	2	9	6
5	3	25	15

$$\sum x = 10 \quad \sum y = 6$$

$$\sum x^2 = 38 \quad \sum xy = 23$$

Tab. 2.5 - Dados para o Cálculo dos Parâmetros da Reta

O sistema de equações fica:

$$\begin{cases} 38a_1 + 10a_0 = 23 \\ 10a_1 + 3a_0 = 6 \end{cases}$$

Resolvendo-se o sistema de duas equações com duas incógnitas, tem-se:

$$a_1 = \frac{9}{14}$$

$$a_0 = -\frac{1}{7}$$

Logo a reta fica:

$$y = a_1 x + a_0 = \frac{9x}{14} - \frac{1}{7}$$

Graficamente, eis a solução:

**Erro! Argumento de opção desconhecido.**  
Fig. 2.10 - Ajuste da Reta pelo Método dos Mínimos Quadrados

Observe que o coeficiente de correlação, neste exemplo, é,

$$\rho_{xy} = \frac{\sum x_i y_i}{\sqrt{\sum x_i^2} \sqrt{\sum y_i^2}} = \frac{23}{\sqrt{38} \sqrt{14}} = \frac{23}{23,048} = 0,998$$

Portanto, pelo coeficiente de correlação de 99,8% (> 60%), o método dos mínimos quadrados foi adequado.

**Dr. Carlos Thompson** é Prof. Titular do Centro Universitário Augusto Motta, Ph.D pela Universidade de Houston, USA e Prof. da Universidade Federal Fluminense no curso de Mestrado.  
Home Page:  
<http://home.openlink.com.br/cthompson>  
e-mail:  
[cthompson@openlink.com.br](mailto:cthompson@openlink.com.br)



# APRENDA INFORMÁTICA SEM SAIR DE CASA COM RICARDO FLORES

## CADA CURSO É COMPOSTO DE:

- Cronograma de aula a ser seguido.
- Apostila pelo Método Tutorial "Passo a Passo" para você treinar cada exercício diretamente em seu PC.
- Disquete contendo os exercícios prontos do Tutorial, para você tirar suas dúvidas.

Mais informações? Solicite pelo correio Convencional ou visite minha Home Page no endereço  
[HTTP://W3.OPENLINK.COM.BR/RICARDOFLORES](http://W3.OPENLINK.COM.BR/RICARDOFLORES)

## Desejo receber por carta registrada, os cursos:

CURSO	De R\$	Por R\$	Qde.	R\$
Introdução à Informática até o MS/DOS 6.2	38,00	22,80	x	
Windows 3.1	44,00	26,40	x	
<b>Para Windows 3.1, 3.11 ou 95 ou Cairo:</b>				
Paintbrush dos 8 aos 80 Anos	27,00	16,20	x	
CorelDRAW! 5.0	47,00	28,20	x	
Programação em Visual Basic 16/32 Bits	44,00	26,40	x	
Programação em Delphi 16/32 Bits com SQL	58,00	35,00	x	
HTML - Criando sua Home Page na Internet	27,20	16,30	x	
Word 6.0/7.0 - Editor de Texto	44,00	26,40	x	
Excel 5.0/7.0 - Planilha de Cálculo e Gráficos	45,00	27,00	x	
<b>Para DOS:</b>				
dBase III Plus - Banco de Dados Interativo	27,00	16,20	x	
Programação em dBase III Plus	27,00	16,20	x	
Programação em Clipper 5.01 / 02 (Básico)	35,00	21,00	x	
<b>TOTAL DO PEDIDO:</b>				

**Não perca!**

**Promoção Arrasadora!**

**Desconto de 40%**

**Aproveite! Faça já seu pedido!**

### Pagamento Via Banco

Depósito em numerário a favor de Audit System Serviços Ltda.  
 Banco Itaú, agência 0934 Vila Isabel, Rio de Janeiro, C/C 22.142-5  
 Ligue ou passe um Fax para (021) 571-5903 - Ricardo Flores, informando:  
 seu nome, endereço completo e confirmando o crédito no valor TOTAL DO PEDIDO,  
**já incluídas as despesas postais.**  
 Prazo de Entrega: **Imediato.**

e-mail  
[ricardoflores@openlink.com.br](mailto:ricardoflores@openlink.com.br)

### Pagamento Via Correio

Cheque cruzado e nominal à Audit System Serviços Ltda.  
 Envie para CAIXA POSTAL 25.096 - RIO, CEP 20.552-970,  
 no valor TOTAL DO PEDIDO, **já incluídas as despesas postais.**  
 Prazo de Entrega: **Após a compensação do cheque.**

Nome:	Fone:	
Endereço:	Bairro:	
Cidade:	Estado:	CEP:
Data:	Assinatura:	



# ARTIGO

## Proteção ..... Máxima ..... Para o Seu ..... Hard Disk .....

Por: Rubem Nascimento da Rocha

*Veja com Rubem Nascimento da Rocha,  
como obter a proteção máxima para seus  
arquivos gravados no Hard Disk.*

Algumas vezes, os usuários sentem necessidade de proteger a sua máquina de bisbilhoteiros, principalmente quando esses usuários se ausentam por algum tempo (dias, semanas, meses, etc.) de suas preciosas máquinas, estando as mesmas a mercê daquele irmãozinho que está aprendendo Windows 95 no curso de informática.

Sendo assim, apresento aos leitores de Micro Sistemas uma solução simples e segura para garantir a integridade de seus arquivos. Trata-se de um pequeno utilitário para travar e destravar o disco rígido. Esta solução foi escrita em Turbo Pascal (De novo Turbo Pascal, Rubem?), podendo ser adaptada para outras linguagens (C, C++, Assembly, etc.) sem problemas.

### *Travando e Destravando Seu Disco Rígido*

Antes de mostrar o modo de funcionamento deste utilitário de travamento e destravamento do disco rígido, é necessário que sejam esclarecidos alguns conhecimentos básicos para entendermos o funcionamento do programa.

Quando o FDISK faz partições em um disco rígido, ele armazena informações numa tabela de 64 bytes, chamada *tabela de partição mestre*. A tabela começa no deslocamento 1BEh no início do primeiro setor e contém quatro entradas de 16 bytes, correspondendo cada uma a uma partição no disco rígido. O quinto byte de cada entrada, chamado *byte identificador de sistema*, identifica o sistema operacional dono da partição correspondente. O valor 0 significa uma partição não usada; qualquer outro valor é um identificador de sistema operacional. Os valores que este byte identificador de sistema pode assumir são os seguintes:

- 00h** - Sistema Operacional Desconhecido;
- 01h** - MS-DOS com FAT de 12 Bits, Números de Setores Com 16 Bits;
- 02h** - XENIX (O Unix da Microsoft);
- 04h** - MS-DOS com FAT de 16 Bits, Números de Setores Com 16 Bits;
- 05h** - Partição Estendida do MS-DOS (versões 3.3 e posteriores);
- 06h** - MS-DOS 4.0 (Compaq 3.31), Número de Setores Com 32 Bits;



51h - Partição Estendida Ontrack;  
64h - Partição Novell;  
75h - PCIX;  
DBh - CP/M  
FFh - BBT

Fonte: Ralf Brown Interrupt List,  
Version 5.2A

O MS-DOS (e também o Windows 95) utiliza os valores 1, 4 e 6 para especificar o tamanho de suas partições, onde 1 significa menos de 16 MB, 4 indica de 16 a 32 MB, e 6 identifica uma partição grande (mais de 32 MB). Partições ampliadas do MS-DOS - que contém acionadores lógicos de discos não definidos na tabela de partição - são indicadas por 5.

Portanto, o que nosso utilitário fará é simplesmente reescrever o setor de boot do disco rígido com valores zerados para todas as tabelas de partição do disco rígido. Sempre antes de

efetuar o bloqueio, o programa pode salvar as informações das tabelas de partição em um arquivo binário para posterior recuperação.

É importante observar alguns pontos quanto ao uso deste utilitário. Ele funcionará perfeitamente em micros sem ou com discos compactados, sejam eles com o DriveSpace, DoubleSpace ou Stackler. Este programa foi testado em máquinas com sistema MS-DOS e/ou Windows 95, com/sem softwares de compactação de disco. Apesar do programa salvar as informações das partições, é sempre bom manter um backup do seu disco rígido e preparar um disco de emergência para restaurar o disco ao seu estado original. Tal disco de emergência poderá ser criado através de utilitários como o Norton Utilities, que possuem um programa para geração de discos de resgate (Rescue

Disk) contendo informações vitais da máquina, como listagens do CONFIG.SYS e AUTOEXEC.BAT, informações sobre o setor de boot e partições do disco rígido, informações da CMOS, etc.

Com esta super dica, espero que os leitores de Micro Sistemas se sintam mais seguros em deixarem suas máquinas livres de curiosos e assimilados que, de vez em quando, nos deixam altas horas da madrugada para recuperar dias, semanas e meses de trabalho. E também peço desculpas pelas mensagens do programa estarem em inglês. É que eu já estou distribuindo o programa na Web. Até a próxima!

**RUBEM NASCIMENTO DA ROCHA**  
tem 21 anos e é programador em diversas linguagens. Para eventuais contatos: [rubemrocha@netium.com.br](mailto:rubemrocha@netium.com.br)

```
== HDLOCK.PAS =====
{ HDLOCK.PAS - Hard Drive Lock/Unlock Utility
  Author:      Rubem Nascimento da Rocha
  E-mail:      rubemrocha@netium.com.br
  This program will include new features, however, in a new
  registered release. Further more details, contact the author }

{$R-,V-,L-,D-}
{$M 2048,0,0}

Uses Crt, Dos;

Const
  HeaderFile : String = 'HDLOCK Image File (r) Cybercop Software' + #26;

Type{ Estrutura de dados para manipular um registro de partição }
  PartitionTableInfo = Record
    BootIndicator,
    StartHead, StartSector, StartTrack,
    SystemIndicator,
    EndHead, EndSector, EndTrack : Byte;
    PreSectors, Length : LongInt;
  End;

{ Estrutura de dados para manipular todos os registros }
  HardDrivePartn = Array [ 1..4 ] Of PartitionTableInfo;
{ Estrutura de dados para manipular os dados do setor de boot }
  BootSectorInfo = Record
    BootstrapCode : Array [ 1..446 ] Of Byte;
    PartitionRecord : HardDrivePartn;
    Signature : Word;
  End;

{ Tipos de operações a serem realizadas no setor de boot }
  BootOperations = ( BootRead, BootWrite );

Var Lock, Unlock : Boolean; { Variáveis públicas do programa }
```



# ARTIGO / Proteção Máxima para o seu Hard Disk

```
FileName : PathStr;
Regs      : Registers;
BootSector : BootSectorInfo;
HardDrive : Char;
StrParam   : String;
Parm       : Byte;

{ Envia uma mensagem (de erro ou não), e termina o programa }
Procedure MsgExit( Msg: String; Error: Boolean );
Begin
  If Error Then Write( #7, #7, #7 );
  WriteLn( Msg );
  Halt;
End;

{ Verifica se um determinado arquivo já existe }
Function FileExist( Name: PathStr ) : Boolean;
Var Search : SearchRec;
Begin
  FindFirst( Name, AnyFile, Search );
  FileExist := ( DosError = 0 );
End;

{ Efetua a entrada de uma resposta Sim ou Não }
Function YesOrNo : Char;
Var Result : Char;
Begin
  Repeat Result := UpCase( ReadKey ) Until Result In [ #13, #27, 'Y', 'N' ];
  If Result = #13 Then Result := 'Y' Else If Result = #27 Then Result := 'N';
  WriteLn( Result );
  YesOrNo := Result;
End;

{ Efetua a leitura/escrita do setor de boot }
Function BootDisk( Operation: BootOperations ) : Boolean;
Var Result : Boolean;
    Drive  : Byte;
Begin
  Drive := $80 + ( Ord( HardDrive ) - Ord( 'C' ) );
  Result := ( Drive >= $80 );
  If Result Then With Regs Do Begin
    AH := 0;
    DL := Drive;
    Intr( $13, Regs );
    Result := ( Flags AND FCarry = 0 );
    If Result Then Begin
      AH := 2 + Ord( Operation ); { 2-leitura/3-escrita }
      AL := 1;                   { quantidade de setores a ler/gravar }
      CH := 0;                   { trilha }
      CL := 1;                   { setor }
      DH := 0;                   { face }
      DL := Drive;               { unidade de disco }
      ES := Seg( BootSector );   { segmento e ... }
      BX := Ofs( BootSector );   { offset do buffer de leitura/escrita }
      Intr( $13, Regs );         { executa a operação }
      Result := ( Flags AND FCarry = 0 );
      If Not Result Then Begin
        AH := 0;
        DL := Drive;
        Intr( $13, Regs );
      End;
    End;
  End;
  BootDisk := Result;
End;

{ Salva a tabela de partição do HD escolhido num arquivo }
Function SaveHDIImage : Boolean;
Var ImageFile : File;
    Confirm    : Char;
    PartnBuffer : HardDrivePartn;
    Written     : Word;
```



```
{ Elimina definitivamente o arquivo de imagem do HD }
Procedure DestroyImage;
Begin
  Close( ImageFile ); Rewrite( ImageFile, 1 ); Close( ImageFile );
  Erase( ImageFile );
End;

Begin
  If Length( FileName ) = 0 Then Begin
    WriteLn( 'WARNING !' );
    WriteLn;
    WriteLn( 'This hard drive lock option was especificied, but a' );
    WriteLn( 'file name was not. Before to lock the hard drive,' );
    WriteLn( 'it can be make a hard drive partition table save,' );
    WriteLn( 'for unlock it with this program.' );
    WriteLn;
    Write( 'Enter a file name or ENTER for nothing: ' );
    ReadLn( FileName );
  End;
  If Length( FileName ) = 0 Then Begin
    SaveHDIImage := False;
    Exit;
  End;

  If FileExist( FileName ) Then Begin
    WriteLn( 'Hard drive image file was already exist !' );
    Write( 'Overwrite it? [Y/N] ' );
    Confirm := YesOrNo;
    If Confirm = 'N' Then Begin
      SaveHDIImage := False;
      Exit;
    End;
  End;

  { Obtendo e salvando os dados da tabela de partição do HD informado }
  WriteLn( 'Saving hard drive partition table image ...' );
  WriteLn;
  If BootDisk( BootRead ) Then Begin
    Assign( ImageFile, FileName );
    {$I-}
    Rewrite( ImageFile, 1 );
    {$I+}
    If IOResult = 0 Then Begin
      {$I-}
      BlockWrite( ImageFile, HeaderFile, SizeOf( HeaderFile ), Written );
      {$I+}
      If ( IOResult = 0 ) AND ( Written = SizeOf( HeaderFile ) ) Then Begin
        PartnBuffer := BootSector.PartitionRecord;
        {$I-}
        BlockWrite( ImageFile, PartnBuffer, SizeOf( PartnBuffer ), Written );
        {$I+}
        If ( IOResult = 0 ) AND ( Written = SizeOf( PartnBuffer ) ) Then
          Close( ImageFile )
        Else Begin
          DestroyImage;
          MsgExit( 'Error writing image on file !', True );
        End;
      End Else Begin
        DestroyImage;
        MsgExit( 'Error writing image file header !', True );
      End;
    End Else Begin
      DestroyImage;
      MsgExit( 'Error creating image file !', True );
    End;
  End Else
    MsgExit( 'Invalid or Inexistent Hard Drive !', True );
  SaveHDIImage := True;
End;

{ Restaura o arquivo de imagem }
Function RestoreHDIImage : Boolean;
```



```

Var MirrorFile      : File;
PartReaded          : HardDrivePartn;
Partn               : Byte;
Result              : Boolean;
HeaderReaded        : String;
Readed              : Word;

Begin
  If Length( FileName ) = 0 Then Begin
    Write( 'Name for hard drive image: ' );
    ReadLn( FileName );
    Result := ( Length( FileName ) > 0 );
    If Not Result Then Begin
      WriteLn( 'Image file was not specified !' + CHR( 13 ) +
        'Unlock aborted !' );
      RestoreHDIImage := Result;
      Exit;
    End;
  End;

Assign( MirrorFile, FileName );
{$I-}
Reset( MirrorFile, 1 );
{$I+}
Result := ( IOResult = 0 );
If Result Then With BootSector Do Begin
  HeaderReaded[ 0 ] := Chr( Length( HeaderFile ) );
  {$I-}
  BlockRead( MirrorFile, HeaderReaded, SizeOf( HeaderFile ), Readed );
  {$I+}
  Result := ( ( IOResult = 0 ) AND ( Readed = SizeOf( HeaderFile ) ) );
  If Result Then Begin
    Result := ( HeaderReaded = HeaderFile );
    If Result Then Begin
      {$I-}
      BlockRead( MirrorFile, PartReaded, SizeOf( PartReaded ), Readed );
      {$I+}
      Result := ( ( IOResult = 0 ) AND
        ( Readed = SizeOf( PartReaded ) ) );
      If Result Then Begin
        Close( MirrorFile );
        Result := BootDisk( BootRead );
        If Result Then
          For Partn := 1 To 4 Do
            PartitionRecord[ Partn ].SystemIndicator :=
              PartReaded[ Partn ].SystemIndicator
            Else WriteLn( 'Error writing information on hard drive !' );
            End Else WriteLn( 'Error reading file information !' );
            End Else WriteLn( 'Invalid image file header !' );
            End Else WriteLn( 'Error reading header file !' );
          End;
          RestoreHDIImage := Result;
        End;
      { Efetua o bloqueio do disco rígido }
      Procedure HDLock;
      Var Partn      : Byte;
          Confirma   : Char;
      Begin
        If SaveHDIImage Then With BootSector Do
          For Partn := 1 To 4 Do PartitionRecord[ Partn ].SystemIndicator := 0
          Else If Not BootDisk( BootRead ) Then
            MsgExit( 'Invalid or inexistent hard drive !', True )
          Else For Partn := 1 To 4 Do
            BootSector.PartitionRecord[ Partn ].SystemIndicator := 0;

          Write( 'Do you wish to continue with the hard drive lock? [Y/N] ' );
          Confirma := YesOrNo;
          If Confirma = 'Y' Then
            If BootDisk( BootWrite ) Then Begin
              WriteLn;
              WriteLn( 'Hard Drive Locked !' );
              WriteLn( 'Press Any Key To Reset The System ...' );
            End;
          End;
        End;
      End;
    End;
  End;

```



```
    If ReadKey = #0 Then;
    Inline( $EA/$0000/$FFFF );
    End Else MsgExit( 'Error on hard drive lock !', True );
End;

{ Desbloqueia o disco rígido com a informação no arquivo de imagem }
Procedure HDUnlock;
Begin
    If RestoreHDIImage Then Begin
        If BootDisk( BootWrite ) Then Begin
            WriteLn( 'Hard Drive Unlocked !' );
            WriteLn( 'Press Any Key To Reset The System ...' );
            If ReadKey = #0 Then;
                ClrScr;
                Inline( $CD/$19 );
            End Else MsgExit( 'Error on hard drive unlock !', True );
            End Else MsgExit( 'Impossible to unlock hard drive !', True );
        End;
    }
    { Converte uma string Pascal para maiúsculas }
    Function Upper( St: String ) : String;
    Var N, Tam : Byte;
    Begin
        Tam := Length( St );
        For N := 1 To Tam Do St[ N ] := UpCase( St[ N ] );
        Upper := St;
    End;

    { Programa Principal }
    Begin
        WriteLn( 'HDLCK (c) Hard Drive Lock/Unlock Utility, vl.0' );
        WriteLn( 'Copyright 1997, by Rubem Nascimento da Rocha' );
        WriteLn( 'E-mail: cybercop@heniq.com.br / rubemrocha@netium.com.br' );
        WriteLn;
        { Se nenhum parâmetro foi passado, mostra tela de ajuda }
        If ParamCount = 0 Then Begin
            WriteLn( 'Usage: HDLOCK <drive:> /L /U <file_name>' );
            WriteLn;
            WriteLn( '<drive:>      - Hard Drive To Lock/Unlock' );
            WriteLn( '/L              - Lock The Especificed Hard Drive' );
            WriteLn( '/U              - Unlock The Especificed Hard Drive' );
            WriteLn( '<file_name> - File name which will be used to save/' );
            WriteLn( '                restore the hard disk information' );
            WriteLn;
            Halt;
        End;

        { Analisa os parâmetros passados ao programa }
        HardDrive := 'C';
        Lock := False;
        Unlock := False;
        FileName := '';
        For Parm := 1 To ParamCount Do Begin
            StrParam := Upper( ParamStr( Parm ) );
            If Length( StrParam ) = 2 Then Begin
                If Pos( ':', StrParam ) = 2 Then
                    HardDrive := StrParam[ 1 ]
                Else If Pos( '/', StrParam ) = 1 Then
                    Case StrParam[ 2 ] Of
                        'L' : Lock := True;
                        'U' : Unlock := True;
                    End;
                End Else FileName := StrParam;
            End;
        End;

        { Processa as ações pedidas }
        If Lock And Unlock Then
            WriteLn( #7, 'Choose Hard Disk Lock or UnLock !' )
        Else
            If Lock Then HDLock Else
            If Unlock Then HDUnlock Else
            If Length( FileName ) > 0 Then If SaveHDIImage Then;
        End;
```



## Redes Neurais Artificiais

### Parte 1

Por: Carlos A. Thompson Ph.D

*Micro Sistemas traz até você uma excelente oportunidade para entrar no mundo das Redes Neurais Artificiais com o nosso Mestre da matemática, Prof. Carlos A. Thompson Ph.D.*

Rede Neural Artificial (RNA) trata da simulação de um sistema nervoso humano o qual contém uma coleção de neurônios que se comunicam via axônios e dendritos. Os *dendritos* são ramificações destinadas a receber os sinais eletromagnéticos de outros neurônios. Esses sinais são transmitidos pelo axônio, conforme é mostrado na figura 1. A junção de um dendrito com um axônio é denominada de sinapse.



Figura 1

Na figura 1 temos o esquema simplificado de transmissão de ondas entre neurônios de uma rede biológica.

Pelos dados das pesquisas em anatomia, estima-se que existem mais de  $10^3$  sinapses nas entradas e saídas de um único neurônio. Sabe-se também que o cérebro humano contém cerca de  $10^{11}$  neurônios e  $10^{14}$  sinapses. Outro dado importante é que o tempo de resposta dos neurônios seja na ordem de fração do milissegundos enquanto que os computadores, em geral, são cerca de um milhão de vezes mais rápidos. Entretanto, o grau de conectividade desses neurônios, por sua vez, são da ordem de mil vezes maior do que o dos computadores. Ainda existem, por parte dos pesquisadores, alguns questionamentos sobre as interações entre os diversos tipos de neurônios. Sabe-se que diferentes tipos de neurônios interagem distintamente.

#### *O que se entende por threshold?*

Suponha que um instrumento seja submetido a um estímulo ou entrada que aumente gradualmente de zero até um certo valor mínimo em que se comece a observar a resposta ou saída. Este valor mínimo é denominado de threshold ou disparo do instrumento. Cada sistema tem o seu threshold característico. No caso de propagação de ondas cerebrais,



um certo neurônio coleta todos os sinais em suas sinapses somando todos aqueles que são estímulos com aqueles que possuem características inibidoras. Se a influência dos sinais excitadores for maior do que aquela dos inibidores, então o neurônio disparará a mensagem recebida para outros neurônios via suas sinapses de saída. Neste caso tem-se as chamadas funções de classe  $C^1$ , caso contrário as funções são denominadas de classe  $C^2$ .

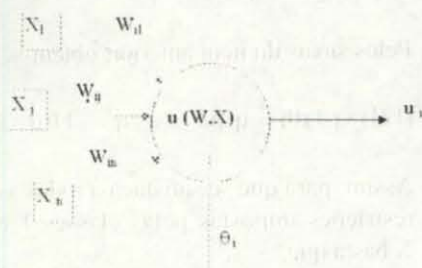


Figura 2

Na figura 2 temos o sinal  $u_i$  que poderá ou não ser transmitido para outros neurônios, dependendo da comparação de sua intensidade com o threshold  $q_i$  da sinapse  $i$ .

## Função Base

A função base para uma determinada sinapse ( $i$ ) corresponde à diferença entre os sinais excitadores e os inibidores (a função base são chamadas de **net function**). A função base  $u(X, W)$  representa, matematicamente, as conexões das redes neurais onde  $X$  é o vetor de entrada e  $W$  a matriz peso. Em cada sinapse,  $u_i$  e  $q_i$  são independentes.

Geralmente a função base pode ser linear ou quadrática. No caso do modelo analítico linear, escreve-se,

$$u_i(W, X) = \sum_{j=1}^n W_{ij} X_j$$

Sendo o peso  $W$  uma grandeza escalar, a função base linear é também chamada de função hiper-plano. De acordo com o que foi mencionado anteriormente, tem-se,

$$\sum_{j=1}^n W_{ij} X_j \geq 0 \Rightarrow \sum_{j=1}^n W_{ij} X_j - 0_i \geq 0 \quad (\text{Classe } C^1)$$

$$\sum_{j=1}^n W_{ij} X_j < 0 \Rightarrow \sum_{j=1}^n W_{ij} X_j - 0_i < 0 \quad (\text{Classe } C^2)$$

No caso do modelo analítico quadrático, escreve-se,

$$u_i(W, X) = \left[ \sum_{j=1}^n (X_j - W_{ij})^2 \right]^2$$

Neste caso, a função base é também chamada de função hiper-esfera. Essas funções podem ser generalizadas para as chamadas funções elíticas ou hiper-elipsóides.

De acordo com a classificação em classes,

$$\sum_{j=1}^n (X_j - W_{ij})^2 \geq 0 \Rightarrow \sum_{j=1}^n (X_j - W_{ij})^2 - 0_i \geq 0 \quad (\text{Classe } C^1)$$

$$\sum_{j=1}^n (X_j - W_{ij})^2 < 0 \Rightarrow \sum_{j=1}^n (X_j - W_{ij})^2 - 0_i < 0 \quad (\text{Classe } C^2)$$

## 3. Função de Ativação ou de Transferência

A função de transferência será vista posteriormente.

## 4. Perceptron

Considere duas moedas idênticas lançadas simultaneamente. Deseja-se construir uma rede neural que aponte ("perceba") a ocorrência de cara e de coroa simultaneamente. De acordo com o conceito do perceptron, considere o sistema binário onde, por hipótese, o que for percebido receberá o dígito 1 (um) e o que não for, o dígito 0 (zero). Seja ainda o evento cara ser representado por  $X_1$  e o coroa por  $X_2$ , conforme a tabela

$X_1$ (cara)	$X_2$ (coroa)	$u$ (saída)
1	0	1
0	1	1
1	1	0
0	0	0

Observe que o perceptron aponta somente a ocorrência do que for cara e coroa ( $u = 1$ ) ou (conectivo excludente)

do que não for cara e coroa. O sistema, por exemplo, não aponta a ocorrência de duas caras e de duas coroas. Assim, quando a saída  $u$  for nula, garante-se que não ocorreram cara e coroa simultaneamente. Esta idéia do perceptron pode ser ainda mais bem entendida quando se começa a ensinar a uma criança o que é uma televisão. Ao adquirir paulatinamente este conhecimento, o cérebro da criança também aprende a distinguir o que é ( $u = 1$ , por exemplo) e o que não é uma televisão ( $u = 0$ ). Nas redes neurais artificiais, o aprendizado se reduz a se obter os pesos de acordo com os testes e com as comparações com os resultados (ou alvos) que se deseja.

Pela figura 2, considerando-se o peso  $W_{0i} = -q_i$  e  $X_0 = 1$ , escreve-se,

$$\sum_{j=0}^n W_{ij} X_j \geq 0 \Rightarrow u = 1 \text{ ou Classe } C^1$$

$$\sum_{j=0}^n W_{ij} X_j < 0 \Rightarrow u = 0 \text{ ou Classe } C^2$$

Para  $q$  positivo, desenvolvendo-se em parcelas para o caso binário, tem-se,

### Classe 1:

$$W_1 X_1 + W_2 X_2 - q \geq 0 \Rightarrow W_1 + W_2 - q \geq 0$$

Somando-se as desigualdades obtém-se,

$$W_1 + W_2 \geq 2q$$

### Classe 2:

$$W_1 X_1 + W_2 X_2 - q < 0 \Rightarrow W_1 + W_2 - q < 0 \Rightarrow W_1 + W_2 < q$$

Observe que existe um conflito de modo que os pesos (aprendizagem) satisficam simultaneamente as classes  $C^1$  e  $C^2$ , ou seja, como é que a soma dos pesos pode ser maior ou igual a um número positivo  $2q$  e ao mesmo tempo essa soma ser menor do que a metade  $q$  desse mesmo número?



## A necessidade da camada

escondida.

Este problema só pode ser resolvido se for inserida uma camada intermediária (ou escondida) de sinapses, conforme a figura 3 (perceptron multilaminar). Os threshold da camada

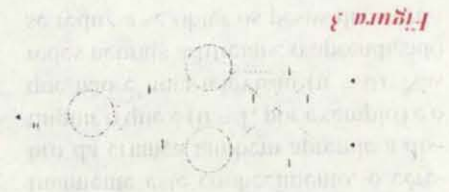
escondida são  $q_1$  e  $q_2$ .

Fig. 3. Perceptron Multilaminar onde

$q_1$ ,  $q_2$  e  $q_3$  são os threshold,  $X_1$ ,  $X_2$ ,  $X_3$  e  $X_4$  são pesos iguais a 1 e -1 e  $u$  o sinal de saída.

Pela tabela anterior.

## Classe 2:

$$X_1 = 0 \text{ e } X_2 = 0 \text{ p } u = 0$$

a1) threshold  $q_1$ 

$$(1)(0) + (1)(0) < q_1 \text{ p } 0 < q_1 \text{ e } q_1 > 0$$

a2) threshold  $q_2$ 

$$(1)(0) + (1)(0) < q_2 \text{ p } 0 < q_2 \text{ e } q_2 > 0$$

a3) threshold  $q_3$ 

Pelos sinais do item anterior, ob-

tem-se:

$$(1)(0) + (1)(0) < q_1 \text{ p } 0 < q_1 \text{ e } q_1 > 0$$

tem-se, para que a saída seja 0, basta que os

threshold  $q_1$ ,  $q_2$  e  $q_3$  sejam positivos.

$$\text{b) } X_1 = 1 \text{ e } X_2 = 1 \text{ p } u = 0$$

b1) threshold  $q_1$  (impondo-se o sinal de saída igual a 1)

$$(1)(1) + (1)(1) < q_1 \text{ p } 2 < q_1 \text{ e } q_1 < 2$$

b2) threshold  $q_2$  (impondo-se o sinal de saída igual a 1)

$$(1)(1) + (1)(1) < q_2 \text{ p } 2 < q_2 \text{ e } q_2 < 2$$

threshold  $q_1$  e  $q_2$  possuem dígitos 1.

Note que os sinais que deixam os

b3) threshold  $q_3$  (impondo-se o sinal de saída igual a 0)

Pelos sinais do item anterior e com

os pesos +1 e -1, tem-se:

$$(1)(1) + (-1)(1) < q_3 \text{ p } 0 < q_3 \text{ e } q_3 = 0$$

Classe 1:

$$X_1 = 1 \text{ e } X_2 = 0 \text{ p } u = 1$$

a1) threshold  $q_1$  (impondo-se que o si-

nal de saída seja 1, neste caso)

$$(1)(1) + (1)(0) \geq q_1 \text{ p } 1 \geq q_1 \text{ e } q_1 \leq 1$$

a2) threshold  $q_2$  (impondo-se que o si-

nal de saída seja nulo)

$$(1)(1) + (1)(0) \leq q_2 \text{ p } 1 \leq q_2 \text{ e } q_2 \geq 1$$

a3) threshold  $q_3$  (impondo-se que o si-

nal de saída seja 1)

Pelos sinais do item anterior (1 pelo

threshold  $q_1$  e 0 pelo  $q_2$ ), obtem-se:

$$(1)(1) + (-1)(0) \geq q_3 \text{ p } 1 \geq q_3 \text{ e } q_3 \leq 1 \text{ p } u = 1$$

$$\text{b) } X_1 = 0 \text{ e } X_2 = 1 \text{ p } u = 1$$

b1) threshold  $q_1$  (impondo-se que o si-

$$(1)(0) + (1)(1) > q_1 \text{ p } 1 > q_1 \text{ e } q_1 < 1$$

b2) threshold  $q_2$  (impondo-se que o si-

$$(1)(0) + (1)(1) < q_2 \text{ p } 1 < q_2 \text{ e } q_2 > 1$$

threshold  $q_1$  (impondo-se que o si-

nal de saída seja 1)

Pelos sinais do item anterior, obtem-se:

$$(1)(1) + (-1)(0) \geq q_3 \text{ p } 1 \geq q_3 \text{ e } q_3 \leq 1 \text{ p } u = 1$$

Assim, para que se satisficam a todas as

restrições impostas pelas classes 1 e

2, basta que:

$$q_1 < 1$$

$$q_2 < 1$$

$$1 < q_3 < 2$$

Dr. Carlos Thompson é Professor Ti-

tuar do Centro Universitário Inguiso

Motta e Professor do curso de

Algoritmos da FEF. Atualmente des-

envolve algoritmos computacionais na

área de Administração de Empresas e

orienta Pós-graduação de Mestrado e de

Home Page: <http://home.open-link.com.br/~c-thompson>

e-mail: [c-thompson@openlink.com.br](mailto:c-thompson@openlink.com.br)

22



# O MELHOR DA INFORMÁTICA NACIONAL

*Quem quer ficar "por dentro" da informática nacional não pode deixar de ler Micro Sistemas. Sempre atenta aos acontecimentos e tendências, Micro Sistemas é a única revista que mais reflete o que acontece no mercado brasileiro. Feita por brasileiros e para brasileiros Micro Sistemas traz sempre em suas páginas os assuntos mais quentes do momento, programas em diversas linguagens, rotinas, livros, cartas de leitores, etc.*

*Por tudo isso você não pode deixar de ler Micro Sistemas. Não perca mais tempo! Garanta mensalmente o seu exemplar fazendo uma assinatura anual de MICRO SISTEMAS por apenas:*

## "PROMOÇÃO"

1 ANO - R\$ 50,00

2 ANOS - R\$ 100,00

Para adquirir números anteriores enviar R\$ 5,00 para cada exemplar desejado.

**NÃO PERCA TEMPO!**  
FAÇA JÁ SUA ASSINATURA  
de MICRO SISTEMAS

**Micro  
Sistemas**

Caixa Postal 18347  
Rio de Janeiro - RJ  
CEP 20722-970

Tel. (021) 532-0770 cod. 4016779

Nome: .....  
Endereço: .....  
Bairro: ..... Cidade: ..... UF: .....  
CEP: ..... Telefone: .....  
Assinatura: ..... Data: ...../...../.....  
☐ Estou enviando cheque cruzado nominal à PRB informática  
Editora Ltda, referente à uma assinatura de MICRO SISTEMAS  
Banco: ..... Ag: ..... cheque nº.: .....



# ARTIGO

## *Jogo Infantil em Visual Basic*

*Por: Mauri Resende Faria*

*O jogo apresentado pelo autor demonstra como obter movimento de uma imagem na tela, usando o Visual Basic.*

Sempre gostei muito de programação e tive o interesse de programar com movimentação e som na tela. O som para mim, ainda não foi possível como vocês poderão perceber neste software, mas estou trabalhando para alcançar este objetivo. Aguardem.

Quando aprendi programação o movimento não era possível, mas hoje com os recursos que as linguagens gráficas oferecem fica bem fácil como veremos na sequência deste texto.

Há seis anos aprendi a programar através da linguagem de programação Basic, que foi sendo modificada e ganhou novos nomes como GBasic, Turbo Basic, etc. Há três meses tive a oportunidade de conhecer uma dessas linguagens gráficas que é o Visual Basic 3.0.

Sou leitor desta revista, porém ainda não tive a oportunidade de assiná-la, mas a compro nas bancas todos os meses. Na edição número 160 li uma matéria, publicada por ela mesma, com o título de "PROCURA-SE TALENTOS - Dê uma chance ao sucesso liberando seus bytes" o qual me interessou bastante e comecei a pensar algo fácil de se fazer e com um bom resultado. Bem, o resultado foi um sensacional jogo para o público infantil e aqueles que gostam de testar seus reflexos.

Este jogo tem como objetivo passar uma nave espacial em um espaço aberto na parte superior do vídeo que se move de lugar a cada tentativa, com três níveis de dificuldade para facilitar ou dificultar a jogada.

A tela principal do jogo (veja figura-1) contém 10 objetos, descritos abaixo, com seus respectivos nomes, posições na tela e propriedades dos objetos que deverão ser alterados.

MIRO  
2679212

16821 Janela 18847  
161 de Janeiro - 161  
161 20723-971

*Figura-1 Formulário principal*



## - Form1

```

Caption = Jogo
ForeColor = &H00FFFFFF& (Branco)
- 1 SSPanel - PnlControle na parte inferior do vídeo.
Name - PnlControle
OutLine - True
ShadowColor - 1-Black
BevelInner - 1-Inset
Caption - "" (Vazio)
- 1 PictureBox - PicNave sobre a barra inferior do vídeo.
Name - PicNave
Picture - C:\VB\Icons\Industry\ROCKET.ICO
AutoSize - True
BorderStyle - 0-None
Depois de feito os procedimentos acima marque o objeto com um click, ative o menu "Edit" e escolha o submenu "Send to Back", para que a Nave fique abaixo da barra de controle.
- 3 CommandButton
CmdNivel dentro da barra inferior (PnlControle)
Name - CmdNivel
Caption - "Nível"
CmdBarral no canto superior esquerdo do vídeo
Name - CmdBarral
Caption - "" (Vazio)
CmdBarra2 no canto superior direito do vídeo
Name - CmdBarra2
Caption - "" (Vazio)
- 2 Label
LblPontos dentro da barra inferior (PnlControle)
Name - LblPontos
Caption - "Pontos : "
BackColor = &H00C0C0C0& (Cinza Claro)
LblTentativas dentro da barra inferior (PnlControle)
Name - LblTentativas
Caption - "Tentativas : "
BackColor = Cinza Claro
- 2 TextBox
TxtPontos na frente do objeto LblPontos
Name - TxtPontos
BackColor - Cinza Claro
BorderStyle - 0-None
Text - "" (Vazio)
TxtTentativas na frente do objeto LblTentativas
Name - TxtTentativas
BackColor - Cinza Claro

```

A segunda tela (veja figura-2) contém as opções de níveis com os seguintes objetos:

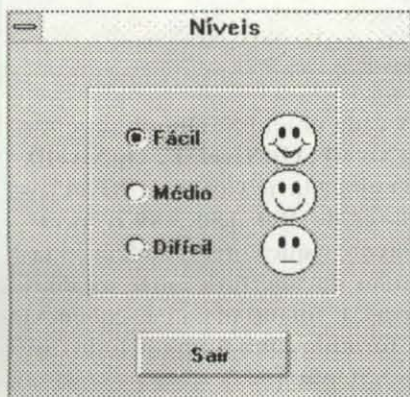


Figura-2 Formulário de Níveis

## - Form2

```

Name - FormNiveis
BackColor - Cinza Claro
FillColor - Cinza Claro
ForeColor - Cinza Claro
Caption - "Niveis"
MaxButton - False
MinButton - False
- 1 CommandButton
CmdSair na parte inferior do formulário
Name - CmdSair
Caption - "Sair"
- 1 SSFrame
FrameContorno abrangendo os textos e as figuras

```

```

Name - FrameContorno
Caption = "" (Vazio)
- 3 SSOption
OpNivelF dentro do FrameContorno
Name - OpNivelF
Caption - "Fácil"
OpNivelM dentro do FrameContorno
Name - OpNivelM
Caption - "Médio"
OpNivelD dentro do FrameContorno
Name - OpNivelD
Caption - "Difícil"
- 3 PictureBox
PicFacil na frente do objeto OpNivelF
Name - PicFacil
Picture = C:\VB\Icons\Misc\FACE03.ICO
AutoSize - True
BackColor - Cinza Claro
BorderStyle - 0-None
PicMedio na frente do objeto OpNivelM
Name - PicMedio
Picture = C:\VB\Icons\Misc\FACE02.ICO
AutoSize - True
BackColor - Cinza Claro
BorderStyle - 0-None
PicDificil na frente do objeto OpNivelD
Name - PicDificil
Picture = C:\VB\Icons\Misc\FACE01.ICO
AutoSize - True
BackColor - Cinza Claro
BorderStyle - 0-None

```

Seguem alguns comentários relacionados com o jogo, para melhor compreensão dos comandos usados no formulário principal.

- No evento *PicNave\_Click*, foi usado o comando **Rnd** para gerar um número que irá alterar o tamanho da abertura por onde a nave irá passar. Foi usado uma variável com o nome "**NovoEspaco**" que armazenará um valor gerado pelo comando, este será transferido para o objeto *CmdBarral.Width* com a fórmula "**NovoEspaco = Int(6 \* Rnd + 1) \* 1000**" através da qual será deslocada a posição direita do objeto *CmdBarral*.

- Para que a posição da segunda barra (*CmdBarra2*) também seja alterada, fazendo com que sempre a abertura seja



igual a cada tentativa, foram feitos os seguintes procedimentos dentro do evento *PicNave.Click*: Armazenou-se o valor da posição esquerda do objeto *CmdBarra2* em uma variável com o nome de *Barra2W* (*Barra2W = CmdBarra2.Left*) para que posteriormente o mesmo seja ajustado no canto esquerdo do vídeo. Logo em seguida foi posicionado o objeto *CmdBarra2* onde terminou o objeto *CmdBarra1* somando o espaço dependendo do nível escolhido pelo usuário, usando a fórmula "*CmdBarra2.Left = CmdBarra1.Width + PicNave.Width + Nivel*". Para que o seu comprimento seja ajustado até a última coluna do vídeo foi utilizado a fórmula "*CmdBarra2.Width = (Barra2W - CmdBarra2.Left) + CmdBarra2.Width*", ou seja, a diferença que ocorreu mais o seu tamanho existente.

- Para que a nave movimente somente para a direita e a esquerda, foi usado o evento "*MouseMove*" do formulário principal (*Form MouseMove*) onde a variável "*X*", fornecida pelo evento, é adotada para o valor esquerdo do objeto *PicNave* como se segue: "*PicNave.Left = X*". E para que a nave fique sempre sobre a barra inferior do vídeo foi adotado o comando "*PicNave.Top = PnlControle.Top - PicNave.Height*".

- Para que a nave suba em linha reta quando for dado um clique sobre a mesma, foi usado um "loop" dentro do evento "*PicNave.Click*" como descrito abaixo:

```
Do While Not PicNave.Top =
TopoMax%
    PicNave.Top = PicNave.Top
    - 1
Loop
```

- Se a tentativa de acertar o espaço estiver na direção correta a nave subirá até que desapareça do vídeo, caso contrário ela irá somente até encontrar com uma das barras que estiver em sua direção. Para este caso foi verificado se as coordenadas da esquerda da nave (*PicNave.Left*) estão após as coordenadas da direita do objeto *CmdBarra1* e se as coordenadas da direita da nave (*PicNave.Width*) estão antes do lado esquerdo do objeto *CmdBarra2*. Para tal processo foi usado o seguinte procedimento no evento *Click* do objeto *PicNave*.

```
If CmdBarra1.Width - 20 < PicNave.Left And PicNave.Left + PicNave.Width < CmdBarra2.Left Then
    TopoMax% = -(PicNave.Width)
Else
    TopoMax% = CmdBarra1.Height
End If
```

O valor "-20" na fórmula acima é usado para tirar uma diferença

que contém na própria figura da nave.

- Para obter o valor da quantidade de vezes que a nave foi acionada acumulou-se ao objeto *TxtTentativas* o valor de mais 1 (um) a cada clique dado na figura da nave. A fórmula é "*TxtTentativas.Text = Str(Val(TxtTentativas.Text) + 1)*".

Para verificar se o jogador obteve pontos o valor da variável *TopoMax%* tem que ser menor que zero, isto significa que a nave passou na abertura e sumiu do vídeo. Então é somado ao objeto *TxtPontos.Text* o valor de mais 1 (um), caso contrário, não é somado. A equação está descrita abaixo:

```
If TopoMax% < 0 Then
    TxtPontos.Text =
    Str(Val(TxtPontos.Text) + 1)
End If
```

- A cada *Click* na figura da nave é mostrada uma mensagem na barra inferior do vídeo (*PnlControle*) para que o jogador posicione o mouse naquela área, para que, quando a nave terminar de percorrer a trajetória, o jogador veja o resultado, porque se o mouse estiver fora da barra a nave é posicionada novamente no ponto de partida.

No início do evento "*PicNave.Click*" é colocada a mensagem da seguinte forma "*PnlControle.Caption = "Posicione o mouse nesta área."*" e após o loop que faz com que a nave suba, a mensagem é retirada com o comando "*PnlControle.Caption = ""*".

- No evento "*Form\_Load()*" do formulário principal, foi usado um comando para centralizar o mesmo e iniciar o jogo com o nível de dificuldade "Difícil" como se segue:

```
Left = (Screen.Width - Width) / 2
Top = (Screen.Height - Height) / 2
Nivel = 80
Barra2W = CmdBarra2.Left
CmdBarra2.Left = CmdBarra1.Width + PicNave.Width + Nivel
CmdBarra2.Width = (Barra2W - CmdBarra2.Left) +
CmdBarra2.Width
```

- Toda vez que o comando de níveis (*CmdNivel*) for acionado, será zerado o número de pontos e a quantidade de tentativas que estiver armazenada no objeto *TxtTentativas* e será aberto o formulário de níveis (*Form2*) contendo três níveis de dificuldade que são Fácil, Médio e Difícil. Os comandos foram escritos dentro do evento *CmdNivel.Click* como descritos abaixo:

```
TxtPontos.Text = 0
TxtTentativas.Text = ""
FormNiveis.Show
```

Vejamos em seguida alguns comentários sobre o formulário de níveis.

- Quando o comando "Sair" deste formulário for acionado com um clique, apenas fechará o formulário como se fosse uma janela ativa.  
*Unload FormNiveis*

- Também no evento "*Form\_Load()*" deste formulário, existem os mesmos comandos do formulário principal, que o centralizam no vídeo, que são:

```
Left = (Screen.Width -
Width) / 2
Top = (Screen.Height -
Height) / 2
```

- A cada objeto *SSOption*, que são *OpNivelF*, *OpNivelM* e *OpNivelD*, foi colocada na variável "*Nivel*" os valores de 300, 180 e 80 respectivamente e que correspondem ao tamanho da abertura entre as duas barras. E foi criada uma sequência de fórmulas iguais para todos os três *SSOption*:



# ARTIGO / Jogo Infantil em Visual Basic

```
Barra2W = Form1.CmdBarra2.Left  
Form1.CmdBarra2.Left = Form1.CmdBarral.Width + Form1.PicNave.Width + Nivel  
Form1.CmdBarra2.Width = (Barra2W - Form1.CmdBarra2.Left) + Form1.CmdBarra2.Width
```

Esta sequência de fórmulas faz com que a cada opção escolhida o nível seja alterado automaticamente no momento da escolha, sem que seja preciso fechar a janela de opções de níveis.

Finalmente foi criado um módulo (.BAS) para que a variável "Nível" possa ser aceita em ambos os formulários. Neste módulo foi iniciado a variável da seguinte forma: Global Nivel As Integer

**Mauri Resende Faria** é programador em Clipper e Cobol e atualmente está ampliando seus conhecimentos em Visual Basic e Delphi. Eventuais contatos com Mauri Resende Faria podem ser feitos pelo endereço: R.Jataí, 328 - Bom Jesus - Uberlândia - MG - CEP 38406-016

## Comandos do formulário principal Form1:

```
Sub CmdNivel_Click ( )  
    'Zerando os pontos e chamando a tela de níveis.  
    TxtPontos.Text = " "  
    TxtTentativas.Text = " "  
    FormNiveis.Show  
End Sub  
  
Sub Form_Load ( )  
    'Centralizar o formulário.  
    Left = (Screen.Width - Width) / 2  
    Top = (Screen.Height - Height) / 2  
    'Inicia o espaço com o nível Difícil.  
    Nivel = 80  
    Barra2W = CmdBarra2.Left  
    CmdBarra2.Left = CmdBarral.Width +  
    PicNave.Width + Nivel  
    CmdBarra2.Width = (Barra2W - CmdBarra2.Left) +  
    CmdBarra2.Width  
End Sub  
  
Sub Form_MouseMove (Button As Integer, Shift As  
Integer, X As Single, Y As Single)  
    'Movimentar a Nave para a esquerda e a direita  
    PicNave.Left = X  
    'Posicionar a Nave sobre a Barra de Controle  
    PicNave.Top = PnlControle.Top - PicNave.Height  
End Sub  
  
Sub PicNave_Click ( )  
    'Mensagem  
    PnlControle.Caption = "Posicione o mouse nesta  
área."  
    'Verificando se vai acertar o espaço  
    'é se a Nave vai sumir do video ou não.  
    If CmdBarral.Width - 20 < PicNave.Left And PicNave.Left +  
    PicNave.Width < CmdBarra2.Left Then  
        TopoMax% = -(PicNave.Width)  
    Else  
        TopoMax% = CmdBarral.Height  
    End If  
    'Faz com que a Nave suba em linha reta.  
    Do While Not PicNave.Top = TopoMax%  
        PicNave.Top = PicNave.Top + 1  
        Loop  
    'Somando os pontos e os números de tentativas.  
    TxtTentativas = Str(Val(TxtTentativas.Text)  
+ 1)  
    If TopoMax% < 0 Then  
        TxtPontos.Text = Str(Val(TxtPontos.Text)  
+ 1)  
    End If  
    'Gerar nova posição do espaço.  
    Dim NovoEspaco  
    Randomize  
    NovoEspaco = Int(6 * Rnd + 1) * 1000
```

```
CmdBarral.Width = NovoEspaco
```

```
'Ajustar a segunda Barra com o novo espaço.  
Barra2W = CmdBarra2.Left  
CmdBarra2.Left = CmdBarral.Width +  
PicNave.Width + Nivel  
CmdBarra2.Width = (Barra2W -  
CmdBarra2.Left) + CmdBarra2.Width  
'Retirando a mensagem  
PnlControle.Caption = " "  
End Sub
```

## Comandos do formulário de níveis Form2:

```
Sub CmdSair_Click ( )  
    Unload FormNiveis  
End Sub  
  
Sub Form_Load ( )  
    'Centraliza o formulário na tela.  
    Left = (Screen.Width - Width) / 2  
    Top = (Screen.Height - Height) / 2  
End Sub  
  
Sub OpNivelD_Click (Value As Integer)  
    'Espaço entre as Barras  
    Nivel = 80  
    'Atualiza a posição entre as Barras  
    Barra2W = Form1.CmdBarra2.Left  
    Form1.CmdBarra2.Left =  
    Form1.CmdBarral.Width + Form1.PicNave.Width + Nivel  
    Form1.CmdBarra2.Width = (Barra2W  
    Form1.CmdBarra2.Left) +  
    Form1.CmdBarra2.Width  
End Sub  
  
Sub OpNivelF_Click (Value As Integer)  
    'Espaço entre as Barras  
    Nivel = 300  
    'Atualiza a posição entre as Barras  
    Barra2W = Form1.CmdBarra2.Left  
    Form1.CmdBarra2.Left =  
    Form1.CmdBarral.Width + Form1.PicNave.Width + Nivel  
    Form1.CmdBarra2.Width = (Barra2W  
    Form1.CmdBarra2.Left) +  
    Form1.CmdBarra2.Width  
End Sub  
  
Sub OpNivelM_Click (Value As Integer)  
    'Espaço entre as Barras  
    Nivel = 180  
    'Atualiza a posição entre as Barras  
    Barra2W = Form1.CmdBarra2.Left  
    Form1.CmdBarra2.Left =  
    Form1.CmdBarral.Width + Form1.PicNave.Width + Nivel  
    Form1.CmdBarra2.Width = (Barra2W  
    Form1.CmdBarra2.Left) +  
    Form1.CmdBarra2.Width  
End Sub
```



## Microprocessadores e Placas de CPU

Por: Laércio Vasconcelos

Nesta série, iniciada na Micro Sistemas # 164, Laércio Vasconcelos apresenta os microprocessadores usados nos PCs atuais, além de discutir aspectos técnicos sobre as placas de CPU que os utilizam.

O texto extraído do livro "Upgrade Expert" volume 1.

Se você trabalha no ramo de upgrade de micros, faz isto por hobby ou apenas deseja aumentar seus conhecimentos, o livro "Upgrade Expert" de Laércio Vasconcelos é uma ferramenta necessária para você.

### Alguns elementos da placa de CPU

Mais adiante, faremos uma apresentação mais detalhada dos barramentos (slots) e tipos de memórias usadas nas placas de CPU. No momento, vamos apresentar uma discussão adicional sobre uma pequena miscelânea de elementos presentes em todas as placas de CPU, desde as mais antigas até as mais modernas.

### Chips SSI, MSI, LSI e VLSI

Os circuitos integrados, mais popularmente conhecidos como "chips", existem desde os anos 60. Em seu interior, existe uma grande quantidade de transistores. O número de transistores existentes depende do grau de miniaturização obtido pelo fabricante. Os primeiros circuitos integrados continham apenas algumas dezenas de transistores. Um microprocessador 486 possui em seu interior cerca de 1.500.000 transistores, enquanto o Pentium e o Pentium Pro possuem, respectivamente, 3.500.000 e 5.500.000 transistores.

Figura 19  
Chips SSI e MSI.

Os circuitos integrados mais simples, em cujo interior existem algumas dezenas de transistores, são chamados de SSI (Short Scale of Integration, ou seja, baixa escala de integração). Circuitos integrados um pouco mais sofisticados, com o equivalente a cerca de algumas centenas de transistores,

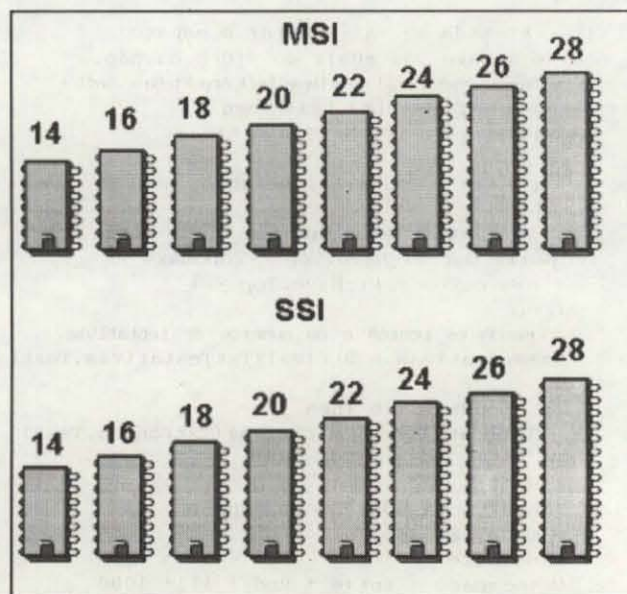


Figura 19



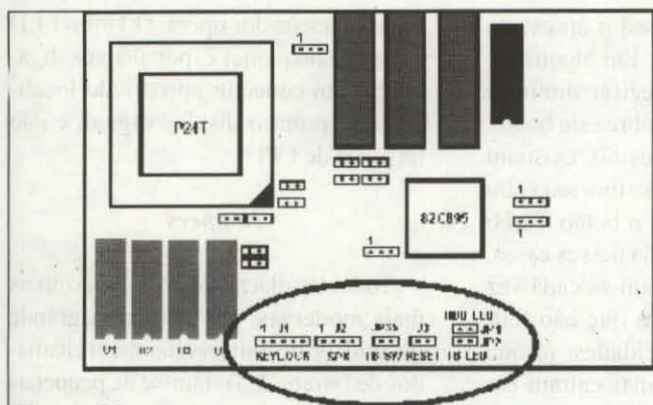


Figura 20

são chamados de MSI (Medium Scale of Integration, ou seja, média escala de integração). A figura 19 mostra o aspecto desses chips. Apresentam um encapsulamento conhecido como DIP (Dual In-Line Package), podendo possuir desde 14 até 28 pinos.

Todas as placas de CPU antigas apresentavam uma grande quantidade de chips SSI e MSI. A placa de CPU do IBM PC XT, bem como suas placas de expansão, estão repletas desses chips. Com o passar dos anos, esses chips foram aos poucos incorporados dentro de outros, mas mesmo assim ainda são utilizados, apesar de ser em pequenas quantidades. Observe por exemplo as placas de CPU 386, 486 e Pentium apresentadas nas figuras anteriores. Todas elas possuem pelo menos alguns chips MSI ou SSI.

As placas usadas nos primeiros PCs, fabricados no início dos anos 80, já empregavam alguns chips especiais, com circuitos equivalentes a alguns milhares de transistores. Esses chips são chamados de LSI (Large Scale of Integration, ou seja, larga escala de integração). Alguns desses chips, usados nas placas de CPU XT, e nas primeiras placas de CPU 286 eram os seguintes:

**8237** - Controlador de DMA (acesso direto à memória). O XT possuía um controlador, e o AT possuía dois deles.

**8259** - Este é o controlador de interrupções usado no XT. No IBMAT existiam dois deles.

**8253** - Este é o chamado TIMER, responsável pela operação do relógio, da interface de alto falante e da operação de refresh das memórias DRAM.

**8254** - Este é o TIMER usado no IBM AT, que veio a substituir o 8253.

**8250** - Este chip é conhecido como UART (transmissor-receptor assíncrono universal), e é usado para a implementação das interfaces seriais.

6845 - Este é o chip controlador de vídeo gráfico, usado nas primeiras placas de vídeo para PC: CGA (Color Graphics Adapter) e MDA (Monochrome Display Adapter).

**8272** - Também fabricado pela NEC, com o nome de NEC 765, este era o chip que fornecia a interface de drives usada nos primeiros PCs.

As placas de CPU e as placas de expansão atuais raramente usam chips LSI. Todas elas empregam chips muito mais avançados, chamados de VLSI (Very Large Scale of Integration, ou seja, escala de integração muito alta). Esses chips contêm, em seu interior, várias dezenas, e até mesmo centenas de milhares de transistores. São funcionalmente equivalentes ao conjunto de chips SSI, MSI e LSI usados nos PCs antigos, porém, com recursos mais avançados, sobretudo no que diz respeito à velocidade. Os chips VLSI também são conhecidos como CHIP SET, e graças a eles as modernas placas de CPU e

de expansão são mais compactas, rápidas, baratas e confiáveis, além de apresentarem menos aquecimento que as placas antigas.

Os chips VLSI apresentam o formato quadrado ou retangular. Possuem nos seus quatro lados, uma grande quantidade de contatos (que os leigos chamam de “perninhas”, mas o correto é chamá-los de terminais). Você pode observar chips VLSI nas placas de CPU 386, 486 e Pentium mostradas nas figuras anteriores.

Os principais fabricantes de chips VLSI para placas de CPU são a própria Intel, a OPTI, a UMC e a National Semiconductor. Existem ainda os chips VLSI utilizados em controladoras de discos de vários tipos, cujo principal fabricante é a Western Digital, e ainda os chips usados em placas gráficas, cujos principais fabricantes são a S3, a ATI, a Cirrus Logic e a Trident.

### Conectores para o Painel

Todas as placas de CPU possuem conectores para serem ligados aos dispositivos existentes no painel frontal do gabinete. O conhecimento desses conectores é importante para quem pretende montar seu computador, e também para quem pretende instalar uma nova placa de CPU. Tais conectores são ligados aos seguintes dispositivos do painel frontal do gabinete:

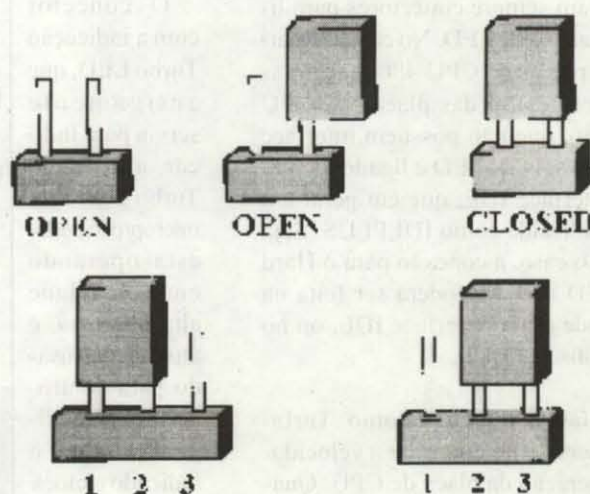
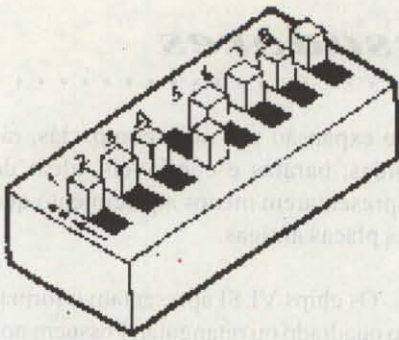


Figura 21





**Figura 22**

- Botão RESET
- Turbo Switch
- Turbo LED
- Chave que tranca o teclado
- Power LED
- Hard Disk LED
- Display Digital

O manual da placa de CPU sempre apresenta as instruções a respeito dessas conexões, como vemos na figura 20. Além disso, muitas vezes encontramos, ao lado de cada um desses conectores, uma indicação do nome da conexão a ser realizada (Reset, HDD LED, etc.).

**Figura 20**  
**Conexões para o painel frontal do gabinete.**

Todos esses dispositivos são ligados na placa de CPU, mas existem algumas ressalvas a serem feitas. Por exemplo, o Hard Disk LED, que acende quando o disco rígido é acessado, poderá sempre ser ligado ao próprio disco rígido. Alternativamente, pode ser ligado também na sua interface. Placas de CPU Pentium possuem interfaces IDE para conexão ao disco rígido, portanto, apresentam sempre conectores para ligar ao Hard Disk LED. No caso da maioria das placas de CPU 486 (exceto as mais recentes), e das placas de CPU 386 e 286, que não possuem interface IDE, o Hard Disk LED é ligado na placa de interface IDE, que em geral é a placa conhecida como IDEPLUS. Seja qual for o caso, a conexão para o Hard Disk LED sempre poderá ser feita na placa onde está a interface IDE, ou no próprio disco rígido.

O botão conhecido como Turbo Switch serve para controlar a velocidade de operação da placa de CPU. Quase sempre, operamos em modo Turbo,

no qual o microprocessador apresenta a sua plena velocidade. Em alguns raros casos, podemos precisar diminuir a velocidade, atuando sobre este botão. Durante o final dos anos 80, existiam vários programas que não funcionavam em altas velocidades, e o botão Turbo era usado com frequência nesses casos. Depois de 1990, tornaram-se cada vez mais raros os programas que não funcionam em altas velocidades, já que aqueles antigos programas caíram em desuso. Em geral são antigos utilitários para MS-DOS que realizam acessos diretos ao hardware, como é o caso de programas que fazem cópias de disquetes protegidos contra cópias (COPYIIPC e COPYWRIT). A necessidade de diminuir a velocidade do microprocessador é tão rara que já existem várias placas de CPU Pentium que não possuem conexão para o Turbo Switch, e permanecem o tempo todo operando em velocidade máxima. Mesmo assim, muitas dessas placas permitem que seja usada uma combinação de teclas para comutar entre o modo de alta (Turbo) e o de baixa velocidade. Em geral essas combinações são:

- Control Alt +
- Control Alt -
- Control Alt T

Nem sempre esses comandos podem ser usados, pois certos programas usam essas combinações para outros fins.

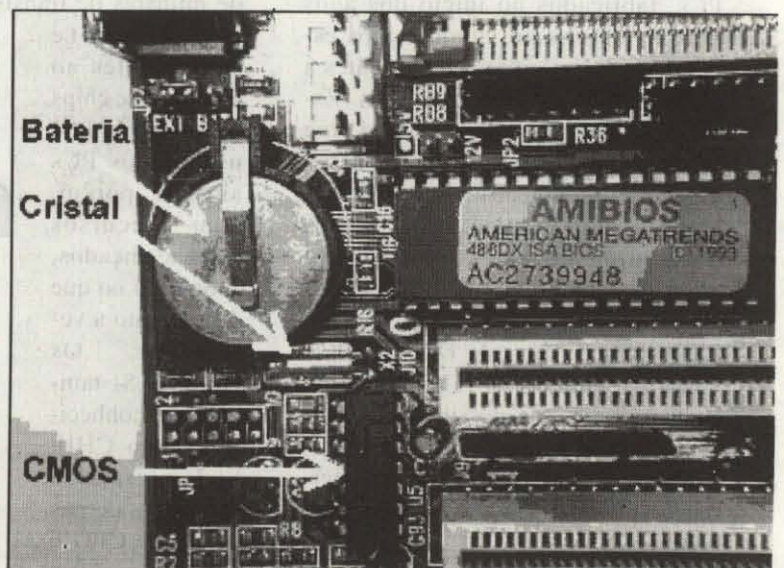
O conector com a indicação Turbo LED, que antigamente servia para indicar, através do Turbo LED se o microprocessador está operando em velocidade alta ou baixa, é atualmente usado para controlar o display digital, no qual é indicado o clock com o qual o

microprocessador opera. O Turbo LED existente no painel é, por sua vez, ligado em um conector apropriado localizado no próprio display digital, e não na placa de CPU.

## Jumpers

Todas as placas de CPU, mesmo as mais modernas, possuem uma grande quantidade de jumpers (também chamados de "straps"). Tratam-se de pequenas peças plásticas, medindo alguns milímetros, nas quais existem dois contatos ligados internamente. Esses jumpers são encaixados em pinos metálicos existentes nas placas de CPU e nas placas de expansão, e servem para definir determinadas opções de funcionamento a nível de hardware. Em cada par de pinos metálicos, existem duas opções: com jumper ou sem jumper. Existem também trios de pinos metálicos, nos quais existem três opções de encaixe: 1-2, 2-3 ou sem conexão. Cada uma delas possui um significado diferente. Para saber o exato significado de cada jumper, é preciso consultar as instruções existentes no manual da placa em questão. Podemos encontrar jumpers com diversas finalidades, tais como:

- Programação do clock do microprocessador
- Habilitação da recarga da bateria
- Definição da quantidade de memória CACHE





- Definição da voltagem entregue ao microprocessador
- Especificação do tipo de memória ROM instalada

Os jumpers definem opções de funcionamento a nível de hardware, mas também é comum encontrar várias opções definidas por software, através do CMOS Setup. Existem ainda casos de configuração automática. Por exemplo, existem placas de CPU que detectam automaticamente o tipo e a quantidade de memória DRAM ou cache instalada.

**Figura 21**  
*Jumpers.*

Normalmente, todos os jumpers existentes nas placas são configurados corretamente na fábrica, mas isto não é uma regra fixa. Quando um fabricante de placas de CPU fornece suas placas, já com o microprocessador e a memória cache instaladas (a DRAM é instalada posteriormente, ou pelo revendedor, ou pelo próprio usuário), todos os jumpers das placas são corretamente configurados na própria fábrica. Existem entretanto fabricantes que vendem suas placas sem o microprocessador e sem a memória cache. O usuário pode comprar, por exemplo em uma loja, uma placa de CPU com o microprocessador e a memória cache instaladas, mas esta instalação pode ter sido baseada apenas no encaixe desses componentes, feito na própria loja, sem a preocupação na correta programação dos jumpers. Isto pode causar o mau funcionamento, e até mesmo a danificação da placa. Por exemplo, podemos instalar um microprocessador que opere com 3,3 volts, e a voltagem pode estar erradamente programada para 5 volts. A regra geral é que, antes de instalar uma placa de CPU, devemos conferir todos os seus jumpers. Placas de expansão que não requerem a instalação adicional de nenhum componente, como é o caso das placas de som e das placas modem/fax, também podem requerer que o usuário configure seus straps, mas nesses casos, uma programação errada pode impedir o funci-

onamento correto, mas não poderá causar nenhum tipo de dano.

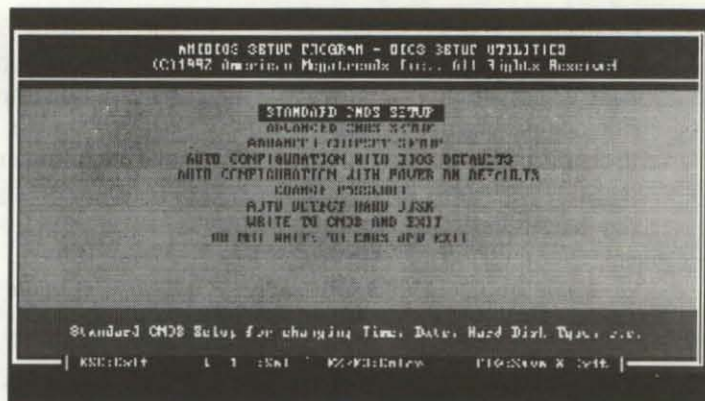
Para programar os jumpers, ou mesmo conferir se a programação vinda de fábrica é adequada, é indispensável o uso do manual que acompanha cada placa. Sem esses manuais pode ser até mesmo impossível realizar a instalação. Esta é uma regra geral para todos aqueles que pretendem realizar expansões de hardware: exigir sempre os manuais de todas as placas e dispositivos a serem instalados.

## CMOS e bateria

As placas de CPU dos micros IBM PC e IBM PC XT possuíam um pequeno componente chamado DIP SWITCH. Tratava-se de um conjunto de oito microchaves que serviam para definir algumas opções de funcionamento a nível de hardware. Essas opções eram:

- O número de drives
- A quantidade de memória RAM
- A presença do coprocessador aritmético
- O tipo de placa de vídeo instalada

A figura 22 mostra o aspecto do DIP SWITCH. Eletricamente, são equivalentes aos jumpers. Sua vantagem é o seu manuseio, muito mais simples. Para alterar posicionamentos de jumpers, em geral é preciso segurá-los usando um alicate, e corremos o risco de deixá-los cair e perdê-los. Já o DIP SWITCH não apresenta essas dificuldades, e podem ser facilmente posicionadas usando, por exemplo, a ponteira de uma caneta. Os micros PC e XT já são peças de museu, mas muitas placas modernas apresentam chaves de configuração do tipo DIP SWITCH, da mesma forma como também apresentam jumpers.



**Figura 24**

**Figura 22**

*Chaves de configuração do IBM PC e do IBM PC XT.*

Os micros da série AT (286 e superiores) também utilizam diversos jumpers para a definição das opções de hardware, mas a maior parte das suas opções de funcionamento ficam armazenadas em uma memória especial, alimentada por uma bateria. Esta memória fica localizada em um chip conhecido como CMOS, mostrado na figura 23.

**Figura 23**

*Exemplo de chip CMOS e sua bateria.*

O chip CMOS possui dupla função:

- Um relógio-calendário
- Uma pequena quantidade de memória RAM

Graças à presença da bateria que mantém o chip CMOS em funcionamento permanente, mesmo quando o computador está desligado, o relógio-calendário do chip CMOS passa o tempo todo marcando horas, minutos, segundos, dias, meses e anos. A memória RAM possui uma capacidade muito pequena, em geral apenas 64 bytes, mas é suficiente para armazenar as diversas opções de funcionamento a nível de hardware. Alguns exemplos dessas opções são:

- A quantidade de memória RAM
- O tipo de microprocessador instalado
- O número e os tipos dos drives de disquete



- Parâmetros do disco rígido
- Parâmetros relacionados com a velocidade de acesso à memória
- Parâmetros relacionados com a velocidade de acesso aos slots
- Senhas
- Habilitação das interfaces existentes na placa de CPU
- Modos de atuação da memória cache

Todos esses itens devem ser programados quando é feita a instalação da placa de CPU. Esta instalação ocorre quando é montado o computador, ou então quando é feita a substituição da placa de CPU. Para realizar esta programação, é usado um programa conhecido como CMOS Setup. Este programa está armazenado na mesma memória ROM onde fica localizado o BIOS da placa de CPU. Em geral, para executar o CMOS Setup, devemos realizar um boot, e durante a contagem de memória, é apresentada uma mensagem indicando qual é a tecla que deve ser pressionada para executar o CMOS Setup. Normalmente esta mensagem tem o seguinte aspecto:

## Press DEL to run Setup

Ao pressionar a tecla indicada (isto precisa ser feito durante a contagem de memória, depois disso a tecla não ativa mais o CMOS Setup, sendo portanto necessário realizar um novo boot para que a contagem de memória apareça novamente), o programa CMOS Setup entra em execução. Seu aspecto pode

variar bastante, dependendo do fabricante, do microprocessador e até mesmo da idade da placa de CPU. Computadores antigos apresentavam Setups que operavam em modo texto, como o que vemos na figura 24.

**Figura 24**  
*Setup em modo texto.*

**Figura 25**  
*Setup em modo gráfico.*

Muitas das placas de CPU atuais apresentam Setups com apresentação gráfica, como o que vemos na figura 25. Os principais fabricantes de Setups são a AMI (American Megatrends, Inc.), a Award e a Phoenix.

## Barramentos

Barramentos são conjuntos de sinais digitais com os quais o microprocessador comunica-se com o seu exterior. Isto inclui:

- Memória
- Chips da placa de CPU (Ex: VLSI)
- Placas de expansão

A maior parte dos sinais digitais que compõem os barramentos são originados no próprio microprocessador, a partir dos seus três barramentos básicos:

- Barramento de dados
- Barramento de endereços
- Barramento de controle

Esses barramentos podem ser combinados de várias formas, dependendo da finalidade. Podemos citar, por exemplo:

- **Barramento local** - Faz a conexão entre microprocessador e memória
- **Barramento ISA** - Constitui os slots de 8 e 16 bits
- **Barramento VLB** - Usado em muitas placas de CPU 486
- **Barramento PCI** - Usado nas placas de CPU Pentium e 486 recentes

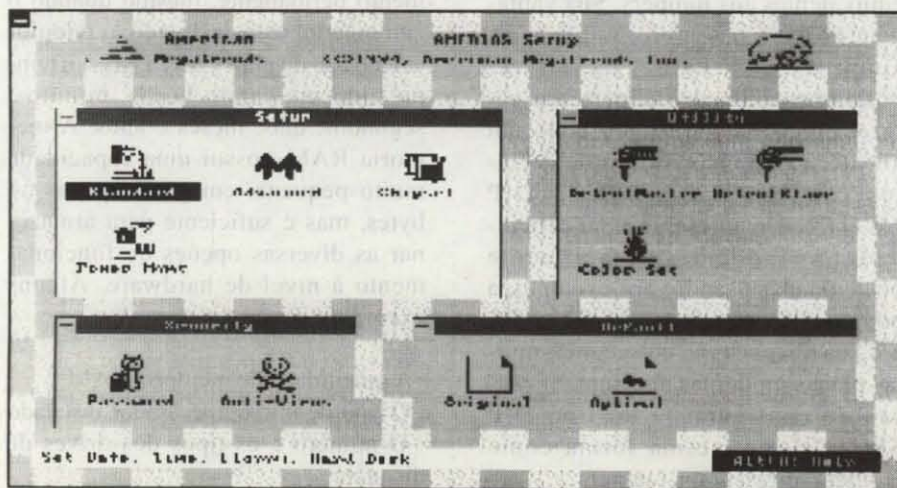
Cada um desses barramentos possui sinais de dados, endereços e controle. Vejamos em que constitui cada um deles.

## ISA

Este barramento é formado pelos slots de 8 e 16 bits existentes nas placas de CPU, e também é usado internamente nessas placas, para a comunicação entre o microprocessador e determinados dispositivos da placa de CPU, como a interface de teclado, controladores de interrupção, timers e diversos outros circuitos. Do ponto de vista do usuário, o que interessa é a presença dos slots de 8 e 16 bits, conhecidos como "Slots ISA". O barramento ISA (Industry Standard Architecture) foi originado no IBM PC, na versão de 8 bits, e posteriormente aperfeiçoado no IBM PC AT, chegando à versão de 16 bits. Possui características herdadas do IBM PC AT, no qual o microprocessador utilizado era o veloz (na sua época) 80286 de 8 MHz. Portanto, suas principais características são:

- Transferências em grupos de 8 ou 16 bits
- Clock de 8 MHz

Cada operação de leitura ou escrita através deste barramento consome no mínimo dois períodos de clock (uma característica do 80286). Portanto, é possível realizar no máximo 4 milhões de transferências por segundo. Como cada transferência pode envolver no máximo 16 bits (2 bytes), a máxima taxa de transferência que pode ser obtida com o barramento ISA é de 8 MB/s.



**Figura 25**



As placas de CPU 286 em geral possuíam vários slots ISA de 16 bits, e alguns slots ISA de 8 bits, sempre apresentando ao todo, no máximo 8 slots. Placas de expansão ISA de 16 bits (Ex: placas de som) devem ser obrigatoriamente conectadas em slots ISA de 16 bits, mas as placas de expansão ISA de 8 bits (Ex: placas modem/fax) podem ser conectadas, tanto em slots de 8 como de 16 bits. Como os slots ISA de 16 bits permitem a conexão,

tanto de placas de expansão ISA de 8 como de 16 bits, muitos fabricantes aboliram totalmente os slots de 8 bits.

As figuras 26 e 27 mostram placas de expansão ISA de 8 e 16 bits, bem como os respectivos slots. Todas as figuras de placas de CPU existentes neste artigo também apresentam slots ISA.

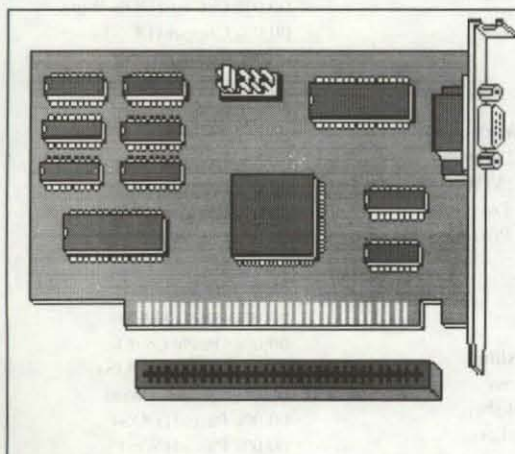


Figura 26

**Figura 26**  
*Uma placa de expansão ISA de 8 bits e um slot ISA de 8 bits.*

**Figura 27**  
*Uma placa de expansão ISA de 16 bits e um slot ISA de 16 bits.*

Apesar de ser considerado lento para os padrões atuais, o barramento ISA ainda é muito utilizado. Mesmo as mais moder-

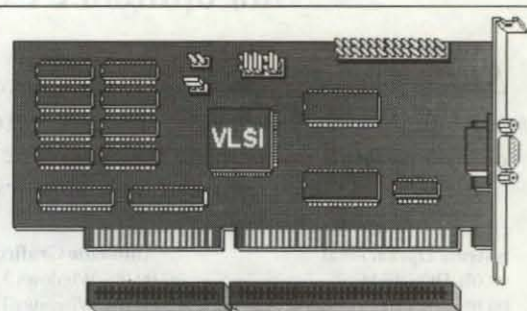


Figura 27

nas placas de CPU Pentium possuem 3 ou 4 slots ISA de 16 bits, nos quais podem ser conectados diversos tipos de placa, para os quais a transferência de 8 MB/s é satisfatória. Podemos citar as placas modem/fax, placas de som e placas de rede, entre diversas outras.

**Laércio Vasconcelos** é Engenheiro Eletrônico e autor de vários livros.

# TECNOCAPAS

**FABRICAMOS CAPAS PARA TODA LINHA DE INFORMÁTICA**

■ Capas para telefones celulares nacionais e importados (em couro)

■ Revenda com personalização:

MOUSE PADS COM BASE DE ESPUMA LATEX.

CONFECCIONAMOS CAPAS SOB MEDIDA E COM VÁRIOS TIPOS DE MATERIAIS.

**FAÇA UM ORÇAMENTO SEM COMPROMISSO:**

**TEL.: (041) 279-1206**



# Perfil do Leitor da Revista Micro Sistemas

Sua opinião é extremamente importante para nós!

**Software** Cada caixa representa uma categoria de itens que você usa ou não em seu micro. Como estamos iniciando a pesquisa, todos os itens têm o percentual 0,0% e encontram-se em ordem alfabética. Para indicar um ou mais itens que você usa em uma categoria, basta riscá-los. Se um item não estiver listado numa categoria, acrescente-o ao final da lista com letra bem legível. Se desejar, acrescente novas categorias e os respectivos itens.

## Sistema Operacional

00,0% DOS-IBM  
00,0% DR-DOS  
00,0% MS-DOS  
00,0% OS-Mac  
00,0% OS2  
00,0% Unix

## Antivírus

00,0% VShield  
00,0% Viruscan  
00,0%

## Processador de Texto

00,0% Carta Certa (DOS)  
00,0% Carta Certa (Win)  
00,0% Fácil (DOS)  
00,0% Fácil (Win)  
00,0% Word (DOS)  
00,0% Word (Win)  
00,0% WordPerfect (DOS)  
00,0% WordPerfect (Win)  
00,0% WordStar (DOS)  
00,0% WordStar (Win)  
00,0% WordPad (Win)  
00,0% Write (Win)  
00,0%

## Ambiente Gráfico

00,0% Windows 3.1  
00,0% Windows 3.11  
00,0% Windows 95  
00,0% Windows NT

## Utilitários Independentes

00,0% Norton  
00,0% PCTools  
00,0% XTGold  
00,0%

## Editor de Desenho/Foto

00,0% Corel Photo Paint  
00,0% Paint Shop Pro  
00,0% Paintbrush  
00,0% Photo Shop  
00,0%

## Cad / Editor 3D

00,0% 3D Studio  
00,0% AutoCad (DOS)  
00,0% AutoCad (Win)  
00,0% MicroStation  
00,0%

## Compactador

00,0% Arj (DOS)  
00,0% Arj (Win)  
00,0% Lha (DOS)  
00,0% PKZip (DOS)  
00,0% PKZip (Win)

## Planilha

00,0% Excel (Win)  
00,0% Lotus 123 (DOS)  
00,0% Lotus 123 (Win)  
00,0% Quattro Pro (DOS)  
00,0% Quattro Pro (Win)  
00,0%

## Desktop Publishing

00,0% CorelDraw  
00,0% MS Publisher  
00,0% Page Maker  
00,0% Ventura  
00,0%

## Linguagem de Programação

00,0% Access (Win)  
00,0% Assembler  
00,0% Basic (DOS)  
00,0% C, C++ (DOS/Win)  
00,0% Clipper (DOS)  
00,0% Cobol (DOS)  
00,0% Cobol (Win)  
00,0% dBase (DOS)  
00,0% dBase (Win)  
00,0% Delphi (Win)  
00,0% FiveWin (Win)  
00,0% Fox Pro (DOS)  
00,0% Fox Pro (Win)  
00,0% HTML (Win)  
00,0% Java (DOS/Win)  
00,0% Oracle (DOS)  
00,0% Oracle (Win)  
00,0% Paradox (DOS)  
00,0% Paradox (Win)  
00,0% Pascal (DOS)  
00,0% Pascal (Win)  
00,0% QBasic (DOS)  
00,0% Visual Basic (Win)

## Hardware

### Tipo de Micro

☐ Mac  
☐ MSX  
☐ PC-AT  
☐ PC-XT  
☐ Pentium  
☐  
☐

CPU (Mhz)  
RAM (Mb)  
HD (Mb/Gb)  
Fax/Modem (bps)  
Multimídia (Vel.)  
No-break (KVA)  
Fita Strimmer (Gb)

### Tipo de Impressora

☐ Jato de Tinta ☐ Color  
☐ Laser ☐ P&B  
☐ Matricial

### Tipo de Scanner

☐ De Mesa ☐ Color  
☐ Manual ☐ P&B  
\_\_\_\_\_ dpi

## Internet / BBS

### Navegador Web

00,0% Microsoft Internet Explorer  
00,0% Mosaic  
00,0% Netscape Navigator 00,0%

### Acesso | Internet | BBS

☐ Baixar aplicativos  
☐ Baixar fotos  
☐ Baixar fotos pornôs  
☐ Bater papo com amigos  
☐ Compras  
☐ E-mail  
☐ Fazer novas amizades  
☐ Fazer sexo virtual  
☐ Namoro virtual  
☐ Participar de workgroup  
☐ Pesquisa escolar  
☐ Pesquisa profissional  
☐

## Sobre MS e Você

01 - Como soube da MS?

☐ Banca ☐ Internet ☐ Indicação de ☐ Amigo ☐ de Professor

02 - Quando compra MS?

☐ Sou assinante ☐ Todo mês ☐ Quando o assunto me interessa

03 - Gostou desse exemplar? ☐ Sim ☐ Não

04 - Se não, o que mudaria nele?

05 - O que gostaria de ver nas próximas edições?

06 - Quais os artigos, colunas, seções, séries que mais gosta?

Envie sua resposta para um dos  
endereços constantes  
da página com o anúncio  
"Procura-se Talentos"



# CURSO

## *FiveWin e a Migração de Aplicações Clipper para o Windows*

### *Parte 3 de 5*

Por: Vilian F. Arraes

*Como o próprio leitor pode observar, Vilian F. Arraes expõe de forma didática e objetiva os recursos que você poderá obter usando o FiveWin.*

*O curso introdutório sobre o FiveWin iniciou na página 14 da Micro Sistemas 163. A parte 2 do presente curso, que em princípio está dividido em 5 partes, foi publicada na edição 166. Veja na página com o anúncio "O Melhor da Informática Nacional" como adquirir números anteriores.*

Uma aplicação for Windows tem que ter muito mais que apenas uma interface gráfica. Ela deve proporcionar aos seus usuários níveis de controle e flexibilidade compatíveis com o ambiente Windows. Isso acaba gerando muita programação modular, obrigando o programa a ser executado em pequenas unidades, que podem ser iniciadas e finalizadas a qualquer momento. A **orientação a objeto** é intrinsecamente modular sendo portanto o caminho natural para o desenvolvimento de aplicativos For Windows. Objetos, Classes, métodos, são conceitos que o programador moderno não pode ignorar.

O CLIPPER desde a versão 5.01 passou a disponibilizar para os programadores um pequeno, mas poderoso conjunto de classes, objetos e ferramentas para manipulação desses objetos. Conjunto este incrementado pelo FiveWin com classes que possibilitam a migração das aplicações CLIPPER para o ambiente Windows de modo simples e eficaz.

Em nosso curso já foram utilizados alguns tipos de objetos das classes disponibilizadas pelo FiveWin (Exemplo: Janelas, Menus, etc).

Dando continuidade à implementação do nosso SCC - **Sistemas de Controle de Clientes**, utilizaremos um objeto de uma das suas mais poderosas e versáteis classes: A **LISTBOX** cuja função assemelha-se a da classe **TBROWSE** do CLIPPER, para implementar a manutenção do arquivo de clientes. Com isto exibiremos o conteúdo do arquivo de clientes em um browser, com botões correspondentes a cada uma das operações de manutenção (inclusão, exclusão, etc). Observe o código fonte (listagem).

Estudando o código fonte apresentado, constataremos os seguintes comandos ainda não comentados:



```

PROCEDURE SCC100
LOCAL oDlg, oLbx
USE clientes ALIAS cli NEW EXCLUSIVE //Abre arquivo de clientes
IF .NOT. File("CLIENTES.NTX")
INDEX ON cli->codigo TO clientes
ELSE
SET INDEX TO clientes
ENDIF
GO TOP

DEFINE DIALOG oDlg; //Define dialog a ser exibido
FROM 2,0 TO 17,64 ;
TITLE "Manutenção do cadastro de clientes" ;
OF oWnd

@ 0.5,0.5 LISTBOX oLbx ; //Define browser para o arquivo CLIENTES.DBF
FIELDS cli->codigo,cli->nome ;
FIELDSIZES 56, 320 ;
HEADERS "CÓDIGO", "NOME DO CLIENTE" ;
SIZE 195, 100 ;
OF oDlg

@ 1.0,29 BUTTON "&Incluir" SIZE 40,10 OF oDlg //Define botões pra janela
@ 2.5,29 BUTTON "&Alterar" SIZE 40,10 OF oDlg
@ 4.0,29 BUTTON "&Excluir" SIZE 40,10 OF oDlg
@ 5.5,29 BUTTON "Im&primir" SIZE 40,10 OF oDlg
@ 7.0,29 BUTTON "&Fechar" SIZE 40,10 ACTION ( oDlg:End() ) OF oDlg

ACTIVATE DIALOG oDlg CENTERED //Ativa a janela de dialogo
RETURN

```

comando especificadas as coordenadas, o título, tamanho e a ação a ser executada quando o botão for selecionado. No programa apresentado apenas o último botão tem uma ação definida. Vamos estudá-lo melhor:

```
@ 7.0,29 BUTTON "&Fechar" SIZE 40,10 ACTION ( oDlg:End() ) OF oDlg
```

É definido um botão a ser exibido a partir das coordenadas **7.0,29** com **"&Fechar"** como título, **40,10** como tamanho em pixels e **(oDlg:End())** como ação a ser executada quando o mesmo for acionado.

**ACTIVATE DIALOG**, aciona a janela de diálogo definida em **oDlg**. A cláusula **CENTERED** leva o Windows a exibir a janela centralizada dentro da janela principal;

Note que o tamanho dos campos no comando **LISTBOX**, sua própria dimensão e o tamanho dos botões são definidos em **pixels** (pontos). As janelas do ambiente Windows são formadas por pontos, normalmente 640 x 480. Observe ainda que as coordenadas de posicionamento podem ser fracionadas (Ex: 0.5, 2.5, etc), sendo internamente convertidas para o corresponden-

te em pontos pelo próprio FIVEWIN. Na verdade poderíamos inclusive já indicar todas as coordenadas também em pixels bastando para isso adicionar a cláusula **PIXEL** no final de cada comando.

Para que possamos observar o browser sendo executado na nossa aplicação, devemos alterar o trecho do código fonte que implementa a opção "Clientes" do seu menu para o seguinte:

```

MENUITEM "&Clientes..." ;
ACTION Scc100( ) ;
MESSAGE "Manutenção do cadas-
tro de clientes"

```

Efetuada todas as alterações e gerado um novo executável com o arquivo **BUILD.BAT**, sua execução e o acionamento da opção "clientes" do menu "Arquivos" já nos permitirá visualizar a janela da figura 1:

É importante ressaltar, que em todos os programas exemplos procura-

· O comando **DEFINE DIALOG** define em **oDlg** uma janela do tipo diálogo, com **2,0** como coordenadas do canto superior esquerdo, **17,64** como coordenadas do canto inferior direito e a mensagem **"Manutenção do cadastro de clientes"**, como título do diálogo. A cláusula **OF** é utilizada para especificar a subordinação deste diálogo a janela definida em **oWnd**. As janelas do tipo diálogo são utilizadas para interação com o usuário:

· O **@ LISTBOX** define um browser sobre o arquivo de clientes, com o canto superior esquerdo em **0.5,0.5**, sendo exibidos na janela os campos **cli->codigo** e **cli->nome** com tamanhos de colunas definidos em **56** e **320** respectivamente. A cláusula **HEADERS** é utilizada para especificar os títulos das colunas e a cláusula **SIZE** o tamanho total do browser;

· **@ BUTTON** define botões de comando para a janela indicada por **oDlg**, sendo no

mos elaborar o código fonte da forma mais simples possível, muitas vezes ignorando importantes recursos do **CLIPPER**, para nos concentrarmos com maior profundidade nos recursos do **FIVEWIN** e do ambiente **Windows**.

Até a próxima.

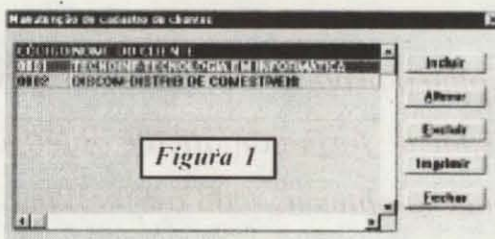


Figura 1

**Vilian F. Arraes**, Tecnólogo em Processamento de Dados pela UFPa, atua há dez anos na área de informática, tendo desempenhado funções de treinamento, análise e programação de sistemas. Atualmente é sócio e gerente de desenvolvimento da **TECNOINF Sistemas Ltda.**  
e-mail: vilian@expert.com.br



# ARTIGO

## *Menu ..... Fácil ..... Para ..... Pascal .....*

*Por: Gilberto Vincentin*

Um dos mais terríveis males que atingem os programadores das mais diversas linguagens, é criar procedimentos e funções que sejam rápidas e versáteis, podendo se adaptar facilmente à qualquer programa desejado, sem precisar fazer demoradas alterações.

Essa rotina não deve se tornar lenta em nenhum tipo de PC, por mais lento que ele seja, pois foi desenvolvida em um idoso XT Turbo de 12,5 MHz, e ele rodou sem nenhum problema ( eu até me surpreendi ).

Para obter um resultado, em tela, mais limpo, coloquei um procedimento para "sumir" com o cursor (o que dá um acabamento incrível) e outro para desenhar uma moldura retangular na tela e centralizar seu título.

Aproveitei a oportunidade para apresentar aos programadores a esquecida posição (80,25) da tela, muitas vezes necessária, porém inutilizada por causa do temível rolamento de tela. No procedimento "WRITEXY", você poderá utilizar essa posição sem medo, pois ele faz a escrita sem que o cursor se mova.

A rotina em geral é de fácil utilização, definem-se as posições e os títulos do menu, a ultima posição a ser usada e a posição de inicio. A função "ESPERA" faz a troca das posições no menu com as setas, e quando outra tecla for pressionada ela retorna com o seu valor na tabela ASCII. Os principais valores de retorno serão :

Ü #13 - Para -> <ENTER>

Ü #27 - Para -> <ESC>

Ü ou qualquer outra letra, número e código de teclado.

As cores utilizadas no menu são definidas como constantes no inicio da Unit, sendo fácil para alterá-las. Ahh ! Existe ainda um procedimento para alterar cor de escrita e de fundo, é super-simples, mas pode poupar muitas linhas do programa fonte.

Bom, agora é só digitar. Mãos à obra.

**Gilberto Vicentin Somacal**, tem 16 anos, está cursando o 2º ano Científico, programa em Pascal e se defende em Clipper, Basic e C.

P.S Gostaria de solicitar por algum bondoso programador aficionado em Pascal, que tenha algum procedimento ou função que faça acesso à memória de video, para manipulação de telas gráficas. Agradeço antecipadamente.

O Meu endereço de contato é :

Gilberto Vicentin Somacal

Caixa Postal nº 32

CEP 89500-000 Caçador - SC



## Listagem 1

```
Unit Menu;

Interface

Uses Crt, Dos;

Const
  Lig = True;
  Des = False;
  CTH = LightBlue;
  CFH = Cyan;
  CTL = DarkGray;
  CFL = LightGray;
  MAX = 15;
  TAM = 15;

Type
  Escolha = Record
    PosX : Byte;
    PosY : Byte;
    Titulo : String[ TAM ];
  End;

  TipoTela = Array[ 1..MAX ] Of Escolha;

Var
  Ultimo : Byte;
  Atual : Byte;
  Tela : TipoTela;
  Reg : Registers;

Procedure Cor( T, F : Byte );
Procedure WriteXY( X, Y : Byte ; Frase : String );
Procedure Moldura( X1, Y1, X2, Y2 : Byte ; TT : String );
Procedure Cursor( Lig_Des : Boolean );
Procedure Inverte( Func : Byte );
Procedure Mostra;
Function Espera : Char;

Implementation

Procedure Cor( T, F : Byte );
Begin
  TextColor( T );
  TextBackground( F );
End;

Procedure WriteXY( X, Y : Byte ; Frase : String );
Var
  Nada : Byte;

Procedure Car( CC : Char );
Begin
  DirectVideo := False;
  Reg.ah := $9;
  Reg.al := Ord( CC );
  Reg.bh := $00;
  Reg.bl := TextAttr;
  Reg.cx := 1;
  Intr( $10, Reg );
  DirectVideo := True;
End;

Begin
  For Nada := 1 To Length( Frase ) Do
    Begin
```

```
      GotoXY( X, Y );
      Car( Frase[ Nada ] );
      X := X + 1;
    End;
  End;

Procedure Moldura( X1, Y1, X2, Y2 : Byte ; TT : String );
Var
  Lixo1 : Byte;
Begin
  For Lixo1 := X1 To X2 Do
    WriteXY( Lixo1, Y1, #205 );
  For Lixo1 := X1 To X2 Do
    WriteXY( Lixo1, Y2, #205 );
  For Lixo1 := Y1 To Y2 Do
    WriteXY( X1, Lixo1, #186 );
  For Lixo1 := Y1 To Y2 Do
    WriteXY( X2, Lixo1, #186 );
  WriteXY( X1, Y1, #201 );
  WriteXY( X2, Y1, #187 );
  WriteXY( X1, Y2, #200 );
  WriteXY( X2, Y2, #188 );
  IF TT <> '' Then
    Begin
      WriteXY(((X1+X2) Div 2) - ((Length(TT) Div 2) + 2),
        Y1, #181 + ' ' + TT + ' ' + #198 );
    End;
  End;

Procedure Cursor( Lig_Des : Boolean );
Begin
  IF Lig_Des Then
    Begin
      Reg.ah := 1;
      Reg.cl := 7;
      Reg.ch := 0;
      Intr( $10, Reg );
    End
  Else
    Begin
      Reg.ah := 1;
      Reg.cl := 0;
      Reg.ch := 7;
      Intr( $10, Reg );
    End;
  End;

Procedure Inverte( Func : Byte );
Begin
  Cor( CTL, CFL );
  IF Func = Atual Then
    Cor( CTH, CFH );
End;

Procedure Mostra;
Var
  Lixo : Byte;
Begin
  For Lixo := 1 To Ultimo Do
    Begin
      Inverte( Lixo );
      WriteXY( Tela[ Lixo ].PosX, Tela[ Lixo ].PosY, Tela[ Lixo ].Titulo );
    End;
  End;

Function Espera : Char;
Var
  Carac : Char;
```



## ARTIGO / Menu Fácil Para Pascal

```
Begin
Cursor( Des );
Carac := #0;
While Carac in [#0] Do
Begin
Mostra;
While Not KeyPressed Do;
While KeyPressed Do
Carac := ReadKey;
IF Carac = #72 Then
Begin
IF Atual > 1 Then
Atual := Atual - 1
Else
Atual := Ultimo;
Carac := #0;
End;
IF Carac = #80 Then
Begin
IF Atual < Ultimo Then
Atual := Atual + 1
Else
Atual := 1;
Carac := #0;
End;
End;
Espera := Carac;
Cursor( Lig );
End;
End.
```

### Program Exemplo\_da\_Utilizacao\_de\_Menus;

```
Uses Menu, Crt;
Const
Frase : Array[ 1..5 ] Of String[ 10 ] =
('Exemplo','da','Utilizacao','de','Menus');
Var
Cores : Byte;
Begin
Cor( 14, 1 );
ClrScr;
Moldura( 1, 1, 80, 25, 'Exemplo 1' );
Cores := 2;
Tela[ 1 ].PosX := 34;
Tela[ 1 ].PosY := 6;
```

```
Tela[ 2 ].PosX := 34;
Tela[ 2 ].PosY := 8;
Tela[ 3 ].PosX := 34;
Tela[ 3 ].PosY := 10;
Tela[ 4 ].PosX := 34;
Tela[ 4 ].PosY := 12;
Tela[ 5 ].PosX := 34;
Tela[ 5 ].PosY := 14;
Tela[ 6 ].PosX := 34;
Tela[ 6 ].PosY := 16;
Tela[ 7 ].PosX := 34;
Tela[ 7 ].PosY := 18;
Tela[ 1 ].Titulo := 'Exemplo';
Tela[ 2 ].Titulo := 'da';
Tela[ 3 ].Titulo := 'Utilizacao';
Tela[ 4 ].Titulo := 'de';
Tela[ 5 ].Titulo := 'Menus';
Tela[ 6 ].Titulo := 'Cor';
Tela[ 7 ].Titulo := 'Sair';
Atual := 1;
Ultimo := 7;
Moldura( 32, 4, 47, 20, '' );
While Atual <> Ultimo Do
Begin
IF Espera = #13 Then
Begin
Cor( Cores, 4 );
IF Atual = 1 Then
WriteXY( 24, 22, Frase[ 1 ] );
IF Atual = 2 Then
WriteXY( 32, 22, Frase[ 2 ] );
IF Atual = 3 Then
WriteXY( 35, 22, Frase[ 3 ] );
IF Atual = 4 Then
WriteXY( 46, 22, Frase[ 4 ] );
IF Atual = 5 Then
WriteXY( 49, 22, Frase[ 5 ] );
IF Atual = 6 Then
Begin
IF Cores < 15 Then
Cores := Cores + 1
Else
Cores := 0;
End;
End;
Cor( 7, 0 );
ClrScr;
End.
```

**VOCE FICARIA DE BOCA ABERTA SE SOUBESSE  
DE QUANTO PRECISA PARA  
PREENCHER ESTE ESPAÇO COM O ANÚNCIO  
DE SUA EMPRESA**

**REVISTA MICRO SISTEMAS TEL: (021) 589-2088**



# PROCURA-SE TALENTOS

## Dê uma chance ao sucesso liberando seus bytes

Se você passou horas ou dias, ou por acaso descobriu aquela super-dica sobre uma rotina de uma linguagem de programação, ou sobre a Internet, a Multimídia, o Windows, o OS2 ou o DOS, ou sobre uma planilha, editor de texto, gerenciador de banco de dados, ou editor gráfico que você usa com frequência, e portanto, **domina**, divulgue seu conhecimento pela Micro Sistemas.

A Micro Sistemas é a primeira Revista Brasileira (PRB) de informática e é a única publicação nacional que abre espaço para trabalhos feitos no Brasil, por leitores e para leitores. Esta é a principal razão do sucesso imbatível da revista. Afinal, grande parte dos profissionais com projeção neste mercado, foram ou ainda são colaboradores da Micro Sistemas.

### *O que você está esperando para fazer parte deste time?*

Para que seu trabalho (artigo, crônicas, curso, dicas, programas, rotinas, etc) seja avaliado por nossa equipe é imprescindível que:

- A matéria venha impressa (texto e figuras) e também gravada em disquete;
  - As figuras sejam gravadas separadamente;
- Os programas e as rotinas venham com uma versão compilada com exemplos;
  - O disquete seja padrão IBM PC (disco 5 1/4" ou 3 1/2");
  - O texto seja em formato Word for Windows.

Se aprovada por nossa equipe, a matéria só será publicada se acompanhada de autorização do autor. A autorização é dada quando o autor define o texto de fechamento da matéria, ou seja, nome, endereço, e reduzido histórico de suas atividades. Sempre indicar se deseja a publicação de dados pessoais, tais como, telefone, caixa postal, nome da empresa, e-mail, etc.

O material enviado para a revista não será devolvido.

Os autores das matérias publicadas, além de estarem divulgando conhecimento para milhares de leitores, receberão inteiramente grátis uma assinatura anual de Micro Sistemas, ou um pagamento simbólico, por matéria publicada a título de agradecimento.

**Enviar o material para:**

**REVISTA MICRO SISTEMAS**

Caixa Postal 18347, CEP: 20722-970, RJ - RJ

Tel. (021)532-0770 cod. 4016779

e-mail: [ricardoflores@openlink.com.br](mailto:ricardoflores@openlink.com.br)





# ARTIGO

## *Informática na Medicina*

*Por: Renato Francisco Gastaldelo Amaral*

*Prezado leitor, este artigo de Renato Amaral resume a situação atual da informática no Brasil, ou seja, pouca gente que possui a mídia na mão enganando milhares de pessoas que estão recém entrando na informática. Com certeza, ao ler este artigo você vai se lembrar de alguém de seu ciclo de amizade que recentemente adquiriu um micro e pensa que está tudo resolvido...*

A informatização de clínicas e consultórios médicos representa um aumento de produtividade para os facultativos e um melhor atendimento para os seus pacientes.

Não basta colocar um PC no consultório do esculápio e esperar que o microcomputador faça tudo sozinho. É preciso haver critério na escolha do equipamento, dos programas e promover um treinamento satisfatório do médico e de suas secretárias.

Muitos analistas de sistemas cansaram de ver profissionais da área de saúde que têm micros em seus consultórios, mas que os usam apenas como uma cara máquina de escrever ou sequer os empregam. Já presenciamos situações muito estranhas:

1) Um médico que usava uma planilha de cálculos para redigir seu receituário. A receita emitida não era armazenada no micro por falta de software adequado. Não havia qualquer forma de efetuar um "follow-up" do paciente. Todas as informações relevantes (história do paciente, achados do exame físico, medicamentos receitados, resultados laboratoriais) eram arquivadas manualmente como há cinquenta anos atrás. O micro servia apenas como uma cara máquina de escrever de US\$ 3000,00.

2) Já vimos três médicos que possuíam sistemas em suas instalações, mas que não os utilizavam. Eram usuários frustrados, perplexos, que não sabiam trabalhar direito com os micros, que desconfiavam das máquinas porque não foram treinados como manda o figurino. Eles caíram na conversa de vendedores inescrupulosos que tão-somente queriam um lucro imediato, prometiam mundos e fundos, mas, que na hora de treinar os compradores, sumiam do mapa e os deixavam desamparados.

Para evitar a desagradável realidade expressa na segunda situação, os doutores devem haver-se com profissionais sérios de informática e que sejam autênticos analistas de sistemas. O treinamento é muito importante e isso os vendedores não podem fornecer. Tudo o que eles fazem é decorar um texto,



como se fosse um papagaio inconsciente do que fala, e o cliente que se lixe para aprender a mexer com o sistema!

Trabalhar com PCs não é café pequeno. A instalação de certos programas e de algumas placas pode encerrar verdadeiras armadilhas para os neófitos. Um programa pode entrar em atrito com o driver do mouse ou do CD-ROM e gerar um conflito de interrupções. A instalação de um scanner de mesa pode comprometer o sistema como um todo. A placa de rede pode dar defeitos cuja resolução seja altamente complexa.

A tendência é a de que o PC torne-se um eletrodoméstico, mas ele ainda está longe de ser fácil de operar como um liquidificador. O mito da facilidade de operação, instalação e conserto dos computadores é divulgada por vendedores ignorantes que são bons apenas em memorizar o seu "sales speech" e em pegar a grana dos pobres consumidores.

Os médicos devem precaver-se contra tais trogloditas falantes e buscar ajuda nos excelentes analistas de sistemas como os que, por exemplo, graduaram-se no excelente Curso Superior de Processamento de Dados da FATEC-SP (Faculdade de Tecnologia de São Paulo) e em outros ótimos cursos superiores de computação que existem Brasil afora (USP, ITA, UNICAMP, Mackenzie, etc).

### *Veremos, a Seguir, de que Modo a Informática Facilita a Vida dos Doutores*

#### **1) Recuperação rápida e segura dos dados administrativos, clínicos laboratoriais do paciente.**

Existem sistemas de atendimento médico como aqueles desenvolvidos por mim e minha equipe que possuem uma agradável, intuitiva e divertida interface gráfica. A atividade primária de um sistema desses é armazenar informações sobre os pacientes e recuperá-las rapidamente, mesmo que o médico tenha já atendido mais de dez mil clientes. O nosso SAM (Sistema de Atendimento Médico) foi desenvolvido por um grupo

de pessoas seletas: médicos da Escola Paulista de Medicina – Universidade Federal de São Paulo (EPM–UNIFESP) e analistas de sistemas do Curso Superior de Processamento de Dados da FATEC-SP (Faculdade de Tecnologia de São Paulo).

#### **2) Facilidade de backup.**

Fazer um backup é providenciar para que haja uma cópia dos arquivos que armazenam as informações. Já imaginou tirar cópias xerox de um sistema de fichário com mais de três mil clientes? É uma loucura! No entanto, com um bom sistema de atendimento médico, o doutor realiza backups rapidamente em disquetes ou CD-Rs. Se dispuser de uma rede, pode simplesmente copiar os arquivos para um diretório de backups presente no micro da secretária, por exemplo.

#### **3) Programas de anatomia humana.**

Existem programas de anatomia humana tais como o BodyWorks e o ADAM, muito úteis para os pacientes aprenderem rudimentos anatômicos e assim melhor compreenderem as doenças que os afligem. Por exemplo: um gastroenterologista pode mostrar aos seus pacientes como é o estômago por dentro e assim melhor explicar o surgimento das úlceras gástricas.

#### **4) Home page na Internet.**

O médico pode possuir uma home page na Internet contendo uma foto dele, informações sobre sua formação acadêmica, pós-graduação, especializações, mestrado e áreas de atuação (gastroenterologia, medicina interna, endocrinologia, etc). Trata-se de uma espécie de currículo eletrônico.

Poderá haver também uma seção de perguntas e respostas para os pacientes, nos moldes das FAQs (Frequently Asked Questions) dos sites norte-americanos. Tais questões são aquelas geralmente levantadas no dia-a-dia do consultório. Ao invés de o médico repetir ad nauseam as mesmas ladainhas, bastará referenciar o doente para que ele obtenha maiores informações na citada HP profissional.

Tais home pages geralmente disponibilizam uma versão em língua inglesa para os webmaníacos estrangeiros e amigos forasteiros do doutor.

#### **5) Correio eletrônico (e-mail).**

Se um internauta quiser entrar em contato com o doutor, poderá fazê-lo por meio do envio de um e-mail. O correio eletrônico é conveniente porque o médico responderá somente quando existir tempo. Não há necessidade de respostas imediatas. O médico lida com vidas humanas e precisa de tranquilidade para bem desempenhar o seu papel.

Telefones, bips, pagers e cartões de ponto não combinam com a carreira médica. É altamente insensato exigir que o médico interrompa seu trabalho para responder alguma pergunta que poderá ser dirimida posteriormente. É por isso que o correio eletrônico disponibilizado pela formidável realidade que é a Internet presta-se muito bem para o citado propósito de entrar em contato com o doutor sem atrapalhar-lhe a vida.

#### **6) Busca de informações científicas na Internet.**

Um dos propósitos mais nobres da Internet é a divulgação científica. Os médicos possuem dois endereços virtuais excepcionalmente bons para buscar informações sobre doenças raras ou complicadas, novos medicamentos, novas terapias, novas técnicas cirúrgicas, etc. Ei-los aqui:

O mwsearch é o medical world search, uma ferramenta de busca exclusivamente voltada para o mundo médico. É o paraíso dos doutores internautas.

Já o medscape é um site com múltiplos artigos médico-científicos. O pessoal do medscape oferece a possibilidade de enviar e-mails semanais contendo informações sobre novos artigos do citado gênero. Basta que o navegante preencha um formulário e forneça o seu endereço de correio eletrônico. Trata-se, como podemos ver, de um site de primeira grandeza para os profissionais da área de saúde.



O site dos National Institutes of Health, mantidos pelo governo norte-americano, costumam figurar nas listas do msearch e é muito bom também.

As famosas e clássicas Faculdades de Medicina de Harvard, Yale e Johns Hopkins, situadas nos EUA, possuem ótimos sites na Internet.

## 7) Reciclagem dos conhecimentos médicos via Internet.

Certos sites da Internet formulam casos clínicos hipotéticos, com fotografias do paciente fictício, imagens de raios-X, ultra-som, tomografia, ressonância, doppler e resultados dos exames laboratoriais (glicemia, cortisol, hemograma completo, níveis de colesterol, etc).

Os doutores navegantes são encorajados a arriscar um diagnóstico e estabelecer condutas de tratamento para o "paciente".

Uma outra tela do site fornece as respostas corretas. É uma boa maneira do médico treinar e reciclar seus conhecimentos sem matar os doentes.

## 8) Participação em fóruns médicos internacionais sem precisar viajar.

A Internet brinda os médicos com a fabulosa possibilidade de encontrar colegas de profissão no estrangeiro. Existem facultativos que discutem seus casos mais graves com especialistas de primeira grandeza da Alemanha, dos EUA, do Canadá, da França, da Suécia, do Japão e de outros grandes centros da medicina mundial.

O doutor surfista assim participa de fóruns médicos no exterior sem a necessidade de tirar seus pés do Brasil.

## 9) Sistemas de reconhecimento de voz ("speech recognition").

Existem sistemas de "speech recognition" que captam o que o médico fala, sem a necessidade de digitar algo no teclado do micro. Isso facilita sobremaneira a vida do doutor, pois ele fica com as mãos livres para realizar o exame físico do paciente. Enquanto examina-o, o facultativo vai falando ao computador o que encontra: "figado palpável a cinco dedos do rebordo costal direito, consistência endurecida, bordo irregular, superfície nodular, talvez seja uma cirrose crônica. Achados físicos consistentes com a anamnese. Baço palpável, aumentado de tamanho ++++. Presença de ascite no abdômen ++++. Provável hipertensão portal. Tudo decorrente da cirrose hepática. Observam-se vasos aumentados na região umbilical."

O médico realiza suas anotações/apontamentos enquanto examina o paciente. Não é preciso parar para ficar ralando os dedos no teclado do PC. Somente com isso, o facultativo economiza cerca de cinco minutos, os quais serão melhor gastos com o paciente. A produtividade médica melhora em cerca de 20-30%. E o doente sente-se mais amparado, porque o esculápio dispensa-lhe uma maior atenção.

Temos usado com sucesso um software da IBM chamado VoiceType Simply Speaking para Windows 95.

### Resumo:

Não basta enfiar um microcomputador no consultório do médico e dizer a ele: "pronto! Agora, vire-se sozinho! Torne-se um Ph.D. em compu-

tação para poder usar o sistema... Adeus! Até nunca mais!"

Essa atitude bastante questionável, tomada por vendedores inescrupulosos do setor de informática, deixou desamparados os doutores pioneiros na introdução de PCs em seus consultórios e os demais facultativos ressabiados.

Os médicos desejosos de aumentar a sua produtividade e ficar com mais tempo para atender aos seus pacientes devem procurar profissionais sérios que realmente dominem computação e que lhes proporcionem um senhor treinamento, além da certeza de que não ficarão desamparados no futuro quando houver um problema qualquer.

A classe médica é muito laboriosa e necessita de uma atenção especial devido à falta de tempo. Necessita de um treinamento dirigido e rápido, além de analistas de sistemas de confiança que estarão lá quando o vento soprar mais forte.

*Renato F. G. Amaral é analista de sistemas e web designer com quinze anos de experiência em programação. Desenvolve software para clínicas e consultórios médicos na plataforma Visual Basic 5.0. Foi clípeiro durante sete longos anos. Domina Visual Basic, Turbo Pascal, Clipper/dBASE IV, C/C++, QuickBASIC e ferramentas CASE. Atualmente faz o Curso Superior de Processamento de Dados da FATEC (Faculdade de Tecnologia de São Paulo). Seu hobby é jogar xadrez, nadar, surfar na Internet e adora escrever como passatempo. Seu e-mail para trocar idéias e informações é renatofa@dedalus.net.*

## ANÚNCIE AQUI

### 4,5 X 18 cm

## APENAS R\$ 100,00

Informações pelo telefone (021) 589-2088



# ARTIGO

## WordStar to ..... Windows ..... (WS2WIN) .....

Por: Carlos Roberto Macedônio

*O primeiro editor de texto muito usado em micros foi o Magic Windows. Galera, isso ocorreu há mais de 15 anos. Era uma febre!!! O Magic Windows vinha gravado em fita K7 e para carregá-lo eu usava um gravador K7 ligado ao micro. Surgiram as placas CP/M que possibilitaram várias melhorias nos micros da época. Uma das mais importantes foi a possibilidade dos microiros migrarem para um editor de texto AT, ou seja, Actual Technology [Tecnologia Atual] com um nome muito inponente "O Astro da Palavra" - o famoso WordStar. Nas primeiras versões do WordStar para se produzir um acento e palavras com "ç" exigia do microiro um verdadeiro malabarismo digital. Houve até uma versão denominada "WordStar 2000". Sem a menor sombra de dúvida o WordStar reinou muitos anos quase que absoluto, até que surgiu o Windows. E aí? O que fazer com todos os arquivos de relatórios, cartas, memorandos, etc, tudinho feito em WordStar?*

### Introdução

O WordStar foi muito provavelmente um dos mais importantes processadores de textos, e com ele era feito de tudo no escritório, as cartas mais curtas até os mais longos tratados. Muito fácil de usar e trazendo do CP/M toda uma clientela, tornou-se realmente uma praga, porém, um belo dia, chega o Windows, com o Word, todo gráfico, pomposo, "rápido", coqueluche pura. Tomou o lugar do WS, e gerou uma pergunta. O que fazer com os textos criados no WS? O Word for Windows possui um conversor. Fim do problema. Médio. Onde estão os acentos? Este mundo de espaços, o que eu faço com eles?

Para minimizar o tempo gasto no êxodo do WordStar 3.3 para o Word for Windows, desenvolvi um conversor chamado WS2WIN - WordStar to (2) WINDows, que limpa o texto gerado pelo WS 3.3 e gera um arquivo no formato texto (TXT) do Windows conservando as acentuações, permitindo assim sua posterior edição no Write, Word 2.0 e Word 6.0. O programa foi desenvolvido utilizando o Turbo C 2.0, com modelo de memória SMALL.

### Funcionamento

Para gerar um caractere acentuado, o WordStar utiliza o recurso de retorno da cabeça de impressão. Assim, para gerar o cedilha minúsculo (ç), ele envia à impressora o caractere c, o comando de retorno do carro (8), e em seguida o caractere vírgula (,) que devido ao retorno do carro será impresso sobre o caractere c, causando a impressão de haver ali o caractere cedilha.

Outra manipulação importante feita pelo WordStar é a ligação das linhas de um mesmo parágrafo, que é feita através da alteração do byte CR (carry return), no qual é ligado o oitavo bit, passando 0x0D (13) para 0x8D (141), conservando o 0x0A (LF ou Line Feed). Processando esta sequência o programa consegue manter a ligação lógica entre os parágrafos.

O Word possui ferramentas bastante potente para a pós-produção (após a digitação) do texto, por exemplo: recuos, tabulações, alinhamentos, etc. Para que estes recursos funcionem bem é necessário na maioria das vezes que não haja espaços em excesso, ou seja, que não haja espaço no início do pará-



# ARTIGO / WordStar to Windows (WS2WIN)

grafo e somente um espaço entre as palavras. Para isso existe a opção /U, a qual realiza esta função, senão deixa no texto o número de espaços original.

## Uso

A listagem 1 mostra o programa completo. O uso do programa é muito fácil.

### Exemplos:

C>ws2win origem destino.txt - converte a origem e mantém o número de espaços.

C>ws2win origem destino.txt /u - converte a origem e deixa apenas um espaço entre as palavras e nenhum espaço no início do parágrafo.

Se o arquivo destino já existir será necessário confirmar a sobrescrição ou abortar.

Para usar o texto convertido, o caminho mais fácil é importá-lo, ou se preferir inseri-lo, através do menu inserir arquivo. O Word perguntará qual o formato do arquivo, e fará a sugestão So-

mente texto, aceite. Para que o texto seja gravado como um autêntico .DOC é só salvá-lo. No Write, você abre o texto, em seguida ele perguntará se deve ou não converter, escolha não converter, depois é só gravar, quando então será gravado no formato do Write (WRI), e *c'est fini*.

Se você tem diversos arquivos a converter crie os batchs abaixo:

C>copy con ws2.bat

ws2win %1 c:\winword\cartas\%1.txt /u

CTRL + Z

ENTER

C>copy con ws2a.bat

ws2win a:%1 c:\winword\cartas\%1.txt /u

CTRL + Z

ENTER

C>copy con ws2b.bat

ws2win b:%1 c:\winword\cartas\%1.txt /u

CTRL + Z

ENTER

O primeiro serve para o caso de estar a origem no drive e diretório atuais. O

segundo quando está no drive A, e o terceiro no drive B. Assim é só informar o nome do arquivo origem (sem extensão, aliás, neste caso ele não deve ter extensão), que ele será convertido e colocado no diretório C:\WINWORD\CARTAS, que geralmente é o diretório padrão do Word. Todos estes batch deverão ser copiados para um diretório público para que possam ser vistos por todos os outros diretórios.

A Listagem 1 mostra o programa, totalmente comentado. É digitar, compilar e utilizar.

Formado em Análise de Sistemas pela UNIMEP de Piracicaba, **Carlos Roberto Macedônio** trabalha atualmente no Depto. de Genética, ESALQ/USP. Programa em C, BASIC, Pascal, Clipper e HP 11-C. Para eventuais contatos, escreva para: Carlos Roberto Macedônio, R Prof. Raul Duarte, 292 - Piracicaba, SP - CEP 13403-005 e-mail: crmacedo@carpa.ciagri.usp.br

## Listagem 1

```
*
|-----|
| Programa: WS2WIN.C |
| Objetivo: Converte arquivos feitos no WS para padrao |
| .TXT do Windows. |
| Linguagem.....: C (Turbo C 2.0) |
| Equipamento....: IBM (PC,XT,AT) |
| Autor.....: Carlos Roberto Macedonio |
| Data criacao...: 26/09/1993 Ult. Alteracao: 06/02/1997 |
|-----|
*
#include <stdio.h>
#include <string.h>
#include <dir.h>
#include <dos.h>
#define SPC 0x20 /* espaco */
#define CR 0x0d /* carry return */
#define BS 0x08 /* backspace */
#define QUEBRA 0x0d /* quebra logica de paragrafo para o WS */

int atch(char *str, char ch); /* acha um caractere dentro de uma string */
int existe_arq(char *arq); /* verifica se ha um arquivo (arq) */
void erro(int nerr); /* Mostra o erro ocorrido e aborta. */

main(argc,argv)
int argc;
char *argv[];
{
FILE *saida,*entrada;
int c,buffer=0,i,p;
int flag_geral=1,flag_spc=0,flag_char=0;
int tab[14][5] = { 193,192,194,195,196,201,200,202,0,203,
```

```
205,204,206,0,207,211,210,212,213,214,
218,217,219,0,220,225,224,226,227,228,
233,232,234,0,235,237,236,238,0,239,
243,242,244,245,246,250,249,251,0,252,
0,0,0,209,0,0,0,0,241,0,
221,0,0,0,0,253,0,0,0,255};

printf("WS2Win (Word Star to Windows)\n");
printf("Carlos R. Macedonio Set./1993\n");
if (argc < 3) {
printf("Uso: WS2WIN entrada[.ext] saida[.ext] /U\n");
printf("entrada[.ext]: fonte do texto a ser convertido: n");
printf("saida[.ext]: destino do texto convertido: n");
printf("/U: deixa apenas um espaco entre as palavras. n");
exit(1);
}
strupr(argv[1]);
strupr(argv[2]);
if (!existe_arq(argv[1])) /* procura o arquivo fonte */
printf("Arquivo %s nao encontrado!\n",argv[1]); /* nao achou */
exit(1); /* termina e sinaliza o DOS */
if (existe_arq(argv[2])) /* procura o arquivo destino */
do {
printf("Arquivo %s ja existe! Sobrecrevo-o. S/N? ",argv[2]);
c=getche();
printf("\n");
} while (atch("SsNn",c) == -1);
printf("\n");
if (c == 'N' || c == 'n') exit(1); /* termina e sinaliza o DOS */
if (!(!entrada=fopen(argv[1],"rb"))) /* tenta abrir arquivo fonte */
printf("Problemas com: %s\n",argv[1]);
exit(1);
```



```

if (! (saida=fopen(argv[2], "w"))) { /* tenta abrir arquivo destino */
    printf("Problemas com: %s\n", argv[2]);
    exit(1);
}

if (argc >= 4) { /* se ha opcao, processa */
   strupr(argv[3]);
    if (argv[3][0] == '/') {
        if (!strcmp(argv[3], "/U")) flag_geral=0; /* U - So Um Espaco */
    }
}

printf("\nos -> %s\n", argv[1], argv[2]);
while ((c=getc(entrada)) != EOF) { /* le ate o fim do arquivo */
    if (c == QUEBRA) { /* quebra de paragrafo */
        if ((c=getc(entrada)) == EOF) erro(1); /* pega o LF */
        if (buffer == SPC) { /* Se o caractere no buffer = espaco */
            if (flag_spc || flag_geral) { /* e espaco e permitido */
                if (fputc(buffer, saida) == EOF) erro(0); /* grava */
                flag_spc=0; /* Desabilita espaco */
            }
        }
        else { /* se maior que espaco, grava */
            if (buffer > SPC) {
                if (fputc(buffer, saida) == EOF) erro(0);
            }
            flag_spc=1; /* habilita gravacao de espaco */
        }
        buffer=0; /* zera o buffer */
        continue;
    }
    if (c == CR) {
        if ((c=getc(entrada)) == EOF) erro(1); /* le o LF */
        if (buffer > SPC) { /* ha algo no buffer que nao espaco */
            if (fputc(buffer, saida) == EOF) erro(0);
            flag_spc=1;
        }
        if (fprintf(saida, "\n") == EOF) erro(0); /* grava direto CR+LF */
        buffer=0; /* zera buffer */
        flag_char=0; /* Novo paragrafo, desabilita */
        continue; /* espaco no inicio deste */
    }
    c&=127; /* desliga o oitavo bit */
    if (c == BS) { /* indica acento */
        if ((c=getc(entrada)) == EOF) erro(1); /* le o acento */
        c&=127; /* desliga o oitavo bit */
        if (buffer == 'C' || buffer == 'c') { /* cedilha */
            buffer=(buffer == 'c' ? 0xc7 : 0xc7); /* ç ou Ç */
        }
        else {
            i=atrch("^~`_ao", c); /* encontra a coluna */
            p=atrch("AEIOUaeiouNnYy_", buffer); /* encontra a linha */
            if (i >= 6 && p == 14) /* primeiro c/caractere "_ " antes */
                buffer=(i == 6 ? 0xaa : 0xba); /* º ou º */
            else { /* primeiro(a) com caractere "_ " depois */

```

```

            if (i == 5 && (p == 5 || p == 8))
                buffer=(p == 5 ? 0xaa : 0xba); /* º ou º */
            else if ((buffer=tab[p][i]) == 0) buffer=SPC;
        }
    }
    /* o buffer tornou-se "acentuado" */
    if (fputc(buffer, saida) == EOF) erro(0);
    buffer=0; /* zera o buffer */
    flag_spc=1;
    flag_char=1;
    continue;
}
if (buffer) { /* ha algo no buffer, tenta grava-lo */
    if (buffer >= SPC) {
        if (buffer == SPC) { /* se espaco */
            if ((flag_spc && flag_char) || flag_geral) { /* espaco ok */
                if (fputc(buffer, saida) == EOF) erro(0); /* grava */
                flag_spc=0; /* desabilita permissao */
            }
        }
        else { /* nao e espaco */
            flag_spc=flag_char=1; /* permite espaco */
            if (fputc(buffer, saida) == EOF) erro(0);
        }
    }
    buffer=c; /* salva ultima leitura */
}
fcloseall();
printf("\n");
return(0); /* devolve Ok para o DOS */
}
/* Fornece a posicao na qual se encontra ch dentro de str, senao -1. */
int atrch(char *str, char ch)
{
    register int i=0;
    while (str[i]) {
        if (str[i] == ch) return(i);
        ++i;
    }
    return(-1);
}
/* Se acha devolve 1, senao 0. */
int existe_arq(char *arquivo)
{
    struct fblk dir;
    return((findfirst(arquivo, &dir, FA_ARCH) == -1) ? 0 : 1);
}
/* Mostra o erro ocorrido. */
void erro(int nerr)
{
    if (nerr) printf("Erro de escrita!!!\n");
    else printf("Erro de leitura!!!\n");
    exit(1);
}

```

**ANÚNCIE AQUI**  
**4,5 X 18 cm**  
**APENAS R\$ 100,00**  
 Informações pelo telefone(021)589-2088

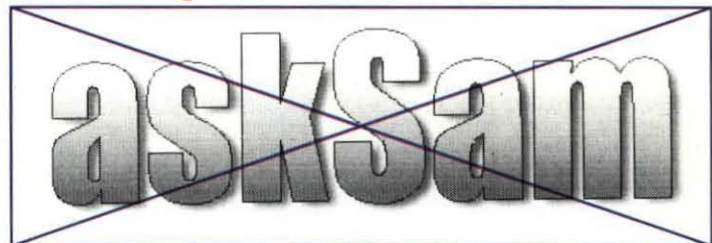
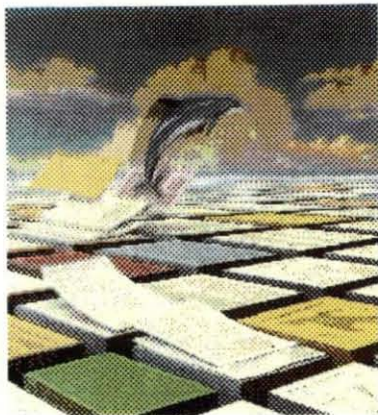


# ~~ONDE VOCÊ ARMAZENA SUAS HOME PAGES PREFERIDAS?~~

**S**e você costuma armazenar seus textos em um editor de textos e depois não consegue lembrar do nome dos arquivos...

**S**e você necessita trabalhar com informações estruturadas em campos e com informações desestruturadas ao mesmo tempo...

**Então você precisa do banco de dados  
textual e hipertexto**



*for Windows*

**Que tal um software que, além de buscar informações por qualquer palavra ou frase do texto, permite que você organize as Home Pages e as suas mensagens do correio eletrônico em bases de dados para posterior recuperação?**

**Nas versões Monousuário, Rede, Electronic Publisher  
e Web Publisher.**

**Consulte-nos:**



**Tel/Fax: (021)571-5372 E-mail: [srodas@prolink.com.br](mailto:srodas@prolink.com.br)  
<http://www.egeria.com.br>**



# ABRA SUA EMPRESA DE SOFTWARE COM 50.000 PROGRAMAS!



É isso aí! São mais de 50.000 programas entre aplicações gráficas, utilitários em geral, programas comerciais, editores de textos, programas matemáticos, educacionais, utilitários para internet, jogos, etc. Adquirir uma cópia deste nosso fantástico acervo de programas, gravados em 15

CD-ROMs, juntamente com o programa de cópia que facilita a transferência dos programas dos CDs para disquete (ou para seu winchester), um programa localizador que acha com facilidade o programa que você precisa em um banco de dados com a listagem completa dos programas incluindo nome, descrição, data, tamanho em bytes, categoria e código.

SIM, todos os 50.000 programas serão seus, para você usar, copiar, vender ou alugar a qualquer hora?

E quanto custa este pacote com mais de 50.000 programas e todos os outros itens descritos acima? Apenas R\$1.500,00 que também podem ser pagos em 3 parcelas de R\$600,00 ou seis parcelas de R\$350,00. Todo o material tem garantia de 5 anos e vem acompanhado de nota fiscal para sua segurança e tranquilidade.

NEMESIS INFORMÁTICA LTDA.

Rua Sete de Setembro, 92 sala 1203-centro-Rio de Janeiro-RJ-Brasil

TELEFONE: (021) 242-0348-fax (021) 242-4760

INTERNET: <http://www.ibase.org/~nemesistur>