

Spooler

MAGAZINE DOS UTILIZADORES PC

O Fantástico Mundo das BBS (III)



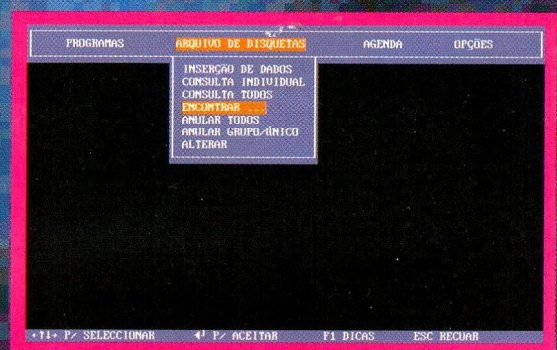
IMPRESSORAS (II)

- ◆ XUTILS
- ◆ HABILIDADES
- ◆ PROGRAMAÇÃO GRÁFICA PARA TURBO PASCAL
- ◆ SPOOLDOS
- ◆ ASTÚCIAS
- ◆ MOUSE «DEMO»
- ◆ CIFRA & BCIFRA
- ◆ FC MANAGER
- ◆ MEMÓRIA ESTENDIDA (II)
- ◆ TESTE C – A SOLUÇÃO
- ◆ CONFIG
- ◆ ETC.

PLANO Segunda-feira, 12 de Julho de 1993 17:46:13,36 207 Semanas N2/93

SEG, 93-07-05	TER, 93-07-06	QUA, 93-07-07	QUI, 93-07-08	SEX, 93-07-09
Esta semana não há mensagens de utilidade para o caso de espaço de memória.	1,00 Levantar e ir para o trabalho. 1,00 Manter com o conselho de administração.	10,00 11,00 12,00 13,00 14,00 15,00 16,00 17,00 18,00 19,00 20,00 21,00 22,00 23,00 24,00 25,00 26,00 27,00 28,00 29,00 30,00 31,00 32,00 33,00 34,00 35,00 36,00 37,00 38,00 39,00 40,00 41,00 42,00 43,00 44,00 45,00 46,00 47,00 48,00 49,00 50,00 51,00 52,00 53,00 54,00 55,00 56,00 57,00 58,00 59,00 60,00 61,00 62,00 63,00 64,00 65,00 66,00 67,00 68,00 69,00 70,00 71,00 72,00 73,00 74,00 75,00 76,00 77,00 78,00 79,00 80,00 81,00 82,00 83,00 84,00 85,00 86,00 87,00 88,00 89,00 90,00 91,00 92,00 93,00 94,00 95,00 96,00 97,00 98,00 99,00 100,00	PERÍODO **** Ir visitar o museu com a família. Ler a Spooler	19,00 Atender à conferência de imprensa na sede do partido. 12,00 Elaborar o relatório semanal.

Plano V. 1.10



Base Dados Indispensável



Show V. 1.0

Algumas boas razões para comprar o novo

CORELDRAW!



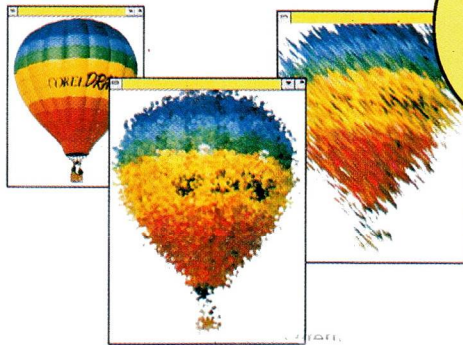
Gratificia PUB

- Mais de 750 fontes True Type e Type 1, incluindo 650 fontes Bitstream e ITC.
- Dois CDs com as fontes e mais de 18 000 imagens, bibliotecas de animação e sons.
- Melhoria acentuada na separação de cores e controlos para pré-impressão.
- Sistema TWAIN para leitura directa de scanner no Corel PHOTO-PAINT e no CorelTRACE.



- São incríveis os melhoramentos introduzidos no CorelDRAW. Desde a ilustração artística, à composição, ao desenho técnico, à paginação, o CorelDRAW 4 é uma ferramenta completa para a Edição Electrónica (DTP). Pode nesta nova versão, criar até 999 páginas e escrever parágrafos de texto com 4 000 caracteres, ou usar qualquer dos 43 templates de concepção profissional. Pode contornar objectos com blocos de texto (externa ou interiormente) e exercer um controlo completo sobre tabulações, indentações, etc.
- No domínio da ilustração poderá usar mais de um milhão de variantes sobre 40 texturas fractais. Poderá desenhar usando formas em gota, cónicas ou formas definidas por si, simulando traços esboçados manualmente.
- Mais de 5 000 símbolos disponíveis, novos filtros para importação/exportação de ficheiros, incluindo Micrografx DRW, JPEG e filtros para importação de 21 processadores de texto.

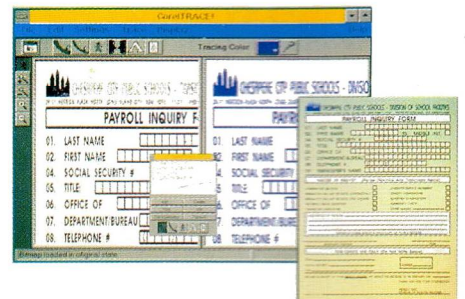
- CorelMOVE é um novo módulo de animação para dar largas à sua criatividade com animação. Com comandos de fácil utilização poderá criar cenas fascinantes, animadas e interactivas.
- Por seu turno o CorelCHART permite-lhe proceder a cálculos a partir do gestor de dados, utilizando mais de 80 fórmulas matemáticas e financeiras. É ainda possível aceder aos padrões e texturas do CorelDRAW.



CorelDRAW 4.0
O melhor programa
para grafismos
torna-se ainda
MELHOR!

ACEITAM-SE
REVENDEDORES

- O CorelSHOW vem proporcionar apresentações dinâmicas. Pode, por exemplo, correr uma animação num quadro e texto e adicionar efeitos de «fade in/out» a textos e/ou gráficos dentro ou fora desse quadro.
- O CorelMOSAIC módulo de grande utilidade, foi também melhorado com uma nova gestão de ficheiros «drag and drop» de fácil utilização.



- No CorelTRACE também muitas novidades. Sendo de destacar a inclusão de um módulo de reconhecimento óptico de caracteres (OCR), de modo a tornar editável o texto de qualquer documento. Novas potencialidades de tracing foram incluídas.

E estas são apenas algumas das novidades do CorelDRAW 4. As outras, só vendo!



Laserdata, Lda.

SEDE: Rua Nina Marques Pereira, 24 - Loja C - 1500 LISBOA - Telef.: (01) 716-1001/2/3/4 - Fax: (01) 716-3363
LASERDATA NORTE: Rua Agostinho da Silva Rocha, 989 - Nogueira - 4470 MAIA - Telef.: (02) 948-0013/948-0015 - Fax: (02) 948-0025



N.º 24 – Agosto/Setembro 1993

Spooler

MAGAZINE DOS UTILIZADORES PC

OS 24 NÚMEROS DA SPOOLER!

Se a **Spooler** já tivesse retomado a periodicidade original – publicação mensal (Olá! Aqui anda gato...) e tivesse conseguido manter essa periodicidade, muito provavelmente estaríamos a comemorar o seu quarto aniversário. Mas (ainda) não é mensal e a comemoração só pode verificar-se através da publicação do seu **vigésimo quarto número**. Digamos que comemora «24 numerozinhos» ao cabo de quatro anos de existência, o que vem confirmar a assumida bimestralidade.

Também consolidou a sua plena maturidade. Adquiriu o respeito geral por ter conseguido controlar uma crónica irregularidade na sua publicação. Hoje, os assinantes e leitores da **Spooler** podem contar com a sua revista, regularmente, em suas casas ou nas bancas. Não menos importante será o facto de termos mantido – e alargado – o carácter didático e formativo que sempre nos guiou.

Ultrapassou uma séria limitação e modernizou-se: a disquete de 3 1/2" permite satisfazer, quase na sua totalidade, a crescente tendência para a cada vez maior dimensão dos programas que queremos publicar. E também para diminuir a «fila de espera» que se está a verificar na rubrica «Colaboração do Leitor».

Em termos editoriais, a **Spooler** constitui um caso ímpar: é a única revista portuguesa de informática que esgota as suas edições, reedições e (re)reedições! Não será demais lembrar que já vamos na terceira edição dos números 1-2-3, na segunda dos números 4-5-6 e preparamo-nos para reeditar os números 7-8-9.

O que se conseguiu deve-se, em grande parte, ao apoio e estímulo sempre presentes por parte dos nossos fiéis assinantes e leitores. E também à amizade e fidelidade do nosso corpo de colaboradores. A todos os nossos agradecimentos. Um abraço da **Spooler** e do



Renato Casquilho

- 2 LIVROS
- 4 O CONTEÚDO DA DISQUETA
- 5 XUTILS
- 8 HABILIDADES (I)
- 10 PLANO v1.10
- 11 ANSITEST.COM E TESTANSI.COM
- 12 PROGRAMAÇÃO GRÁFICA PARA TURBO PASCAL
- 15 SPOOLDOS/LISTA
- 17 SHOW v1.0
- 18 ASTÚCIAS/TRANCA
- 19 SCADA (V)
- 20 IMPRESSORAS
- 23 MOUSE «DEMO»
- 24 O FANTÁSTICO MUNDO DAS BBS (II)
- 26 BDI – BASE DE DADOS INDISPENSÁVEL
- 28 CIFRA & BCIFRA
- 29 FC MANAGER
- 31 CÁLCULOS MATEMÁTICOS
- 32 CONECTIVIDADE
- 35 EDIC
- 36 MEMÓRIA ESTENDIDA (II)
- 39 TESTE C – A SOLUÇÃO
- 40 CONFIG
- 41 MONTRA SPOOLER



- Propriedade: Graficria, Publicidade e Artes Gráficas, Lda.
- Director de Edição: Franco Gomes ■ Director: Renato Casquilho
- Redacção: Franco Gomes, Paulo Bernardo e Renato Casquilho.
- Colaboradores: Carlos Ladeira, Cunha Brandão, Eduardo Cardoso, José Orlando Pereira, José Páscoa, Paulo Bernardo, Vitor Monteiro.
- Ilustrações: Leonor Gomes
- Relações Públicas e Publicidade: Fátima Ribeiro
- Assinaturas e encomendas: Guiomar Russo
- Redacção, Publicidade e Serviços Administrativos: Rua Duarte Pacheco Pereira, 69-A – Damaia – 2700 AMADORA – Telfs.: 475 22 89/474 60 11/41
- Produção Gráfica: Graficria, Publicidade e Artes Gráficas, Lda.
- Impressão: Empresa Litográfica do Sul, S.A. ■ Distribuição: Electroliber
- Revista bimestral ■ Preço: 600\$00 ■ SRIP N.º 213900
- Tiragem: 12500 exemplares ■ Dep. Legal N.º: 28587/89

Livros

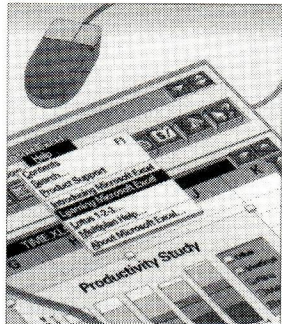
A DUAS CORES

Excel for Windows

Depressa e Bem

VERSÃO 4

Para as versões inglesa e portuguesa
 Opções para Inicializar
 Cria a sua primeira folha de cálculo em minutos
 As ilustrações e dicas cores facilitam a aprendizagem mais rápida
 Apenas as técnicas básicas - uma iniciação excelente!



EXCEL FOR WINDOWS – DEPRESSA E BEM – VERSÃO 4

Uma folha de cálculo nunca foi tão fácil de aprender como com este guia visual para o Excel for Windows, Versão 4. Lições passo-a-passo para o processo de criação e trabalho com folhas de cálculo. Ilustrações mostrando exactamente o que é visto no ecrã. Abordagem dos conceitos fundamentais para as tarefas diárias e essenciais envolvendo folhas de cálculo:

- Criar uma folha de cálculo com dados, títulos, cabeçalhos e legendas
- Somar uma coluna de números
- Criar e imprimir relatórios
- Produzir quadros e gráficos coloridos

Título: Excel for Windows Versão 4 – Depressa e Bem
Autor: Gerald E. Jones
Idioma: Tradução portuguesa
Editadora: FCA – Editora de Informática
Distribuidora: LIDEL – Edições Técnicas, Lda.
Preço: 3.240\$00 (IVA incluído)
ISBN: 9-722-00013-6



Estes e outros livros sobre informática editados pela Sybex, QUE, Ziff, Editorial O Livro, FCA, Edições Cetop, McGRAW-HILL e Graficria, podem ser encomendados através da Montra Spooler. Para o efeito utilize o cupão respectivo. Os assinantes da Spooler beneficiam de 10% de desconto. Os preços aqui indicados poderão sofrer alterações por parte dos editores/distribuidores.

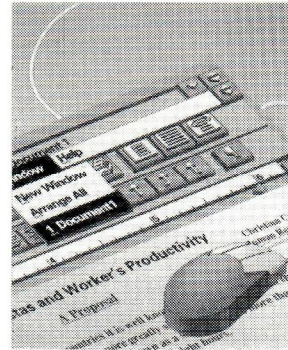
A DUAS CORES

Word for Windows

Depressa e Bem

VERSÃO 2.0

Para as versões inglesa e portuguesa
 Opções para Inicializar
 Imprima o seu primeiro documento em minutos
 As ilustrações e dicas cores tornam a aprendizagem mais rápida



WORD FOR WINDOWS – DEPRESSA E BEM – VERSÃO 2.0

O processamento de texto nunca foi tão fácil de aprender, como com este guia visual para o Word for Windows, Versão 2.0. Lições práticas com instruções pormenorizadas. Ilustrações mostrando, passo-a-passo, o que verá no ecrã. Em pouco tempo o leitor dominará todas técnicas, nomeadamente:

- Escrever, editar e gravar documentos
- Imprimir
- Criar cabeçalhos, rodapés, números de página e tabulações
- Alterar tipos e tamanhos de letra
- Corrigir erros e rearranjar parágrafos

Título: Word for Windows Versão 2.0 – Depressa e Bem
Autor: Christian Crumlish
Idioma: Tradução portuguesa
Editadora: FCA – Editora de Informática
Distribuidora: LIDEL – Editora de Informática, Lda.
Preço: 2.580\$00 (IVA incluído)
ISBN: 9-722-00012-8

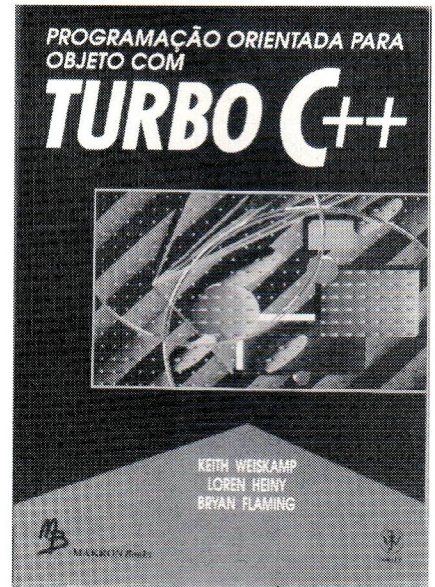


HOW COMPUTERS WORK

Faça uma excitante viagem visual ao coração do computador pessoal. Veja como:

- o PC «acorda» e se auto-analisa quando ligado
- o teclado, o rato e outros dispositivos de entrada enviam informação para o computador
- os dados são enviados para o monitor, o modem ou a impressora
- funciona a memória do computador e como este pode armazenar dados.

Título: How Computers Work
Autor: Ron White
Idioma: Inglês
Editadora: ZD-Press
Distribuidora: LIDEL – Edições Técnicas, Lda.
Preço: 5.381\$00 + IVA
ISBN: 85-216-0924-8

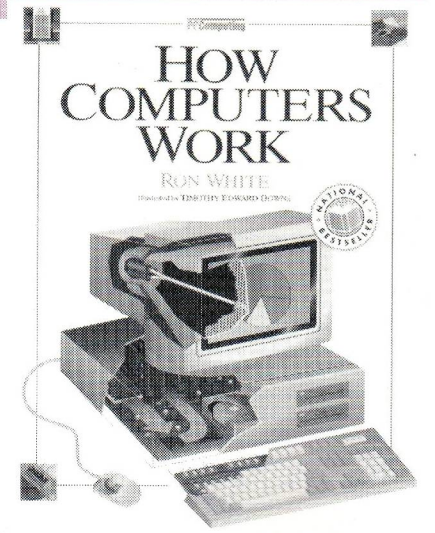


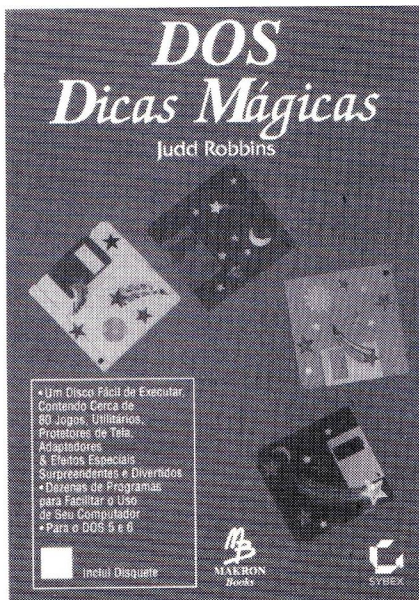
PROGRAMAÇÃO ORIENTADA PARA OBJECTO PARA OBJECTO COM TURBO C++

Este livro desvendará o mistério da programação orientada para objecto (POO) e mostrará que esta abordagem é uma forma natural de projectar e escrever código. Eis alguns dos aspectos apresentados no livro:

- Técnicas básicas da POO
- Abordagens práticas para a montagem de ferramentas orientadas para objectos
- Técnicas para criação de interfaces com o utilizador
- Análise de códigos para introdução de melhoramentos
- Ferramentas e aplicações para modo texto e gráfico

Título: Programação orientada para Objecto com TURBO C++
Autores: K. Weiskamp, L. Heiny e B. Flaming
Idioma: Tradução brasileira
Editadora: Makron Books do Brasil
Distribuidora: McGRAW-HILL de Portugal, Lda
Preço: 7.400\$00 + IVA
ISBN: 3460041-4





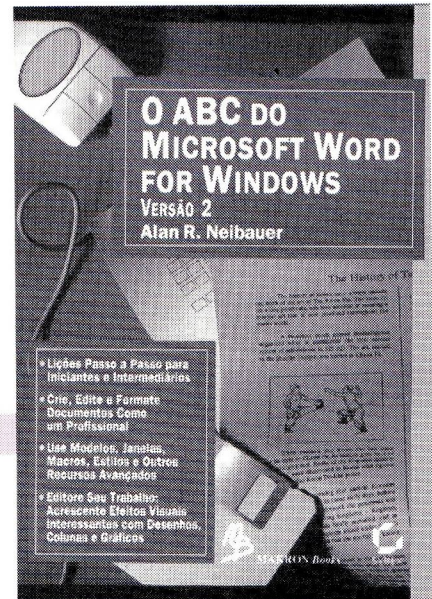
DOS – DICAS MÁGICAS

Obtenha o melhor espectáculo do DOS a partir de uma única fonte: mais de 80 programas fascinantes na disquete incluída no livro! Seleccionámos cerca de 80 programas divertidos e hipnotizantes do DOS, a partir de dezenas de fontes de informação do *Shareware* e *Freeware*. A disquete de alta densidade descomprime mais de 2 MB de *software* com um *batch file* para instalação automática.

O ABC DO MICROSOFT WORD FOR WINDOWS – VERSÃO 2

- Lições passo-a-passo para iniciados e intermediários
- Criar, editar e formatar documentos
- Utilização de modelos, janelas, macros, estilos e outros recursos avançados
- Edição dos trabalhos: efeitos visuais com desenhos, colunas e gráficos.

Título: O ABC do Microsoft Word for Windows Versão 2
Autor: Alan R. Neibauer
Idioma: Tradução brasileira
Editadora: Makron Books do Brasil
Distribuidora: McGRAW-HILL de Portugal, Lda
Preço: 6.200\$00 + IVA
ISBN: 3460068-6



Título: DOS – Dicas Mágicas
Autor: Judd Robbins
Idioma: Tradução brasileira
Editadora: Makron Books do Brasil
Distribuidora: McGRAW-HILL de Portugal, Lda
Preço: 5.000\$00 + IVA
ISBN: 3640055-4

LIVROS JÁ DIVULGADOS NA SPOOLER

Título: Mastering DOS 5
Idioma: Inglês
Autor: Judd Robbins
Editadora: Sybex
Preço: 6 810\$00
Distribuidora: Lidel – Edições Técnicas, Lda.
I.S.B.N.: 0-89588-767-3

Título: Mastering Quattro Pro 3
Idioma: Inglês
Autor: Gene Weisskopf
Editadora: Sybex
Preço: 6 810\$00
Distribuidora: Lidel – Edições Técnicas, Lda.
I.S.B.N.: 0-89588-841-6

Título: dBase III Plus – Programação Avançada (inclui disquete)
Idioma: Português
Autor: Renato Prista Casquilho
Editadora: Graficria, Publicidade e Artes Gráficas, Lda.
Preço: 1 800\$00
Distribuidora: Lidel – Distri Cultural D.L. Nº 48864/91

Título: Teach Yourself AutoCad
Idioma: Inglês
Autor: Genevieve Katz
Editadora: Sybex
Preço: 5 610\$00
Distribuidora: Lidel – Edições Técnicas, Lda.
I.S.B.N.: 0-89588-627-8

Título: DOS 5 – Techniques and Utilities
Idioma: Inglês
Autor: Jeff Proside
Editadora: Ziff – David Press
Preço: 9 950\$00
Distribuidora: Lidel – Edições Técnicas, Lda.
I.S.B.N.: 1-56276-007-6

Título: The ABC'S of Toolbook for Windows
Idioma: Inglês
Autor: Kenyon Brown
Editadora: Sybex
Preço: 5 370\$00
Distribuidora: Lidel – Edições Técnicas, Lda.
I.S.B.N.: 0-89588-795-9

Título: Pc Computing Guide to Excel 3.0
Idioma: Inglês
Autor: Dale Lewaller
Editadora: Ziff – Davis Press
Preço: 6 450\$00

Distribuidora: Lidel – Edições Técnicas, Lda.
I.S.B.N.: 1-56276-019-X

Título: Fundamental do DOS 5.0
Idioma: Português
Autor: Henrique Carreiro
Editadora: FCA
Distribuidora: Lidel – Edições Técnicas, Lda.
Preço: 1 350\$00
I.S.B.N.: 9-722-00000-4

Títulos:
 Volume 1
 - CLIPPER 5.0 – Linguagens, Comandos.
Preço: 7 195\$00
I.S.B.N.: 85-216-0847-0
 Volume 2
 - CLIPPER 5.0 – Funções, Compilação, Ligação, Execução.
Preço: 6 065\$00
I.S.B.N.: 85-216-0848-9
 Volume 3
 - CLIPPER 5.0 – Rede local, Utilitários, Aplicações, Erros.
Preço: 5 800\$00
I.S.B.N.: 85-216-0849-7
 Volume 4
 - CLIPPER 5.0 – Programação, Arquitectura de sistemas.
Autor: António Geraldo da Rocha Vidal
Idioma: Português
Editadora: Livros Técnicos e Científicos Editora, Lda.
Preço: 4 560\$00 S/IVA
Distribuidora: Lidel – Edições Técnicas, Lda.
I.S.B.N.: 85-216-0924-8

Título: Introdução à Programação em Pascal
Idioma: Português
Autores: William Findlay e David A. Watt
Tradução: Ana Cristina Calação Dias
Edição e Distribuição: Edições Cetop – Coleção Infotek
Preço: 3 200\$00
I.S.B.N.: 972-641-048-7

Título: Folhas de Cálculo – Manual de Estilo
Idioma: Português
Autores: David Harrison e John W. Yu
Tradução: Jorge Manuel da Silva Neves
Edição e Distribuição: Edições Cetop – Coleção Infotek
Preço: 2 090\$00
I.S.B.N.: 972-641-156-4

Título: Understanding 1-2-3 for Windows
Idioma: Inglês
Autor: Douglas Herget
Edição: Sybex
Distribuição: Lidel – Edições Técnicas, Lda
Preço: 6 250\$00 S/IVA
I.S.B.N.: 0-89588-845-9

Título: Fundamental do Quatro Pro 3
Idioma: Português
Autor: Renato Reis
Editadora: FCA – Editora de Informática, Lda.
Distribuidora: Lidel – Edições Técnicas, Lda
Preço: 1 350\$00
I.S.B.N.: 972-722-001-0

Título: Fundamental do Windows 3.0 e 3.1
Idioma: Português
Autores: A. Barrote e R. Lemos
Edição: FCA – Editora de Informática
Distribuição: Lidel – Edições Técnicas, Lda
Preço: 1 420\$00
I.S.B.N.: 972-722-002-9

Título: Linguagem Assembler Orientada para Objeto
Idioma: Português
Autor: Len Dorfman
Tradução: Mário Moro Feccchio
Edição e Distribuição: Editora McGRAW-HILL de Portugal, Lda.
Preço: 5 300\$00 S/IVA
I.S.B.N.: 0-07-460614-X

Título: Microsoft C – Dicas, Segredos e Truques
Idioma: Português
Autor: Kris Jamsa
Edição e Distribuição: Editora McGRAW-HILL de Portugal, Lda.
Preço: 6 600\$00 S/IVA
I.S.B.N.: 0-07-460801-0

Título: Microsoft C/C++ 7 – The Complete Reference
Idioma: Inglês
Autores: William H. Murray, III & Chris H. Pappas
Edição e Distribuição: Editora McGRAW-HILL de Portugal, Lda.
Preço: 8 300\$00 S/IVA
I.S.B.N.: 0-07-881664-5

Título: WordStar 6.0 – Guia do Usuário
Idioma: Português
Autor: Walter A. Ettl
Edição e Distribuição: Editora McGRAW-HILL de Portugal, Lda.
Preço: 3 465\$00 (IVA incluído)
I.S.B.N.: 460636-0

Título: Turbo Pascal for Windows – Techniques and Utilities
Idioma: Inglês
Autor: Neil J. Rubenking
Editadora: ZD Press
Distribuidora: Lidel – Editora de Informática, Lda
Preço: 10 450\$00 (IVA incluído)
I.S.B.N.: 1-56276-035-1

GUIAS DO UTILIZADOR
Títulos: Windows 3, Lotus 1.2.3., dBase III Plus
Idioma: Português
Edição e Distribuição: Edições CETOP
Preço: 950\$00 +IVA
I.S.B.N.: 0-07-460801-0

Título: Fundamental do Quattro Pro 3.0 e 4.0
Idioma: Português
Autor: Renato Reis
Editadora: FCA – Editora de Informática, Lda.
Distribuidora: Lidel – Edições Técnicas, Lda.
Preço: 1 470\$00
I.S.B.N.: 972-722-004-5

Título: Introdução à Programação em Cobol
Idioma: Português
Autor: Ray Welland
Edição e Distribuição: Edições CETOP
Preço: 3 380\$00 S/IVA
I.S.B.N.: 972-641-042-8

GUIAS DO UTILIZADOR
Títulos: «Como usar o seu computador com DOS»; «WORDPERFECT 4.2, 5.0 E 5.1»; «Windows 3.1 – Programa em Inglês»; «Windows 3.1 – Programa em Português»; «DOS 5.0»; «HARVARDGRA-PHICS».
Idioma: Português
Edição e Distribuição: Edições CETOP
Preço: 1 050\$00 + IVA (por título)

Título: CorelDRAW 3 – Guia do Usuário
Idioma: Português
Autor: Steve Rimmer
Editadora: MAKRON Books
Distribuidora: McGRAW-HILL de Portugal, Lda.
Preço: 6 195\$00 (IVA incluído)
I.S.B.N.: 460704-9

Título: NetWorking Windows for WorkGroups
Idioma: Inglês
Autor: Martin Matthews & Bruce Dobson
Editadora: Osborne
Distribuidora: Editora McGRAW-HILL de Portugal, Lda.
Preço: 5 565\$00 (IVA incluído)
I.S.B.N.: 881937-7

Título: Guia Prático Windows 3.1
Idioma: Português
Autor: Marshall L. Moseley
Editadora: FCA – Editora de Informática, Lda.
Distribuidora: Lidel – Edições Técnicas, Lda.
Preço: 2 829\$00 +IVA
I.S.B.N.: 972-722-006-1

(Nota: Os preços indicados poderão ter sofrido alterações)



O DA CONTEÚDO DA DISQUETA

ANSITEST.EXE

«São dois utilitários cuja missão é fundamentalmente a mesma: detectar se temos instalado em memória o conhecido *device driver* ANSI.SYS, ou qualquer outro cuja missão seja equivalente.» É assim que José Páscoa inicia o seu artigo. Não deixe de o ler antes de ensaiar os utilitários.

ASTUCIAS.EXE

Bruno Muscão oferece aos programadores de Pascal uma variante ao procedimento EXEC. Eis as suas palavras de introdução: «Esta rotina que envio é uma variante do procedimento Exec do Turbo Pascal. A rotina será especialmente útil a todos aqueles que nos seus programas façam um *shell* ao DOS ou a outro subprocesso e fiquem sem memória suficiente e o processo acabe por abortar, ou por correr com pouca memória.»

BDIEXE.EXE

Este programa já «deambulava» há alguns meses na nossa redacção, sempre objecto de um profundo diálogo com o autor. A versão agora oferecida aos nossos leitores é o resultado desse diálogo e dos meritórios esforços do José Tomás no sentido do aperfeiçoamento. Considerámo-lo de indiscutível interesse, sobretudo pelo carácter didático que a sua construção encerra. Convidamos os entusiastas do QBasic a uma cuidada análise das fontes incluídas na disquete.

CIFBCIF.EXE

Aqui temos mais um programa dedicado à segurança dos nossos dados, um tema de grande actualidade que está a levantar alguma polémica em certos meios. Mas enquanto os outros discutem «o sexo dos anjos», leiam o que o Carlos Batista tem a dizer sobre o assunto.

CONFIG.EXE

O nosso amigo Hugo Meneses oferece-nos um interessante conjunto de utilitários destinados a per-

mitir uma configuração do sistema adaptada a casos particulares. Os exemplos que apresenta são típicos das dificuldades de um 286 em lidar com aplicações exigentes. Mas o «engenho e a arte» conseguem colmatar muitas dessas limitações — foi o que o Hugo fez, e bem. Mas contenham-se: não mexam em nada sem uma leitura atenta do artigo e um estudo não menos apurado dos ficheiros de comandos que seguem na disquete.

EDICEXE.EXE

Estamos em «maré» de editores de caracteres. E ainda bem, pois começa a ser cansativo estarmos sempre com o mesmo «visual» no computador... O nosso amigo Rui Anastácio contribui com mais uma versão destes simpáticos utilitários, desta vez construída em Pascal.

FCMANAG.EXE

Pedro Sinogas não quer que os leitores da Spooler se vejam confrontados com as mesmas dificuldades por que passou, ao ter adoptado a plataforma Windows. Eis um excelente utilitário com a não menos excelente apresentação facultada pelo ambiente Windows. Conforme é dito pelo autor, o programa necessita da presença, no directório Windows, do ficheiro VBRUN100.DLL. Tal ficheiro, pela sua dimensão, não é incluído na disquete deste número. Os nossos assinantes e leitores fiéis já o receberam na disquete da Spooler N.º 18 com o programa WINMENU. Os leitores que não possuem este número da revista poderão passar pelas nossas instalações e obtê-lo, ou escrever-nos enviando uma disquete de 3 1/2": recebê-la-hão de volta com o dito ficheiro.

FREE-S24.TXT

Faça TYPE FREE-S24.TXT | MORE ou visualize este ficheiro em qualquer editor de texto. Tem à sua disposição mais 11 referências *Shareware* da «Montra Spooler».

GRAFPT.EXE

Trata-se do material de demons-

tração referido no artigo de Pedro Andrade.

LISTEXE.EXE

O António Manuel continua a presentear os programadores com pequenos e «úteis» utilitários. O LISTA é mais um para a «lista».

MOUSEMO.EXE

Aqui temos um bom aproveitamento da teoria exposta na série «Roedores Informáticos», publicada nos números 15, 16 e 17 da Spooler. E ainda um bem elaborado conjunto de ensinamentos para quem pretender melhorar os seus programas com as funções do dito «roedor». No «compactado» da disquete também está incluído um ficheiro de texto (TMOUSE.DOC) a não perder.

MQEXE.EXE

Paulo Caridade entrou pelas «matemáticas» e oferece-nos um «pacote» destinado a facilitar cálculos complicados. Eis o seu discurso: «Este programa permite aproximar uma lista de *m* pontos, por um polinómio de grau *n*, pelo método dos mínimos quadrados, ou seja...» O resto está no artigo...

PLANEXE.EXE

Parece uma Agenda, mas não é. Mesmo que fosse, desde que tivesse algo que fugisse à banalidade e introduzisse elementos de programação com interesse para os leitores, não deixaríamos de publicá-la. Mas este PLANO possui algo mais, satisfazendo todos os requisitos ao agrado da Spooler. O Nuno Zoio esclarece-nos no seu artigo.

PROTEC.EXE

Trata-se do programa demonstrativo referido no segundo artigo sobre «Memória Estendida», de José Páscoa, a não perder.

SHOW.EXE

Mais uma vez o BASIC demons-

tra as suas capacidades. O Ricardo Pereira construiu um interessante programa para emular um «placard electrónico» nos nossos computadores. Leia o artigo atentamente, pois são inúmeras as possibilidades de controlo do texto «a correr».

SPOOLDOS.EXE

O Ricardo e o Luciano continuam a apostar no Basic, mostrando-nos que esta linguagem é capaz de fazer o mesmo que as ditas de «elite». Experimentem este SPOOLDOS e editem as fontes para ver como se faz.

TESTEC.EXE

«Antes de mais, queria agradecer a todos aqueles que colaboraram com a Spooler participando no teste de C. No entanto, é de lamentar o escasso número de participantes neste teste/concurso, pois apenas 10 leitores participaram.» Quanto ao teste, este era constituído por 6 perguntas que Carlos Ladeira analisa no artigo.

TRANCEXE.EXE

O António Monteiro Pereira fez bem em insistir na programação do seu TRANCA. Até é interessante estudarmos as diferentes abordagens a um mesmo problema, por parte de programadores distintos.

XUTILS.EXE

«Um dos mais numerosos grupos de utilitários que se podem encontrar num sistema Unix é o conjunto de ferramentas de processamento de texto. Seguindo a ideia original dos criadores do Unix, a maioria destes utilitários são relativamente pequenos, mas concebidos de modo a serem usados em conjunto, o que proporciona uma grande flexibilidade na construção de ferramentas pelo utilizador final.» São estas as palavras introdutórias do interessante artigo de José Orlando Pereira, a não perder.

XUTILS



CUT, PASTE, HEAD & TAIL

Um dos mais numerosos grupos de utilitários que se podem encontrar num sistema Unix, é o conjunto de ferramentas de processamento de texto. Seguindo a ideia original dos criadores do Unix, a maioria destes utilitários são relativamente pequenos, mas concebidos de modo a serem usados em conjunto, o que proporciona uma grande flexibilidade na construção de ferramentas pelo utilizador final. Para o efeito, são geralmente combinadas em *pipes* (ver PIPE, N.º 7 da *Spooler*) e *shell scripts* que, de um modo simplista, se podem descrever como ficheiros *batch* (.BAT).

O que este mês lhes proponho é uma visita guiada a alguns destes utilitários. Para o efeito, reescrevi quatro deles em C com o propósito de serem compilados em DOS, para que todos os possam experimentar. Não são de modo nenhum os mais importantes, mas são de qualquer maneira, bastante úteis.

CUT

Como o nome indica, este utilitário serve para cortar. Mais concretamente, serve para separar um ficheiro de texto em colunas, operação bastante morosa se levada a cabo com um vulgar editor de texto. Podem especificar-se os limites das colunas de dois modos distintos, usando para tal os selectores **-c** ou **-f**:

```
cut -c<lista> [<fich1> <fich2> ...]
cut -f<lista> [-d<car>] [-s] [<fich1> <fich2> ...]
```

No primeiro caso, a lista que se segue a **-c** indica posições de caracteres, relativos ao início de cada linha. Por exemplo,

```
cut -c10-20 tabela.dat
```

servirá para seleccionar as colunas de caracteres 10 a 20 do ficheiro *tabela.dat*. Pode omitir-se um dos extremos:

```
cut -c-20 tabela.dat
cut -c10- tabela.dat
```

casos em que será tomado respectivamente o início e o fim da linha. Podem especificar-se colunas de um só carácter, e várias ao mesmo tempo, como nos exemplos seguintes:

```
cut -c5 ficheiro.dat
cut -c1,3,5 ficheiro.dat
```

Se não for especificado nenhum ficheiro, ou se for usado um **-** como nome, a entrada é tomada da consola, o que permite o uso deste comando em *pipes*:

```
cut -c10,-5-6
cut -c-12,15,-
```

Podem ser especificados nomes de vários ficheiros, que serão cortados sucessivamente segundo as mesmas instruções:

```
cut -c10-20 fich1.dat fich2.dat - fich3.dat
```

No segundo caso, quando se usa **-f**, os valores dados na lista, que tem o mesmo formato da usada com **-c**, referem-se a colunas de dimensão variável separadas por uma tabulação. Se desejável pode-se escolher outro carácter separador, usando **-d**. Podem usar-se caracteres especiais como separadores, usando para tal a notação usada na linguagem C, tais como **\t**, **\xnn** (hexadecimal), **\onn** (octal) e **\w**.

A acompanhar o selector **-f** pode usar-se ainda **-s**, que suprime as linhas onde não seja encontrado nenhum carácter separador. Por defeito estas linhas são escritas inalteradas.

A este comando, tal como a todos os que apresento de seguida, acrescentei o selector **-V**, que mostra uma breve nota que inclui a versão e a sintaxe resumida.

PASTE

Depois de termos as colunas separadas é natural que queiramos fazer com elas um novo arranjo, portanto, nada melhor que um utilitário que faça isso mesmo. Com o **paste** podemos colar lado a lado dois ou mais ficheiros, usando os separadores que quisermos, especificados mais uma vez com **-d**. A sintaxe de utilização é bastante simples, e reduz-se normalmente à enumeração dos diversos ficheiros a concatenar «lateralmente»:

```
paste [-d<lista>] <fich1> <fich2> ...
paste -s [-d<lista>] <fich1> <fich2> ...
```

A segunda forma, distinguida por **-s**, funciona de um modo distinto, pois processa cada um dos ficheiros independentemente dos outros, concatenando linhas. Por exemplo, na concatenação série, daí o **-s**, de um ficheiro com o conteúdo:

```
linha1
linha2
...
```

tem-se como resultado:

```
linha1 linha2 ...
```

Falta referir que no caso do **PASTE**, o selector **-d** pode receber mais do que um carácter de cada vez, que usará alternadamente da esquerda para a direita. O comando:

```
paste -s -d,;
```

com um ficheiro semelhante ao anterior dará como resultado:

```
linha1,linha2;linha3,linha4;...
```

Mais uma vez, um **-** como nome de ficheiro será tratado como a consola, e como tal poderá ser aproveitado para a construção de *pipes*.

HEAD

Este utilitário é muito simples, servindo para retirar as primeiras linhas de um ficheiro. Resolvi incluí-lo sobretudo para completar o grupo. Podem assim experimentá-los

a trabalhar em conjunto, pois é assim que estes utilitários demonstram verdadeiramente o seu poder. A sua sintaxe de invocação é muito simples, sendo a única opção possível, a especificação do número de linhas a recortar:

```
head [-<num>] [<fich> ...]
```

Por exemplo, para capturar uma breve descrição do conteúdo do ficheiro `head.c` num ficheiro `info`, basta escrever:

```
head -5 tail.c > info
```

Se não for especificado um número de linhas a recortar, será usado o valor 10.

TAIL

Mais uma vez temos o verso da medalha, e se para o **CUT** era **PASTE**, para o **HEAD** só podia ser mesmo **TAIL**, que como é esperado serve para retirar as últimas linhas de um ficheiro. Neste caso tem-se a oportunidade de especificar o número de linhas a partir quer do princípio, quer do fim do ficheiro, usando respectivamente as seguintes linhas de comando:

```
tail [+<num>lbc] [<fich> ...]
tail [-<num>lbc] [<fich> ...]
```

Podem ainda acrescentar-se as letras **l**, **b** ou **c**, que indicam respectivamente linhas, blocos ou caracteres como medida. Quando se estão a contar linhas, pode ainda acrescentar-se a letra **r**, que servirá para inverter a ordem pela qual as linhas serão escritas.

```
tail +30
```

Usando o selector **-r** sem especificar um número de linhas, será invertido todo o ficheiro, dispensando no Unix a existência de um comando dedicado a esse fim. Note-se ainda que um bloco é o equivalente a 512 caracteres, pelo que:

```
tail -512c
tail -1b
```

são comandos equivalentes.

E AINDA...

Entre os utilitários de tratamento de texto mais utilizados, incluem-se outros que infelizmente me é impossível transportar para DOS, sob pena de ficar sem tempos livres durante uns anos! Descrevo alguns para lhes dar uma ideia do conjunto, com que espero aguçar a vossa curiosidade.

Tal como a dupla **CUT/PASTE** serve para cortar e colar verticalmente um ficheiro, existem o **GREP** e o **CAT**, que são os equivalentes para operações na «horizontal». O **GREP**, contracção de *Globally search for the Regular Expression and Print*, permite seleccionar linhas baseando-se na procura de padrões, as tais expressões regulares que lhe dão o nome, e que funcionam de um modo semelhante aos meta-caracteres do DOS. O **CAT** serve para concatenar ficheiros, e pode em DOS se simulado com:

```
copy fich1+ fich2 +... con
```

Com um ficheiro apenas, é o equivalente ao **TYPE**, sendo essa a sua utilização mais vulgar.

Mais poderoso é o comando **SORT** que tem uma funcionalidade semelhante ao seu homónimo do DOS, que serve

para ordenar linhas de um ficheiro de texto, mas que permite especificar o tipo de ordenação de um modo bastante mais flexível mas infelizmente complexo.

Falta referir dois dos «canivetes suíços» preferidos dos utilizadores **Unix**: o **SED** e **AWK**. O primeiro, *stream editor* de seu nome, é usado normalmente para efectuar buscas e substituições, embora as suas capacidades vão muito para além disso. É capaz de manipular o texto de formas bastante subtis guiado por listas de comandos, obtendo-se com ele efeitos difíceis de conseguir com outras ferramentas. Tem também a propriedade de poder ser usado em ficheiros de tamanho ilimitado, pois em cada momento tem em memória apenas uma linha do ficheiro em questão. O **AWK**, que ao que parece deve o estranho nome às iniciais dos seus três criadores: Alfred Aho, Peter Weinberg e Brian Kernighan, é nada mais nada menos, que uma linguagem de programação dedicada ao processamento de texto, da qual existem intérpretes e mesmo compiladores destinados às mais diversas plataformas, incluído como é claro, DOS.

CLONES

O desenvolvimento destes quatro utilitários representou para mim um desafio interessante, pois fez-me enfrentar um problema que nunca tinha encontrado antes, e que se resume a uma palavra: compatibilidade. A diferença entre um programa concebido de raiz e um clone de um existente, embora possa não ser evidente, é significativa. Se me propusesse escrever quatro programas equivalentes, mas não necessariamente compatíveis, tomaria certamente opções que me facilitariam a programação, nomeadamente a nível de *interface* com o utilizador.

Mais difícil do que emular o comportamento documentado, é lidar com o que está escrito nas «entre-linhas» do algoritmo original, do cumprimento do qual depende uma boa réplica. Alguns dos problemas são relativamente óbvios e aparecem logo na fase inicial de planeamento, onde dúvidas são resolvidas com experiências com os originais. Outros são mais subtis, e só aparecem como resultado de testes exaustivos. Outros ainda, pura e simplesmente não aparecem, excepto depois de o programa estar publicado, é claro, ou não valessem também aqui as leis de Murphy!

Numa fase final do desenvolvimento, a única maneira de fazer os problemas vir à superfície é o confronto directo com o original. Neste caso, significou o transporte para **Unix** das réplicas, processo relativamente indolor, dada a existência de compiladores **ANSI C** em ambos os ambientes. Além de demonstrar algumas incompatibilidades, a gestão de memória do **Unix** ajudou a capturar um par de erros difíceis de ver em **DOS**, e que certamente provocariam embaraços no futuro.

POR DENTRO

Entre os quatro programas, (quase) todos me causaram algumas dores de cabeça. A excepção foi como já devem ter percebido, o **HEAD**, cujo código essencial se resume a duas (!) linhas, sendo o restante «embalagem».

No **CUT**, os problemas vieram de dois lados. Como se pode verificar facilmente, grande parte do código serve para decifrar o formato das listas que com **-c** e **-f** servem para especificar as colunas a recortar. O segundo problema teve a ver com a afinação da compatibilidade, que segundo acredito, está ainda longe de ser perfeita.

Claro que como complementar do **CUT**, o **PASTE** também deu a sua cota parte de problemas de compatibilidade, nomeadamente no tratamento de ficheiros de dimensões diferentes. Também deu algum trabalho suportar caracteres especiais na lista de delimitadores.

Foi também com este último que aproveitei os testes em **Unix** para detectar um erro bastante aborrecido, mas que surpreendentemente ainda não se tinha manifestado em DOS. Apesar de ter reservado quatro kilobytes de memória para guardar cada linha antes de ser escrita, bastante mais que os cerca de 500 do original, não foram suficientes para suportar alguns dos maquiavélicos testes a que o submeti, tal como fazer a colagem série das linhas do próprio ficheiro de código fonte. Obviamente, a memória reservada extravasava e corrompia a *stack*, o que em DOS poderia resultar num estouro do sistema.

O **TAIL**, para meu espanto, acabou por crescer e resultar no maior dos ficheiros de código C. A este facto não foi alheia a possibilidade de combinações entre as letras **lcb** com os sinais **+** e **-**, juntamente com a possibilidade de inverter a ordem dos resultados.

Até aqui falei apenas na necessidade de atingir a compatibilidade. É porém impossível resistir à tentação de introduzir modificações, sobretudo em situações onde as limitações do original são manifestas. Foi este o caso do **TAIL**, onde não introduzi a limitação artificial de 4096 como comprimento máximo de uma «cauda» retirada a um ficheiro, que ao que presumo, tenha originalmente resultado do apertado espaço de memória onde as versões iniciais de **Unix** se desenvolveram, e onde um mínimo de informação de controlo nas estruturas de dados era desejável. Esta opção veio no entanto penalizar este utilitário em termos de velocidade, campo onde **CUT** e **PASTE** me surpreenderam agradavelmente na comparação com os originais a correr na mesma máquina. Quanto ao **HEAD**, não deixa grande espaço à imaginação, pelo que como seria de esperar o desempenho, na mesma máquina Unix, é sensivelmente o mesmo do modelo.

Sem menosprezar a sua utilidade, espero que estes programas sirvam sobretudo para demonstrar que os compatíveis PC não se esgotam no DOS, e hoje mais que nunca o utilizador tem à disposição um conjunto de ambientes de trabalho onde pode escolher um adequado às suas necessidades.



“
Entre os quatro programas, (quase) todos me causaram algumas dores de cabeça. A excepção foi como já devem ter percebido, o **HEAD**, cujo código essencial se resume a duas (!) linhas, sendo o restante «embalagem».
”

HABILIDADES

(PARTE I)

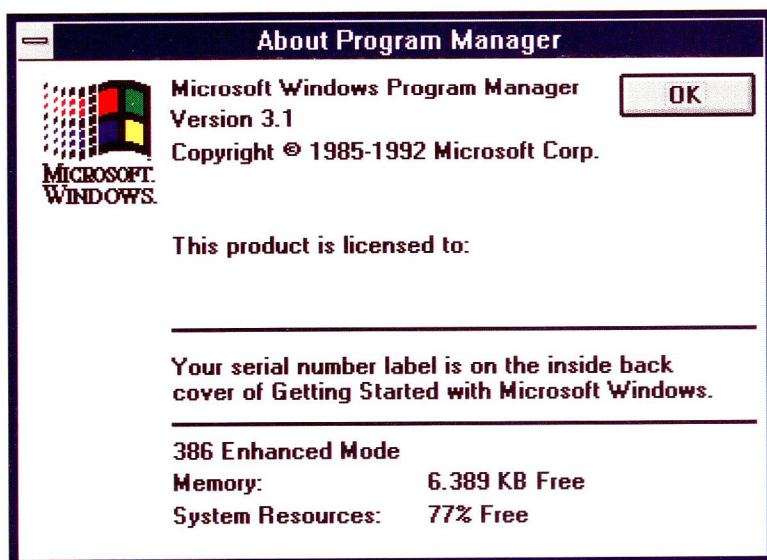
Vamos, nesta rubrica da Spooler, apresentar alguns ecrãs secretos de programas do ambiente Windows. Estes ecrãs secretos têm a particularidade de estarem perfeitamente indocumentados e só surgirem quando é levada a cabo uma determinada sequência de acções pelos utilizadores. O seu aparecimento, com mais ou menos habilidades à mistura serve, no fundo, para mostrar a lista de nomes das pessoas que estiveram envolvidas na elaboração do programa – é uma espécie de assinatura dos autores...

Para ilustrar condignamente este tema escolhemos os três programas que são, muito provavelmente, os produtos mais populares da Microsoft: o WINDOWS, o WORD for WINDOWS e o EXCEL.

HABILIDADES I – O WINDOWS 3.1

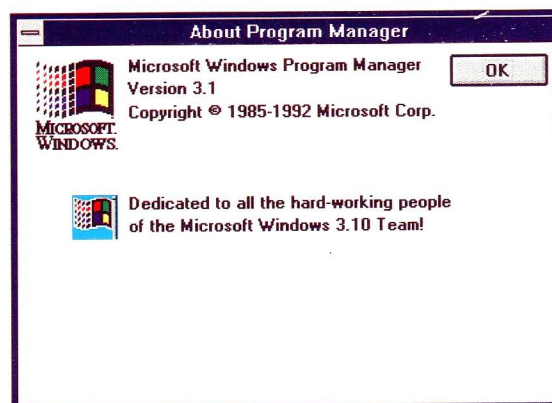
A primeira receita aplica-se ao próprio WINDOWS:

1) No Program Manager, abrir a janela About Program Manager (menu Help). O aspecto desta janela é do mais pacífico que pode existir:



2) Com a janela aberta e premindo simultaneamente as teclas [Shift] e [Control], efectuar um duplo «clic» sobre um dos quadrados coloridos da bandeira do WINDOWS. Premir o botão OK – até aqui nada sucedeu...

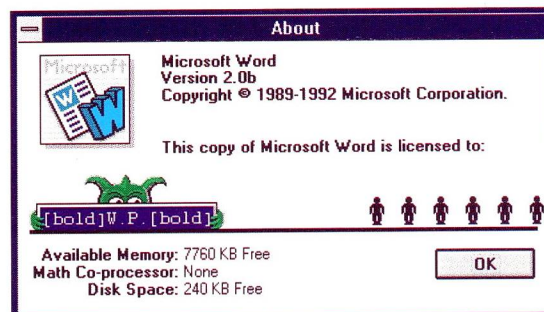
3) Abrir novamente a janela About Program Manager e repetir todos os passos enunciados em 2). Ao efectuar o duplo «clic» (portanto, antes de premir o botão OK), surge no centro da janela uma pequena bandeira animada, ondulado ao vento, e uma inscrição dizendo «Dedicated to all the hard-working people of the Microsoft Windows 3.10 Team!». Neste ecrã não há mais nenhuma novidade, pelo que se prime OK.



4) Abrir novamente a janela About Program Manager e repetir todos os passos enunciados em 2) (uf!). Ao efectuar o duplo «clic» começa então o espectáculo: surge na zona central da janela uma figura apresentando uma lista de legendas em *scroll* ascendente (como no cinema!) onde figuram os nomes das pessoas que trabalharam no desenvolvimento do Windows 3.1.

A figura não é sempre a mesma (há quatro ao todo e surgem aleatoriamente): três figuras são humanas e a outra é um urso (rezam as crónicas que uma das figuras representa o próprio Bill Gates, presidente da Microsoft, mas pensa-se que não será o urso).

Enquanto não for premido o botão OK os nomes passam à desfilada, ficando o programa eternamente em ciclo: convenhamos que podia ter-lhes dado para pior ...

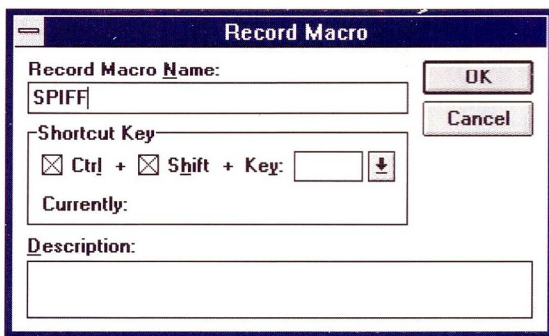


HABILIDADES II – O WORD FOR WINDOWS 2.0

Para este programa, a obtenção do ecrã secreto e da respectiva «historinha» é do mais tortuoso que se possa imaginar...

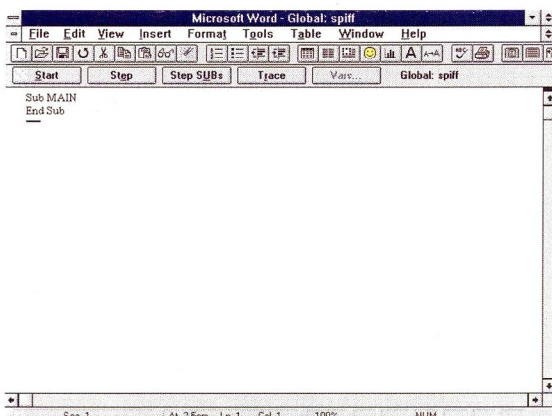
1) Entrar no WORD e abrir um documento novo.

2) Abrir o menu TOOLS, escolher a opção RECORD MACRO, atribuir o nome SPIFF à macro a criar e premir OK.



3) O WORD entra em modo de gravação de macro. Abri novamente o menu TOOLS e escolher a opção STOP RECORDER para terminar a gravação.

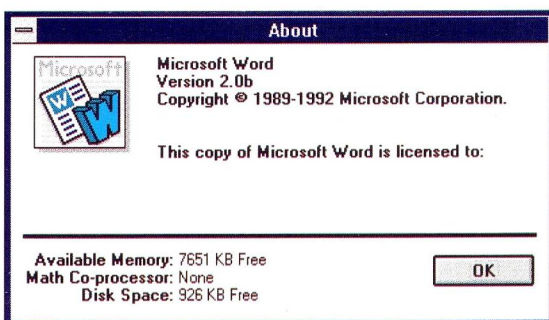
4) Regressar ao menu TOOLS, seleccionar a opção MACRO e editar a macro SPIFF (seleccionar SPIFF e premir o botão EDIT). O WORD fica em modo de edição de macro e mostra o texto da macro, onde estarão apenas as linhas «Sub MAIN» e «End Sub» – estas linhas representam o início e fim de macro e foram colocadas automaticamente pelo WORD.



5) Apagar as duas linhas e terminar a edição, gravando as alterações (menu FILE, opção CLOSE – o WORD perguntará se é pretendido gravar as alterações à macro SPIFF, pergunta à qual se responderá YES).

O bruxedo está terminado! Para ver o resultado, basta escolher a opção ABOUT... do menu HELP e efectuar um «clíc» com rato sobre o símbolo do WORD.

Aparecem seis homenzinhos a correr da esquerda para a direita. De repente, e também do lado esquerdo, surge um monstro «verde e peçonhento», segurando uma placa com a inscrição «[Bold] W.P.[Bold]» e aterrorizando as pobres criaturas.



E quando tudo parece perdido, eis que o logótipo do WORD ganha vida e esmaga (sim, leram bem, ESMAGA!) o monstro. Os homenzinhos, são e salvos, não cabem em si de satisfeitos e vão saltitando até saírem da janela.

Tudo termina com um fogo de artifício para comemorar

a vitória do WORD sobre o WORDPERFECT; entretanto, são mostrados os nomes dos carinhosos rapazes que escreveram o WORD...

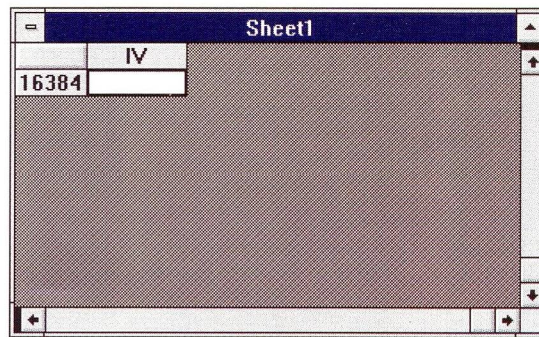
Curiosamente, na versão mais recente do WORD (2.0c) já não surge o «monstro verde» e é apenas mostrado o «fogo de artifício». Remorsos da rapaziada da Microsoft?

HABILIDADES III – O EXCEL 3.0

No caso do EXCEL 3.0, a história anda toda à volta da última célula da folha. A receita é a seguinte:

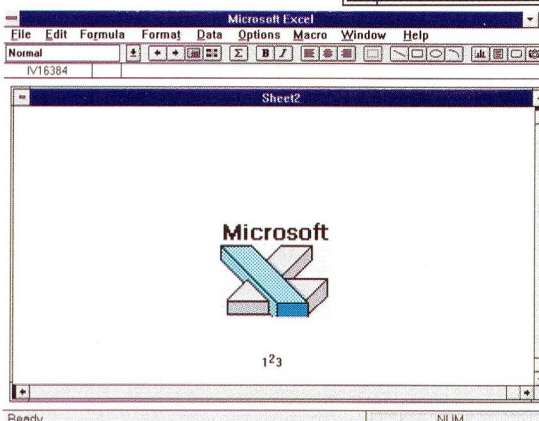
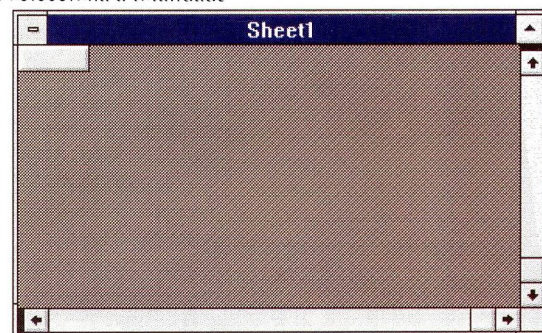
1) Posicionar o cursor (ou a célula activa, como preferirem) na coordenada IV16384 (para este fim pode usar-se a tecla F5, que efectua o GOTO).

2) Depois, e com o auxílio das *scroll bars* vertical e horizontal, deslocar a folha o mais para zona superior esquerda possível, de modo a tornar a célula IV16384 na única visível na janela.



Em seguida, reduzir a altura e largura da célula a zero – pode fazer-se actuando sobre a zona dos títulos ou através dos comandos FORMAT ROW HEIGHT e FORMAT COLUMN WIDTH. Depois de efectuada esta operação, apenas resta na janela o botão que selecciona a totalidade da folha.

4) A partir deste momento, sempre que for premido o referido botão de selecção da folha (canto superior esquerdo da janela) despoleta-se a «historinha», que não é mais que um feroz esmagamento de um pobre e indefeso logótipo do LOTUS (claro!) por um gigantesco logótipo do EXCEL.



E não há mais habilidades a apresentar. Por isso, é com esta demonstração de não convivência, *fair-play* e amor entre os Homens que nos despedimos: até breve.

 Paulo Bernardo

**COLABORAÇÃO
DO LEITOR**

PLANO v1.10

SEG, 93-07-05	TER, 93-07-06	QUA, 93-07-07	QUI, 93-07-08	SEX, 93-07-09
Esta semana contém exemplos de utilização dos espaços, de escrita.	7.00 Levantar e ir para o emprego.	7.45 Seguir as directivas do dossier de 3ª feira.	FERIADO !!!!	10.00 Atender à conferência de imprensa na sede do partido.
Estes exemplos podem ser apagados, primeiro editando o dia que se quer, carregando em 2 para 2ª feira, 3 para 3ª feira,	9.00 Reunião com o conselho de gerência.	10.00 Ligar para a SPOOLER BBS e puxar com o modem a última versão do SCAN anti-virus.	Ir visitar o museu com a família.	12.00 Elaborar o relatório semanal.
	11.30 Tratar das tarefas pendentes.		Ler a Spooler	
	12.00 ALMOÇAR com o grupo.			

Parece uma Agenda, mas não é. Mesmo que fosse, desde que tivesse algo que fugisse à banalidade e introduzisse elementos de programação com interesse para os leitores, não deixaríamos de publicá-la. Mas este PLANO possui algo mais, satisfazendo todos os requisitos ao agrado da Spooler. Será o Nuno Zoio a esclarecer-nos:

INSTRUÇÕES

Este PLANO pretende ser um substituto à tradicional agenda de bolso. Pode ser preparado de modo a aparecer automaticamente aquando da ligação do computador e, nesse instante, mostrar no ecrã, um mapa com os dias úteis da semana actual e, em destaque, a coluna correspondente ao dia de hoje.

Cada coluna, correspondente aos dias da semana actual, é constituída por um espaço de dimensão 15 por 40 caracteres onde se pode escrever o que quer que seja, podendo-se assim fazer uma lista de actividades para o dia de hoje, acompanhada da hora a que cada uma deve ser feita e, depois, na visualização, o texto é escrito sobre um *background* de cor diferente (com o objectivo de proporcionar melhor visualização), consoante sejam tarefas para a tarde ou para a manhã desse dia.

Mas o espaço de escrita é livre e pode-se escrever nele qualquer coisa para além do anterior exemplo de uma lista de tarefas; pode-se também utilizar o espaço para fazer um memorando, um diário pessoal, ou no espaço de um determinado dia, escrever também o nome do amigo que faz anos nesse dia.

Recue nas semanas com **PgUp**, até chegar à semana de 93-Jul-9. Esta contém alguns exemplos de utilização dos

espaços de escrita. Depois, sempre que ligar o computador, ele pode executar o PLANO automaticamente, mostrando toda a semana actual (com base na data/hora do sistema), com todas as informações que nós escrevemos para aqueles dias daquela semana, apresentados cada um em colunas separadas e simultaneamente (esta ideia da apresentação integral de todos os dias da semana foi introduzida com o intuito de, assim, se poder planear melhor as nossas decisões/acções, com base nas informações das actividades agendadas para os próximos dias).

É de salientar que em qualquer altura se pode avançar para a próxima semana ou para a anterior, bastando para isso carregar um **PgDn/PgUp**, imprimir qualquer semana visualizada só carregando em **P** e para voltar à semana actual, basta pressionar **Home**. Pode-se editar o texto para qualquer dia de qualquer semana futura ou passada, com um editor muito versátil e completo, com muitas facilidades, com vista à cómoda introdução/alteração de texto.

O PLANO tem incorporado um **HELP** (Ajuda) tanto para as operações de consulta como para o editor, indicando todas as teclas e o seu significado, e pode ser chamado em qualquer altura bastando carregar em **F1**.

Tem um relógio sempre activo e tem a indicação do N.º da semana do ano que estamos a visualizar actualmente (tal e qual como nas agendas de papel...). Este PLANO tem também incluído um calendário infinito, tanto para trás como para a frente, pelo menos teoricamente (só testei até ao ano 3000 AC/DC...)

Em suma, o meu intuito foi o de fazer um programa o mais *User Friendly* possível, sem toda aquela complicação de menus *pull-down* e operações arvezadas que atrapalham muito os principiantes de informática não tirando, contudo, a vasta gama de possibilidades e potencialidades que oferece, por exemplo, no uso quotidiano de escritório (que foi até a sua razão de nascimento!).

Eis a descrição das teclas:

a) Em modo normal/visionamento

F1	– Uma pequena Ajuda lembrando as teclas principais.
PgDn	– Avançar uma semana.
Pgup	– Retroceder uma semana.
Home	– Voltar à semana actual.
P	– Imprimir semana visualizada.
Esc	– Saír para o DOS.
2 a 6	– Editar na semana visualizada, respectivamente, a coluna da 2ª feira, 3ª feira, etc.

b) Em modo de edição

F1	– Uma pequena Ajuda lembrando as teclas do editor.
Setas de Direcção	– Mover o cursor na tabela.
PgUp	– Mover o cursor 10 linhas para cima.
PgDn	– Mover o cursor 10 linhas para baixo.
Control-PgUp	– Mover cursor para a 1ª linha.
Control-PgDn	– Mover cursor para a última linha.
Home	– Mover o cursor para a primeira letra da primeira palavra da linha actual.

End	– Mover o cursor para o fim da última palavra da linha actual.
Tab	– Mover o cursor 6 colunas para a direita.
Shift-Tab	– Mover o cursor 6 colunas para a esquerda.
Enter	– Mover cursor para a 1ª coluna da linha de baixo.
BackSpace	– Apagar carácter à esquerda do cursor.
Del	– Apagar carácter à direita do cursor.
Alt-Del	– Apagar toda a linha actual.
Control-Del	– Apagar todas as linhas do texto!
Insert	– Inserir um espaço no meio de duas palavras.
Esc	– Abortar edição sem gravar.
Control-Enter	– Gravar alterações e saír do editor.

Nota: O programa PLANO vem ainda acompanhado de um utilitário chamado AUTOBOOT (e da sua fonte), que faz a alteração do AUTOEXEC.BAT de modo a fazer correr o PLANO automaticamente aquando da ligação do computador. Pode-se correr este várias vezes, que ele só altera uma vez, além de que o AUTOEXEC.BAT antigo não é apagado: fica em AUTOEXEC.OLD.

 Nuno Zoio

ANSITEST.COM E TESTANSI.COM

São dois utilitários cuja missão é fundamentalmente a mesma: detectar se temos instalado em memória o conhecido *device driver* ANSISYS, ou qualquer um outro cuja missão seja equivalente.

Todos nós utilizamos, se não sempre, pelo menos ocasionalmente, um desses *device drivers*, senão para redefinir o teclado (o que é perigoso por causa das «bombas Ansi»), pelo menos para visualizar alguns ecrãs bonitos e coloridos que foram desenhados com o recurso a códigos do ANSISYS. Agora, se o leitor experimentar visualizar um ecrã preparado para ser só utilizado com o ANSISYS carregado em memória e por um motivo ou outro não o tiver instalado, sabe qual é o resultado, não sabe?

Pois é, verá parêntesis rectos e mais parêntesis rectos e o bonito ecrã transformou-se numa conjunto de caracteres inestético. Para evitar uma situação dessas preparámos estes dois utilitários:

- Se o leitor dispõe de MS-DOS de versão superior à 4.00 e utiliza o ANSISYS que acompanha os utilitários do DOS, experimente utilizar o TESTANSI.COM pois o mesmo utiliza um método limpo e agora já oficialmente documentado pela Microsoft para a detecção deste *device driver*.
- Se o leitor utiliza um substituto do ANSISYS (por exemplo, o EANSI, o TANSI, o VANSI, o NANSI, o NNANSI, o DVANSI, o ZANSI ou um outro



qualquer entre dezenas) pode não obter resultados correctos com o TESTANSI.COM.

Neste caso tente o ANSITEST.COM. A metodologia utilizada não é tão elegante mas funciona quase sempre.

Para testar os nossos programas, execute o ficheiro *batch* ANS.BAT com a seguinte sintaxe:

```
ANS ansitest
ou
ANS testansi
```

Se tiver o ANSISYS carregado delicie-se.

 José Páscoa

PROGRAMAÇÃO GRÁFICA PARA TURBO PASCAL

INTRODUÇÃO

Como já devem saber, os PC têm dois modos de vídeo para funcionar. Um é o modo de texto a que já estamos habituados (DOS, Turbo Pascal, etc.) e o outro, não menos conhecido, é o modo gráfico. Todos o conhecem do Windows, jogos, etc., mas nem todos o sabem programar. Pois

então, aqui está uma boa oportunidade de se iniciarem. É claro que com estes artigos não conseguirão fazer «jogos espetaculares», já que as rotinas para o modo gráfico disponíveis no Turbo Pascal não são as mais indicadas.

Este artigo destina-se aos programadores de Turbo Pascal mas, quem programar em Turbo C, poderá facilmente entender os exemplos utilizados, pois as funções do Turbo C são praticamente as mesmas que os procedimentos e funções do Turbo Pascal.

Este artigo será seguido doutros dois, cada qual destinado a um ou mais assuntos. Neste primeiro explicarei como se inicializa e termina o modo gráfico e ainda uma série de procedimentos e funções relacionados com escrever no ecrã. Tanto este como os outros artigos serão acompanhados dum ficheiro de texto (na disquete) contendo uma descrição de todos os procedimentos ou funções relacionadas. E ainda um programa exemplo em cada um dos artigos, devidamente comentado, para que seja mais fácil entender cada um dos tópicos apresentados. Estes programas deverão correr na maioria das placas. Vamos então começar e não perca o próximo número da **Spooler**...

Driver Gráfico	Valor	Modo Gráfico	Valor	Resolução
CGA	1	CGAC0	0	320x200
		CGAC1	1	320x200
		CGAC2	2	320x200
		CGAC3	3	320x200
		CGAHi	4	640x200
MCGA	2	MCGAC0	0	320x200
		MCGAC1	1	320x200
		MCGAC2	2	320x200
		MCGAC3	3	320x200
		MCGAMed	4	640x200
		MCGAHi	5	640x480
EGA	3	EGALo	0	640x200
		EGAHi	1	640x350
EGA64	4	EGA64Lo	0	640x200
		EGA64Hi	1	640x350
EGAMONO	5	EGAMonoHi	3	640x350
HERC	7	HERCMonoHi	0	720x348
ATT400	8	ATT400C0	0	320x200
		ATT400C1	1	320x200
		ATT400C2	2	320x200
		ATT400C3	3	320x200
		ATT400Med	4	640x200
		ATT400Hi	5	640x400
VGA	9	VGALo	0	640x200
		VGAMed	1	640x350
		VGAHi	2	640x480
PC3270	10	PC3270Hi	0	720x350
8514	6	IBM8514Lo	0	640x480
		IBM8514Hi	1	1024x768

Tabela 1

MODO GRÁFICO COM O TURBO PASCAL

O Turbo Pascal, até à versão 3.x, utilizava um sistema de programação gráfica com base na inclusão (`{ $I <nome do ficheiro> }`) dum ficheiro com as rotinas destinadas à programação em modo gráfico. Apartir da versão 4.0, com a criação das Units (*TPU), este sistema foi alterado para utilizar esse método. A unit Graph está presente no ficheiro Graph.TPU, que acompanha o TP. Para além deste ficheiro, existem os *drivers* gráficos e as fontes (tipos de letra). Os *drivers* (*.BGI) são ficheiros que contém toda a informação que o TP necessita para inicializar o modo gráfico. As fontes (*.CHR) são os ficheiros onde estão definidos os diferentes tipos de letra. Normalmente existem só quatro fontes, mas o TP 7.0 já traz 10... Se estes ficheiros estiverem presentes, a inicialização do modo gráfico deverá decorrer sem problemas. Vamos então passar à inicialização...

INICIALIZAÇÃO E TERMINAÇÃO GRÁFICA

O TP utiliza, quando trabalhando em modo gráfico, um algoritmo pré-determinado e que consiste em três partes principais:

A primeira parte é a inicialização gráfica. Aqui indica-

-se qual o modo gráfico com que se quer trabalhar (é possível detectar o modo gráfico) e o computador encarrega-se de alterar o modo de vídeo. A segunda parte é essencialmente o corpo do programa, onde se utilizam os vários procedimentos e funções do TP para o modo gráfico e se define a estrutura do código.

A terceira parte consiste em terminar o modo gráfico e passar ao modo de vídeo anterior à inicialização.

Para melhor explicar o algoritmo de inicialização gráfica, vou transcrever um excerto dum programa, procedendo de seguida à sua análise:

```

Uses Graph;{declara a utilização da unit GRAPH}
Var GraphDriver,GraphMode:integer; {as variáveis vão
conter os valores do driver (.BGI) e do modo gráfico}
Begin {inicialização gráfica}
  DetectGraph(GraphDriver,GraphMode); {detecta os
valores default para as variáveis}
  InitGraph(GraphDriver,GraphMode,'C:\TP\BGI');
{inicializa o modo gráfico, segundo os valores das duas
variáveis}
  If GraphResult<>GrOk then Halt;
  {corpo do programa} .
.
.
{terminação gráfica}
CloseGraph;
End.

```

A primeira linha destina-se a declarar o uso da unit GRAPH (e assim poder fazer uso das rotinas destinadas ao modo gráfico). A seguir, na segunda linha, declaram-se duas variáveis: **GraphDriver**, que vai conter o valor do driver gráfico e **GraphMode**, destinada a conter o valor do modo gráfico. Estas variáveis têm de ser do tipo integer.

A tabela 1 lista os valores que se podem utilizar (até à versão 7.0).

As duas primeiras colunas do quadro definem o *driver* gráfico (constante e valor), as duas seguintes os vários modos gráficos de cada *driver* (constante e valor) e a última a resolução correspondente a cada modo gráfico. Continuando a análise do programa:

A quarta linha detecta quais são os valores *default* para as variáveis (normalmente os valores máximos), conforme o *hardware* disponível. A quinta linha inicializa o modo gráfico e utiliza os parâmetros *driver* gráfico e modo gráfico. A *string* contém o *path* para o *driver* gráfico. Se a *string* estiver vazia será considerado o directório corrente.

GraphResult (sexta linha) é uma função que devolve o resultado da última operação efectuada em modo gráfico. Os seus valores possíveis estão indicados na Tabela 2.

O programa irá abortar se algum problema tiver ocorrido durante a inicialização. A penúltima linha termina o modo gráfico, voltando ao modo de vídeo anterior à inicialização. Consulte a disquete para analisar uma descrição destes e outros procedimentos relacionados com a inicialização do modo gráfico.

ESCREVER NO ECRÃ...

Normalmente um jogo ou um programa

Constante	Valor	Significado
GrOk	0	Não ocorreu nenhum erro
GrNoInitGraph	-1	Gráficos não inicializados. Use InitGraph
GrNotDetected	-2	Hardware não detectado
GrFileNotFound	-3	Ficheiro não encontrado
GrInvalidDriver	-4	Driver Gráfico inválido
GrNoLoadMem	-5	Erro na alocação da memória
GrNoScanMem	-6	Memória não varrida
GrNoFloodMem	-7	Memória cheia
GrFontNotFound	-8	Ficheiro de fontes não encontrado
GrNoFontMem	-9	Fontes fora da memória
GrInvalidMode	-10	Modo Gráfico inválido
GrError	-11	Erro genérico
GrIOError	-12	Erro de Input/Output
GrInvalidFont	-13	Fonte inválida
GrInvalidFontNum	-14	Número de fonte inválido

Tabela 2

XPVIR 31



PROTECÇÃO TOTAL PARA LAN'S E UNIX

IPS da NSSI – As UPS INTELIGENTES

Importador e Distribuidor exclusivo:

ENAFER – EQUIPAMENTOS ELÉCTRICOS, LDA

Qta D. João, Lt 1 r/c Esq. – 3000 COIMBRA – Telef.: (039) 40 13 10/40 45 55/40 45 61 – Fax: (039) 40 45 49
 Tv. Sr. de Matosinhos, 74 – 4400 GAIA – Telef.: (02) 712 42 29/712 44 49 – Fax: (02) 712 45 28

em modo gráfico utiliza sempre mensagens no ecrã para comunicar com o utilizador. Mas, como certamente já repararam, as letras são muitas vezes de diferentes tamanhos e feitios em cada parte do jogo. Isto deve-se a ser possível definir tipos de letra diferentes do **default** (o que está presente quando inicializamos o modo gráfico). O TP já tem algumas fontes definidas que podem ser utilizadas por qualquer programa escrito para o modo gráfico. Considere o seguinte excerto dum programa:

```

settextstyle(TriplexFont,HorizDir,1);
{redefine o estilo dos caracteres}
settextjustify(CenterText,CenterText);
{redefine o tipo de justificação}
outtext('Este texto será escrito no centro');
{escreve uma string no ecrã} .
    
```

A primeira linha deste programa altera o estilo dos caracteres. O primeiro parâmetro define a fonte a ser usada. Neste caso será a Triplex, mas poderia ser outra das listadas abaixo:

Constante	Valor
DefaultFont	0
TriplexFont	1
SmallFont	2
SansSerifFont	3
GothicFont	4

Estas são as fontes disponíveis até à versão 6.0 do T.P. É claro que se podem incluir outras fontes definidas pelo utilizador, mas isso será tratado mais tarde (3º artigo).

O segundo parâmetro define a direcção em que o texto será escrito. Pode ser Horizontal ou Vertical, conforme pode verificar na tabela:

Constante	Valor
HorizDir	0 da esquerda para a direita
VertDir	1 de cima para baixo

O terceiro parâmetro define o tamanho com que serão escritos os caracteres. Cada fonte diferente tem diferentes tamanhos, podendo numa fonte o 1 ser muito pequeno e noutra ser já maior. Por isso não existe nenhuma tabela.

A segunda linha define qual o tipo de justificação, horizontal e vertical, que o procedimento de escrita deverá ter em conta. O primeiro parâmetro define a justificação horizontal e o segundo a vertical. A seguinte tabela mostra quais os valores possíveis:

Constante	Valor
CenterText	1 vertical/horizontal
LeftText	0 horizontal
RightText	2 horizontal
BottomText	0 vertical
TopText	2 vertical

A justificação toma como referência a posição corrente no ecrã. Assim, se a justificação for RightText, BottomText (default), os caracteres serão escritos à direita e abaixo da posição corrente.

A terceira linha escreve, com os estilos definidos, uma *string* na posição corrente do ecrã. A complementar este procedimento existe outro, o **Outtextxy(x,y:integer;text:string)**; que escreve também uma *string*, mas nas posições indicadas por *x* e *y*. Se a *string* passada como parâmetro sair dos limites do ecrã, esta será cortada, aparecendo apenas o que couber (não passa para a linha seguinte). Este procedimento apenas aceita *strings*, sendo por isso necessário converter todos os valores numéricos em *strings*, antes de os utilizar com **outtext** (utilizar o procedimento **str**).

É possível saber qual é o estilo de escrita (tipo de letra e justificação) corrente. Para isso utiliza-se o procedimento **GetTextSettings(var info:TextSettingType)**; O tipo da variável está definido da seguinte forma:

```

Type TextSettingsType = Record
Font : word;
Direction : word;
CharSize : word;
Horiz : word;
Vert : word;
end;
    
```

Cada campo do **record** conterá o valor corrente para o item correspondente.

Se se quiser calcular a altura em pixels duma *string* (conforme o estilo de letra em uso) pode-se utilizar a função **TextHeight(text:string):word**; Para calcular o comprimento, também em pixels, utiliza-se a função **TextWidth(text:string):word**; Estas duas funções são normalmente utilizadas para calcular a posição do ecrã em que a *string* será escrita.

Estes procedimentos e funções são descritos no ficheiro ESCREVER.TXT. Não deixe de o consultar.

NA DISQUETA

Na disquete deste número poderão encontrar dois ficheiros contendo a descrição dos procedimentos e funções mencionados no artigo, assim como de alguns outros relacionados com o tema. O programa exemplo está no ficheiro DEMO1.EXE e as fontes no DEMO1.PAS. Este programa consiste numa demonstração das fontes, já que ainda não avançámos o suficiente para fazer melhor. A acompanhar irão os ficheiros das fontes e todos os *drivers* fornecidos pelo TP (o programa funciona em todas as placas).

NO PRÓXIMO NÚMERO...

No próximo número serão tratados os seguintes temas:

- Procedimentos de desenho
- Cores

O primeiro tema tratará de procedimentos relacionados com desenho, coordenadas, estilos de preenchimento, etc.. Nas cores falarei, além de como alterar as cores de escrita, de redifinição de palettes, etc.

Desta vez é tudo. Até ao próximo número.

 Pedro Andrade

SPOOLDOS

**COLABORAÇÃO
DO LEITOR**

O Ricardo e o Luciano continuam a apostar no Basic, mostrando-nos que esta linguagem é capaz de fazer o mesmo que as ditas de «elite». Experimentem este SPOOLDOS e editem as fontes para ver como se faz.

Nunca se sentiram frustrados quando o DEL apagou um ficheiro (daqueles que são mesmo para desaparecer), e mais tarde, para vosso espanto, o ficheiro reapareceu pois alguém o «desapagou»? E nunca tentou você ler um ficheiro com o comando TYPE, e viu aquela coisa a passar «a mais de 300 à hora» parecendo um Ferrari, numa autoestrada estrangeira (pois é sabido que as nossas estradas mais parecem pistas de motocross, ou se preferirem queijo suíço)?

Para tal criámos o Spooldos. Não, não foi um novo sistema operativo. O Spooldos é um conjunto de 4 pequenos utilitários, que resolve os exemplos acima dados. Assim, o conjunto tem na sua constituição:

KILL – Programa que apaga o ficheiro «atrocitando-o».

Sintaxe: Kill <path\nome do ficheiro> Nota: Não se podem utilizar wildcards (* e ?).

Uma vez executado o programa o ficheiro fica irre recuperável.

FTYPE – Programa que exhibe no monitor o ficheiro desejado com uma velocidade regulável através das teclas + e - . A combinação de teclas **Ctrl+Break** termina a execução do programa.

Sintaxe: FTYPE <path\nome do ficheiro>

Uma das vantagens do FTYPE é a capacidade de aumen-

tar a velocidade até 3 vezes mais rápido que o comando TYPE, o que pode ser uma grande vantagem em computadores lentos; por outro lado em computadores rápidos podemos reduzir a velocidade de modo a se poder ler alguma coisa.

O procedimento deste programa é:

- Abre o ficheiro desejado para consulta
- Lê uma linha do ficheiro
- Imprime no ecrã essa linha
- Vê se foram premidas as teclas + ou - para regular a velocidade
- Antes de ler de novo o ficheiro, faz a pausa que regula a velocidade

Quanto ao KILL, primeiro abre o ficheiro para escrita destruindo todo o conteúdo do mesmo; após esta «atrocidade», o KILL apaga o ficheiro, assim caso o tente «undeletar» este ficheiro terá apenas 0 bytes.

SURPRESA

O Spooldos é ainda constituído por um utilitário que compara ficheiros ASCII (*American Standard Code Information Interchange*) assinalando as diferenças, de nome COMPARAR. E ainda o SPEED, um utilitário que, através da cronometragem de minunciosos e complexos cálculos trigonométricos, dá-lhe a relação de velocidade do seu computador perante um processador Intel 8086 original a 4.77MHz.

Esperamos que gostem de utilizar o Spooldos e que este vos seja muito útil.

 **Ricardo Jorge Feliciano Lopes Pereira**
Luciano Manuel Moleirinho de Oliveira

LISTA

O António Manuel continua a presentear os programadores com pequenos e «úteis» utilitários. O LISTA é mais um para a «lista».

Devido à mania que eu tenho de escrever todos os meus programas em letra maiúscula, decidi elaborar um pequeno utilitário que me permita passar um ficheiro de texto de maiúsculas para minúsculas, ou vice-versa. «Lista» permite passar um ficheiro de texto (ou para minúsculas, ou para maiúsculas) para o ecrã ou, então, para um outro ficheiro de texto.

No que me diz respeito, o «Lista» tem-me sido muito útil. E espero que também o seja para todos os «informáticos».

MODO DE UTILIZAÇÃO:

LISTA Opção1 <Ficheiro1> [<Ficheiro2>]

Opção1 -> /M para maiúsculas;
/m para minúsculas;

Ficheiro1 -> Nome do ficheiro de texto a passar ou para minúsculas, ou para maiúsculas;

Ficheiro2 -> Nome do ficheiro onde será gravado o texto «transformado».

Exemplo: Lista /M Lista.Pas Lista.Txt (Copia todo o conteúdo do ficheiro de texto «Lista.Pas», para maiúsculas, num novo ficheiro «Lista.Txt»)

 **António Manuel Monteiro Pereira**

INFORMÁTICA EM NOTÍCIA



SPACE SIMULATOR E NOVA VERSÃO DO FLIGHT SIMULATOR DA MICROSOFT

A Microsoft Corporation anunciou quatro novos produtos de simulação do sector de produtos de entretenimento: o Microsoft Flight Simulator 5.0, para compatíveis; dois novos cenários para utilização com Microsoft Flight Simulator, Microsoft New York e Microsoft Paris; o Microsoft Space Simulator para exploração e viagens espaciais.

O MS New York e o MS Paris oferecem aos utilizadores a possibilidade de experimentar simulações gráficas extremamente realistas e geograficamente correctas destas cidades.

O MS Space, por seu lado, oferece um óptimo meio educativo e de entretenimento para a exploração do Universo, a partir de computadores pessoais.

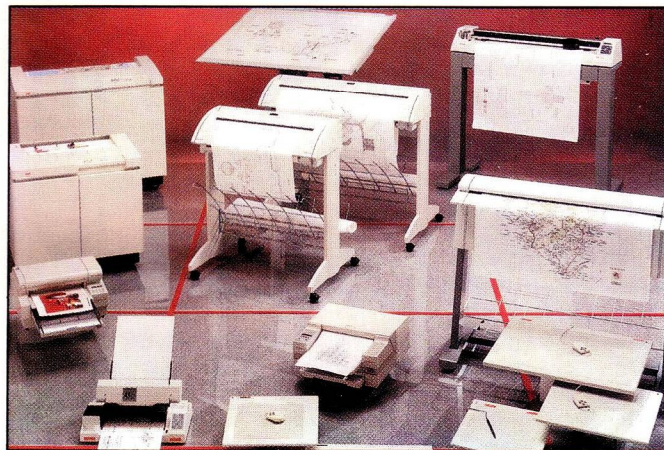
MSFT Lda (Microsoft Portugal)
Telef. (01) 441 22 05 – Fax: 441 21 01

CORELDRAW 4.0

A Corel Corporation apresentou nos últimos dias de Maio, o CorelDRAW 4.0, a mais recente versão do pacote de gráficos, para ambiente Windows. Para além das inovações introduzidas nos módulos CorelDRAW, CorelPHOTO-PAINTE, CorelCHART e CorelSHOW, o CorelDRAW 4 inclui agora o CorelMOVE, um novo programa para criar animações.

De acordo com a estratégia adoptada pela Corel de comercializar o CorelDRAW 3 e CorelDRAW 4 simultaneamente, este último será posicionado como o produto para utilizadores cujos requisitos gráficos ultrapassam as capacidades oferecidas pelo CorelDRAW 3.

Representante: Laserdata, Lda.
Telefs.: (01) 716 10 01/2/3/4
Fax: 716 33 63



OCÉ PORTUGAL – EQUIPAMENTOS GRÁFICOS

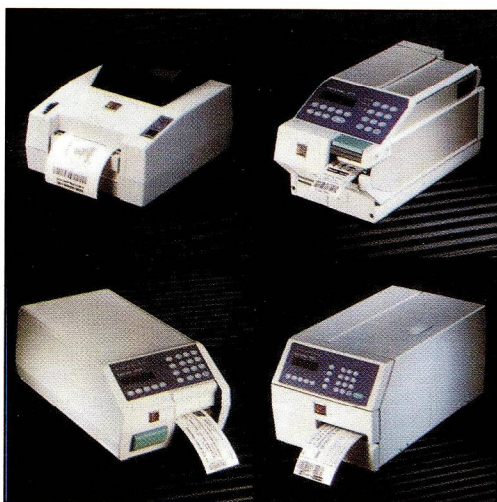
O Grupo Internacional OCÉ anuncia a abertura da Companhia OCÉ Portugal, em Lisboa, direccionada para a comercialização de sistemas de impressão de grandes formatos (Plotters) e de consumíveis. Depois de um estudo intensivo sobre o Mercado de Periféricos CAD/CAM em Portugal, a OCÉ decidiu dar este passo. A nova Companhia Operativa em Lisboa, começou oficialmente a sua actividade em 1 de Junho de 1993.

A inauguração desta empresa coincide com o lançamento de novos modelos, substituindo os plotters térmicos e plotters de canetas existentes e oferecendo a última novidade em tecnologia de Impressão.

O Grupo Internacional OCÉ, com sede

na Holanda, está presente em 80 países e tem companhias operativas em 25 países. O Grupo OCÉ, que emprega mais de 12 000 pessoas em todo o Mundo, comercializa uma extensa gama de produtos e serviços para a reprodução e apresentação de Informação no papel, tanto para sistemas de engenharia, como para sistemas de escritório. Esta gama de produtos é largamente desenvolvida e produzida pela própria OCÉ e inclui fotocopiadoras, impressoras, plotters e respectivos consumíveis, materiais sensíveis à luz e películas gráficas.

OCÉ PORTUGAL, Lda.
Telef.: (01) 352 96 69 – Fax: 351 96 69



IMPRESSORAS DE CÓDIGOS DE BARRAS UBI

UBY EASY CODER é uma geração de impressoras de códigos de barras da United Barcode Industries. Desenvolvidos com o objectivo de proporcionar uma fácil operação e instalação as suas aplicações vão desde a impressão em trabalhos «leves» de escritório até ao trabalho em ambientes rigorosos e exigentes. Sob a denominação UBIEASYPACK cada modelo é apresentado como uma solução completa para funcionar em ambiente Windows. O conjunto inclui uma impressora EASYCODER, o UBILABEL SHOP

(software para desenho e impressão de etiquetas), drivers para Windows e cabos de ligação.

Para além das versões com ou sem teclado, controladas por computador, existem ainda UBI EASYCODERS STAND ALONE cujo funcionamento pode ser controlado através dos seus teclados e visor sem necessidade de ligação ao computador.

Representante: Datacomp, Lda.
Telefs.: (01) 795 18 74/5/6/7
Fax: 795 19 28

NOVOS PORTÁTEIS «THINKPAD» DA IBM

A IBM anunciou 4 novos computadores pessoais portáteis, dimensão «notebook» A4. Trata-se de dois modelos monocromáticos e dois policromáticos, os 720 e 720 C, incorporando tecnologia PCMCIA.

O processador escolhido foi o 486SLC2-50/25 MHz da IBM, com 16 Kb de cache interna. A memória base de origem poderá ser 4 Mb ou 8 Mb, consoante seja um modelo com sistema operativo DOS ou OS/2 pré-instalado. Discos fixos com capacidades compreendidas entre 120 e 160 Mb.

Os ecrãs dos modelos monocromáticos tem 9,5" (diagonal) e são STN (Super-twisted Neumatic), com resolução VGA, 640 x 480 em 64 tonalidades de cinzento, provenientes de uma paleta com 4096 tons.

Nos modelos policromáticos, os ecrãs têm 10,4" e são TFT (Thin Film Transistor), com resoluções VGA, 640 x 480 em 256 cores de uma paleta também de 4096 cores. Ambos os ecrãs possuem iluminação lateral.

BBSpooler

(1) 475 22 89

Dias úteis: das 18.30 às 9.00 horas
Feriados e fins-de semana: 24 horas

SHOW v1.0

**COLABORAÇÃO
DO LEITOR**



Mais uma vez o BASIC demonstra as suas capacidades. O Ricardo Pereira construiu um interessante programa para emular um «placard electrónico» nos nossos computadores. Leia o artigo atentamente, pois são inúmeras as possibilidades de controlo do texto «a correr».

Este programa é baseado nos *placards* electrónicos que se vêem nas lojas. Serve para «fazer passar» uma mensagem pelo monitor mas em grande. Assim, a mensagem dada ao SHOW V1.0 passará pelo ecrã em letras gigantes podendo o modo e tamanho ser controlado por nós. A sintaxe do SHOW é a seguinte:

SHOW <nome de ficheiro com texto ASCII> – neste caso o SHOW usa o conteúdo do ficheiro como mensagem.

SHOW <mensagem> – neste caso o show usa os parâmetros como mensagem.

SHOW – dá o texto de ajuda.

As letras podem ter três tamanhos. O tamanho mais pequeno, utilizando o parâmetro /P, usa letras normais. Quando não se usa parâmetro de tamanho, as letras terão uma dimensão média, o que lhes permite mover-se pelo ecrã. Através do parâmetro /G atinge-se o maior tamanho, o que permite um aproveitamento quase total do tamanho do ecrã.

Durante a visualização da mensagem é possível aumentar ou diminuir a velocidade através das teclas + e -. É ainda possível escolher se queremos o texto a passar no mesmo ponto do ecrã ou a subir e a descer, através da tecla F1 (só válido para tamanho médio). A F2 permite-nos escolher se queremos o texto a passar da direita para a esquerda ou vice-versa. A tecla F5 permite-nos memorizar uma peque-

na música a ser tocada num pequeno órgão via teclado. Ao sair dessa função a música em memória passará a ser permanentemente tocada enquanto o programa corre.

Para mudar a música é só repetir o procedimento. Para «calar» o programa e só necessário carregar o órgão e gravar sem tocar qualquer música, bastando para isso premir duas vezes a tecla SPACE.

O programa possibilita ainda a impressão do texto actualmente visionado. Para tal basta carregar na F6. Imediatamente nos será perguntado se desejamos uma pequena moldura circundando o texto. Após respondermos, o texto será impresso, dependendo do tamanho do resultado da letra actualmente escolhida (o tamanho mais pequeno não possibilita impressão).

O FUNCIONAMENTO DO PROGRAMA.

O SHOW ao ser executado verifica se tem algum parâmetro, se não tiver executa a ajuda, caso contrário tenta encontrar um ficheiro com o nome do parâmetro. Caso o encontre mostra o texto existente nesse ficheiro. Se esse ficheiro não existir usa o parâmetro como mensagem.

A mensagem é depois transformada em 5 cadeias que vão conter cada uma os caracteres (espaço ou quadrado) necessários à formação das letras com 5 linhas de altura; cada cadeia contém os dados para uma linha.

Essas linhas são depois usadas, sendo impressos no ecrã 80 caracteres de cada cadeia por linha. Os 80 caracteres vão sempre mudando de modo a que mensagem passe toda. O tamanho maior limita-se a triplicar o número de caracteres por letra, sendo assim uma cópia do tamanho médio.

A mensagem repete-se infinitamente até ser premeida a tecla ESCAPE. Para mais informações leia o ficheiro em Basic.

Espero que gostem do programa

Ricardo Pereira

“

Este programa é baseado nos placards electrónicos que se vêem nas lojas. Serve para «fazer passar» uma mensagem pelo monitor mas em grande.

”

**COLABORAÇÃO
DO LEITOR**

ASTÚCIAS

Esta rotina que envio é uma variante do procedimento **Exec** do Turbo Pascal. A rotina será especialmente útil a todos aqueles que nos seus programas façam um *shell* ao DOS ou a outro subprocesso e fiquem sem memória suficiente e o processo acabe por abortar, ou por correr com pouca memória.

A rotina executa, tal como o procedimento **Exec**, um *shell*, mas com uma particularidade: pode fazer um *swap* para o disco do *Heap* e da *buffer* de *overlays* e liberta a memória ocupada por estes para uso durante o *shell*. Contudo, esta técnica tem um senão: durante o *shell* não se podem instalar TSR, pois a memória que estes vão ocupar pode ser a memória libertada pela rotina, e que depois vai ser devolvida ao programa e o seu conteúdo substituído pelo conteúdo do ficheiro de *swap* quando o controlo for devolvido ao programa.

Depois de deixar de ser necessário, o ficheiro utilizado no *swap* não será apagado, mas ficará com um tamanho de zero bytes. Se por qualquer razão a rotina de *swap* não conseguir lêr o dito ficheiro quando o controlo for devolvido ao programa, ela abortará o programa, pois o conteúdo da memória não seria restaurado e então os resultados seriam imprevisíveis, sendo o mais provável um *crash* completo da máquina.

Depois desta já longa explicação vou então passar a descrever a rotina.

A rotina é uma função que dá pelo nome de **ShellSwap** e aceita os seguintes parâmetros:

- **ProgName**: aqui indique o nome do programa que deverá ser corrido como segundo processo (normalmente o **COMMAND.COM** ou outro programa qualquer).

- **Command**: aqui indique a linha de comando que deverá ser passada ao subprocesso.
- **SwapWhat**: neste parâmetro indique à função o que quer libertar. Pode ter dois valores definidos pelas duas constantes **swXXXX**:

swHeap : liberta e faz apenas um *swap* do *Heap*.

swOvr : liberta e faz um *swap* da *buffer* de *overlays* e do *Heap*.

A função devolve um valor booleano: **True** se conseguiu criar o ficheiro de *swap* com sucesso; **False** se não conseguiu criá-lo.

Para tirar o máximo rendimento desta função convém pôr a maior parte do programa num *overlay* para que a *buffer* de *overlays* seja passada para o disco e armazenar o menor número possível de dados no «Data Segmen» que não entra no *swap*, dando preferência ao *Heap*. Se não quiser empatar o seu programa com acessos ao disco para carregar *overlays* pode aumentar o tamanho da *buffer* de *overlays*.

NOTA 1: Não pode ter a *unit Dos* do Turbo Pascal num *overlay*, porque a função aqui apresentada recorre a essa *unit*, logo se ela não se encontrar em memória isso resultará num *crash* do computador.

NOTA 2: A *unit* foi testada no Turbo Pascal versão 6.0. O seu uso por versões anteriores à versão 6.0 será no mínimo arriscado, visto a versão 6.0 utilizar uma gestão de memória diferente.

 Bruno Moscão

**COLABORAÇÃO
DO LEITOR**

TRANCA

O António Monteiro Pereira fez bem em insistir na programação do seu TRANCA. Até é interessante estudarmos as diferentes abordagens a um mesmo problema, por parte de programadores distintos.

Este programa serve simplesmente para trancar o teclado do utilizador. Na verdade, iniciei-o no princípio do ano passado mas, para infelicidade minha e por pura coincidência, surgiu nessa altura o KLOCK do colaborador Paulo Bernardo (na **Spooler N.º 12**, se bem se recordam). Por isso, deixei ficar o TRANCA a meio, isto é, incompleto, visto que ambos os programas eram muito parecidos.

No entanto, decidi renovar e completar o meu programa de forma diferente. De facto, o TRANCA necessita unicamente de um código (este encontra-se no ficheiro **CODIGO.TRA** que deverá ser inserido para destrancar o teclado. Só depois de acertar na dita palavra chave é que o teclado fica destrancado e, depois disso, o utilizador ainda poderá mudá-lo (caso assim o entenda).

De referir, ainda, que:

- o tamanho mínimo do código é de 1 carácter;
- o tamanho máximo do código é de 25 caracteres;
- o utilizador ficará a saber quanto tempo esteve trancado o teclado;
- o TRANCA terá de estar sempre acompanhado pelo ficheiro **CODIGO.TRA**, caso contrário não funcionará;
- o código que se encontra no ficheiro **CODIGO.TRA** está encriptado (para tal, obtive uma preciosa ajuda por parte da astúcia publicada na **Spooler N.º 17**);
- o código inicial é **SPOOLER** (em letras maiúsculas).

O modo de utilização é o seguinte:

- TRANCA (mais simples não poderia ser)

Espero que este programa seja de grande utilidade para todos os utilizadores.

 António José Monteiro Pereira

SCADA

(PARTE V)

Nos dois últimos artigos desta série sobre sistemas SCADA – *software* para funções de supervisão, controlo e aquisição de dados baseado em microcomputadores, caracterizámos o ambiente de exploração nomeadamente a nível do sistema operativo e das linguagens de programação. Neste artigo é feita uma introdução ao suporte por sistemas distribuídos às aplicações em tempo real do tipo SCADA de maior complexidade e com mais exigências de integração.

APLICAÇÕES SCADA – SISTEMAS DISTRIBUÍDOS

ARQUITECTURA DISTRIBUÍDA

A utilização de arquitecturas baseadas em múltiplos processadores oferecem grandes capacidades de computação que permitem alargar a dimensão e a complexidade dos sistemas de tempo real (STR).

As funções de sistema (*input/output*, gráficos, etc.) podem estar distribuídas, libertando processadores para as funções de tempo real com uma maior capacidade de resposta.

No entanto, uma maior disponibilidade de cálculo pode não ser condição suficiente para satisfazer as restrições de tempo real, já que o processamento associado à própria distribuição de tarefas e às consequentes necessidades de comunicações aumenta de forma significativa com a complexidade da arquitectura.

Por outro lado, é sempre verdade que a disponibilidade de mais recursos de computação promove solicitações de uma maior funcionalidade para as aplicações que assim consomem recursos.

As arquitecturas distribuídas para STR podem oferecer topologias para interligação de processadores e de circuitos de I/O, que suportem mecanismos para tolerância a falhas e para comunicações mais rápidas e fiáveis, bem assim como para suporte de funções do sistema operativo e mesmo de construções das linguagens de programação.

COMUNICAÇÕES

As redes de comunicações são o suporte necessário aos sistemas distribuídos de tempo real. Além da correcção lógica, as comunicações têm que satisfazer as restrições de tempo impostas pelos processos das aplicações e pelas funções do sistema operativo, para implementar sistemas estáveis e previsíveis.

O sistema de comunicações para STR deve então integrar:

- Encaminhamento (*routing*) dinâmico em tempos determinados.

- Sincronização de processos em tempo real de acordo com a calendarização executada nos processadores.
- Mecanismos para tolerância a falhas.

O desenvolvimento de comunicações com características específicas para aplicações em tempo real tem permitido utilizações mais abrangentes, integrando processamento distribuído em larga escala.

BASE DE DADOS DISTRIBUÍDA

Uma aplicação de tempo real, sendo um sistema de processamento de dados, pode requerer um sistema de gestão de bases de dados (SGBD) para armazenar e disponibilizar informação.

As bases de dados distribuídas são particularmente interessantes nestes sistemas que envolvem processamento paralelo dos dados, com requisitos de velocidade e volume que podem ser críticos.

O controlo do acesso concorrente à base de dados deverá estar sincronizado com os algoritmos de calendarização de tarefas, o que impõe objectivos para os tempos de acesso a dados.

Com a informação distribuída, cada tarefa individual pode trabalhar os seus dados com menos estrangulamentos que um sistema centralizado. Uma falha numa parte do sistema pode ter aqui consequências limitadas, uma característica importante em sistemas que podem ser críticos.

Naturalmente que o uso de SGBD distribuídos implica compromissos, já que os tempos para actualizar e reproduzir as várias bases afectam o tempo de resposta.

ARQUITECTURAS PARALELAS PARA SISTEMAS GRÁFICOS

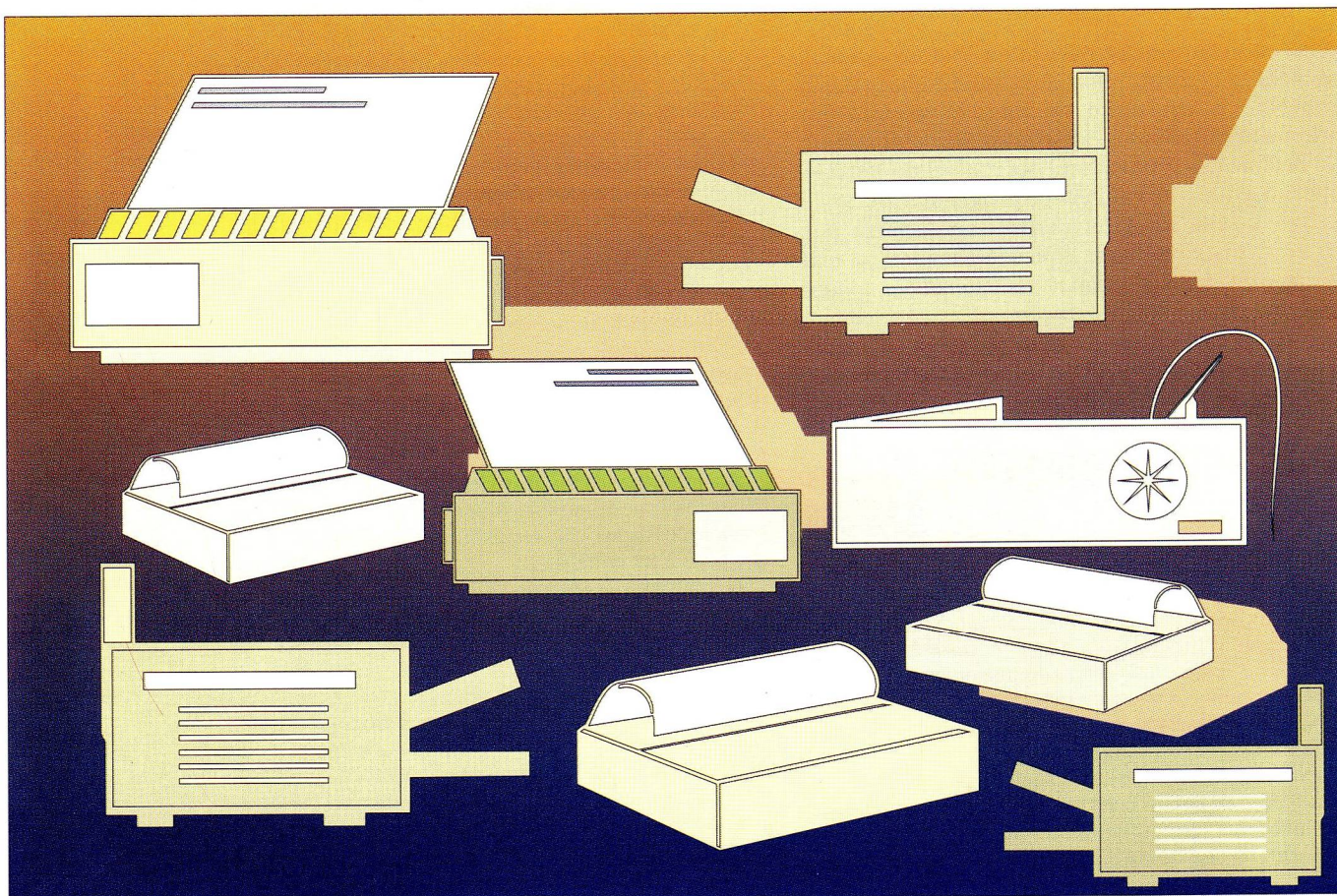
As aplicações interactivas com funções de tempo real na área da gestão de processos, procuram oferecer cada vez mais interfaces gráficas com o utilizador, para uma melhor interacção e compreensão dos múltiplos aspectos relevantes e da sua evolução no tempo.

A informação é comunicada ao utilizador por meio de imagens, símbolos, gráficos, diagramas e quadros sinópticos animados em tempo real, o que acrescenta um grande esforço de computação gráfica ao STR.

As arquitecturas paralelas vêm permitindo soluções para computação gráfica baseadas em multiprocessadores de baixo custo e grande capacidade de actualização de ecrãs, recorrendo, por exemplo, a uma divisão das áreas gráficas por diferentes processadores.

IMPRESSORAS

PARTE II



Dando seguimento ao artigo anterior, nesta segunda parte irão ser abordados alguns dos comandos do modo EPSON. Digo alguns porque, devido à extensão deste artigo, não foi possível incluí-los na sua totalidade neste número da **Spooler**. Assim, no próximo número será concluída a abordagem à programação deste modo. Segue-se a descrição dos comandos para o modo EPSON.

OPERAÇÃO DA IMPRESSORA

Inicialização

ESC @

27 67

Inicializa a impressora

Faz um *reset* à impressora, limpado-lhe o *buffer* e restaurando-lhe todos os seus parâmetros por defeito.

Seleção

DC1

17

Activa a impressora

Faz com que a impressora retorne ao estado ON-LINE depois de ter sido colocada em OFF-LINE pelo comando DC3.

DC3

19

Desactiva a impressora

Faz com que a impressora passe ao estado de OFF-LINE até que esta receba o comando DC1.

Velocidade

ESC s n

27 115 n

Liga/desliga modo de meia velocidade

Se *n* for 0 este modo é desligado, se *n* for 1 este modo é ligado.

ESC i n

27 105 n

Liga / desliga modo imediato

Se *n* for 0 este modo é desligado, se *n* for 1 este modo é

ligado. Neste modo a impressora escreve os caracteres assim que estes lhe chegam ao buffer, ao invés de só os escrever quando estes completarem uma linha.

Direcção de impressão

ESC < 27 60

Selecciona o modo de impressão unidireccional para 1 linha

Depois de receber este comando a impressora escreve só num sentido até receber o comando CR.

ESC U n 27 85 n

Liga/desliga o modo de impressão unidireccional

Se *n* for 0 este modo é desligado, se *n* for 1 este modo é ligado. Este modo é utilizado quando é necessário uma maior precisão no posicionamento da cabeça, quer para texto quer para gráficos.

Controle do papel

ESC 8 27 56

Desinibe o sensor de papel

Torna inactivo o sensor do papel, permitindo escrever até ao limite da folha.

ESC 9 27 57

Inibe o sensor de papel

Cancela o comando ESC 8, permitindo a detecção de falta de papel.

Controle do bit mais significativo

De salientar que estes comandos não funcionam com os modos gráficos ou com caracteres definidos pelo utilizador.

ESC = 27 61

MSB é sempre zero

O bit mais significativo de todos os bytes que chegarem à impressora será forçado ao valor zero, independentemente se o seu valor é zero ou um.

ESC > 27 62

MSB é sempre um

Contrariamente ao comando ESC =, o bit mais significativo é desta vez forçado ao valor um.

ESC # 27 35

Cancela o controle do MSB

Cancela as alterações feitas pelos comandos ESC = ou ESC #.

Som

BEL 7

Alarme sonoro

Faz com que a impressora reproduza um som. De notar que nem todas as impressoras (mesmo sendo compatíveis) têm esta capacidade.

CONTROLE DE DADOS

CR 13

Retorno do carro

Escreve os dados no *buffer* e faz com que a cabeça da impressora regresse à margem esquerda. Facultativamente

poderá também ser feito um avanço de linha, dependendo dos *dip-switch*.

CAN 24

Cancela linha

Remove todo o texto que estiver por imprimir na linha que precede este comando. Os comandos existentes na linha a ser cancelada não são afectados.

DEL 127

Apaga carácter

Apaga o último carácter do *buffer* da impressora, não afectando comandos.

MOVIMENTO VERTICAL

Avanço de folha

FF 12

Form fead

Escreve os dados ainda presentes no *buffer* e avança o papel para o topo da próxima página, de acordo com o tamanho corrente da página.

ESC C n 27 67 n

Define o número de linhas por página

Altera o tamanho da página para *n* linhas. O valor de *n* pode variar entre 1 e 127.

ESC C NUL n 27 67 0 n

Define o tamanho da página em polegadas

Altera o tamanho da página para *n* polegadas. O valor de *n* pode variar entre 1 e 22.

ESC N n 27 78 n

Define o tamanho da margem inferior

O tamanho da margem inferior é alterado para *n* linhas, devendo este valor estar compreendido entre 1 e 127.

ESC 0 27 79

Cancela a margem inferior

Cancela a margem inferior definida pelo comando ESC N.

Avanço de linha

LF 10

Line feed

Escreve os dados presentes no *buffer* da impressora e seguidamente avança para a próxima linha.

ESC 0 27 48

Selecciona 1/8 de polegada de espaçamento entre linhas

Altera o espaçamento entre linhas para 1/8 de polegada (cerca de 3.1 mm).

ESC 1 27 49

Selecciona 7/72 de polegada de espaçamento entre linhas

Altera o espaçamento entre linhas para 7/72 de polegada (cerca de 2.4 mm).

ESC 2 27 50

Selecciona 1/6 de polegada de espaçamento entre linhas

Altera o espaçamento entre linhas para 1/6 de polegada (cerca de 4.2 mm).

ESC 3 *n*27 51 *n*

Selecciona *n*/216 de polegada de espaçamento entre linhas

Altera o espaçamento entre linhas para *n*/216 de polegada. *n* pode variar entre 0 e 255.

ESC A *n*27 65 *n*

Selecciona *n*/72 de polegada de espaçamento entre linhas

Altera o espaçamento entre linhas para *n*/72 de polegada. *n* pode variar entre 0 e 85.

ESC J *n*27 74 *n*

Avança o papel na vertical *n*/216 de polegada

Avança o papel *n*/216 de polegada. *n* pode variar entre 0 e 255.

ESC | *n*27 106 *n*

Recua o papel na vertical *n*/216 de polegada

Recua o papel *n*/216 de polegada. *n* pode variar entre 0 e 255, no entanto para valores maiores que 18 não se pode garantir um funcionamento correcto.

Tab verticalais

VT

11

Tabulação vertical

Avança o papel para o próxima posição de tabulação vertical no canal seleccionado (ESC /). Se nenhum canal estiver seleccionado, o tab será enviado para o canal 0. Se não existir nenhum tab vertical definido (ESC B), o papel simplesmente avança uma linha.

ESC B *n1 n2 ... NUL*27 66 *n1 n2 ... 0*

Define tabs verticais

Define até 16 tabs verticais com o espaçamento entre linhas corrente. A definição de tabs não é afectada por subsequentes mudanças do espaçamento entre linhas. Os tabs são definidos pela ordem *n1 n2 ...*, todos com valores entre 1 e 255 com ordem crescente. O carácter NUL termina a definição. Todas estas definições afectarão só o canal 0. O comando ESC B NUL faz um *reset* de todas as definições.

ESC b *c n1 n2 ... NUL*27 98 *c n1 n2 ... 0*

Defina tabs verticais para um canal

Este comando funciona exactamente da mesma forma que o anterior (ESC B), salvo que o parâmetro *c* aqui indica o canal, que pode variar entre 0 e 7. Para fazer um *reset* deve-se enviar o comando ESC b *c* NUL.

ESC / *c*27 47 *c*

Selecciona um canal de tabs verticais

Este comando é utilizado para seleccionar o canal de tabs verticais, sendo este indicado pelo parâmetro *c* que pode ter valores entre 0 e 7. Depois deste comando todos subsequentes VT's usarão o canal aqui seleccionado.

MOVIMENTO HORIZONTAL

Margens

ESC | *n*27 108 *n*

Define a margem esquerda

Define a margem esquerda para *n* colunas da largura da fonte actual. Os valores máximos para *n* são: 79 para a fonte pica, 95 para a fonte elite, 133 para a fonte condensada e 156 para a fonte elite condensada. Este comando faz um *reset* a definições anteriores de tabs horizontais e limpa todos os caracteres na linha a imprimir.

ESC Q *n*27 81 *n*

Define a margem direita

Define a margem direita para *n* colunas da largura da fonte actual. Os valores máximos para *n* são: 79 para a fonte pica, 95 para a fonte elite, 136 para a fonte condensada e 159 para a fonte elite condensada. Este comando faz um *reset* a definições anteriores de tabs horizontais e limpa todos os caracteres na linha a imprimir.

Movimento da cabeça de impressão

BS

8

Recua a cabeça de impressão uma posição

Escreve os dados no *buffer* da impressora e recua a cabeça uma posição. Se a cabeça já estiver na margem esquerda, este comando é ignorado.

ESC \$ *n1 n2*27 36 *n1 n2*

Move para posição absoluta

Move a cabeça de impressão para *X* pontos a partir da margem esquerda. Podendo *X* ser calculado a partir da seguinte formula: $X = n1 + (n2 \times 256)$. Cada ponto representa 1/60 de polegada. Se o ponto pretendido ultrapassar a margem direita, este comando é ignorado.

ESC \ *n1 n2*27 92 *n1 n2*

Move para posição relativa

Move a cabeça de impressão para *X* pontos a partir da posição corrente. Podendo *X* ser calculado a partir da seguinte formula: $X = n1 + (n2 \times 256)$. Cada ponto representa 1/120 de polegada. Se o ponto pretendido ultrapassar a margem direita, este comando é ignorado.

Tab horizontais

HT

9

Tab horizontal

Avança a cabeça de impressão para o próximo tab horizontal. Cada tab equivale (por omissão) a 8 caracteres.

ESC D *n1 n1 ... NUL*27 68 *n1 n2 ... 0*

Define tabs horizontais

Este comando permite definir até 32 valores para tabs horizontais. Estes valores são representados por *n1 n2 ... 0*. O valor zero tem a finalidade de marcar o fim da lista de valores. Para restaurar os valores por omissão basta enviar o comando ESC D NUL.

PARA A PRÓXIMA

No próximo número será concluída a abordagem do modo EPSON, com alguns temas de grande importância: fontes, impressão de gráficos, caracteres definidos pelo utilizador, etc. Para finalizar será dado um exemplo em que serão demonstradas algumas das capacidades do modo EPSON.


 Carlos Ladeira

MOUSE «DEMO»

**COLABORAÇÃO
DO LEITOR**

Aqui temos um bom aproveitamento da teoria exposta na série «Roedores Informáticos», publicada nos números 15, 16 e 17 da *Spooler*. E ainda um bem elaborado conjunto de ensinamentos para quem pretender melhorar os seus programas com as funções do dito «roedor». No «compactado» da disquete também está incluído um ficheiro de texto (TMOUSE.DOC) a não perder. Passemos a palavra ao Hugo Ferreira.

Programar com o rato em QBasic, Pascal, C, *Assembly*, etc., ..., toda a gente sabe que se utilizam as estruturas para lidar com interrupções e faz-se uma chamada ao DOS (int 33h). Agora será que toda a gente sabe «domesticar» um rato em GWBasic?

Foi esta a razão que me levou a escrever este pequeno «demo» programado em GWBasic. São muitos os programadores iniciados que ainda programam em GWBasic. Então porque não se há de dar a *chance* de também estes poderem utilizar o rato nos seus programas?

O PROGRAMA

O «demo» está estruturado em subrotinas, o que leva a que ele tenha um desempenho muito lento (Encontra-se uma versão compilada juntamente com o código fonte). Embora o desempenho com tantas subrotinas se possa tornar um pouco mais lento, eu optei por assim fazer pois torna-se muito mais fácil compreender o programa.

O programa tem no início os dados técnicos dos computadores testados e onde o programa foi elaborado. Decidi incluir estas informações pois, se este programa é dedicado a um grupo de programadores ainda inexperientes, torna-se natural que, quando algo corre mal, seja o computador a «pagar as favas», ou então que se pense que algo correu mal porque isto não funciona. Estes dados ajudarão a comparar estas máquinas com as dos utilizadores, afim de se evitarem dúvidas.

O programa tem no seu código duas rotinas que se encontram no fim da listagem, as quais são responsáveis pela activação do rato e a chamada à «interrupção do rato».

A primeira é responsável pelas chamadas à interrupção a fim de utilizar os serviços do rato. No exemplo que se segue eu iria localizar o cursor na linha 100, coluna 150 (Píxeis):

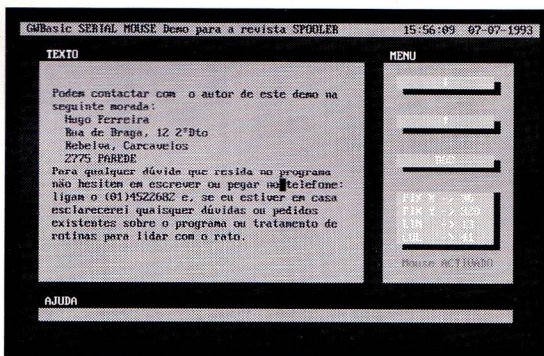
```
AX% = 4; CX% = 100; DX% = 150;
```

CALL MOUSE(AX%, BX%, CX%, DX%) – Esta chamada à interrupção do rato é que vai permitir utilizar os serviços do dito, sendo o seu equivalente em *assembly*:

```
MOV AX, 4
MOV CX, 100
MOV DX, 150
INT 33h
```

ainda em Pascal:

```
Var Regs: registers;
begin Regs.AX:= 4;
```



```
Regs.CX:= 100;
Regs.DX:= 150;
Intr($33, Regs);
end;
```

Como se pode ver, tanto em Pascal, como em *assembly* ou em GWBasic, lida-se mais ao menos da mesma maneira para fazer a chamada à interrupção. A maior discrepância é sem dúvida na maneira como as interrupções do rato são activadas. Enquanto nas outras linguagens o rato é logo activado com uma chamada ao serviço 00h, em GWBasic faz-se exactamente a mesma coisa mas, primeiro, temos que indicar ao GW onde se encontra a interrupção do rato, de maneira a que ele possa fazer a chamada do serviço 00h:

```
DEF SEG = 0
GMSEG = PEEK(51 * 4 + 2) + 256 * PEEK(51 * 4 + 3)
MOUSE = 2 + PEEK(51 * 4) + 256 * PEEK(51 * 4 + 1)
DEF SEG = GMSEG
```

São estas linhas é que vão permitir «activar» a interrupção, sendo o procedimento muito simples:

- 1 – Ajusta-se o segmento a 0 (segmento por omissão);
- 2 – em seguida acha-se o segmento da interrupção do rato (GMSEG);
- 3 – depois acha-se o *offset* da interrupção (MOUSE);
- 4 – finalmente, ajusta-se o segmento do rato como segmento base.

Como se pode concluir, é bastante simples operar com o rato pois, após estes passos terem sido feitos, o dito comporta-se como em qualquer outra linguagem, tendo sempre que se indicar o serviço pretendido em AX%.

Para uma listagem total dos serviços disponíveis, podem consultar os números 15, 16 e 17 da *Spooler*, que contém uma série de artigos sobre o assunto.

A restante parte do programa é composta por uma rotina principal que controla o programa, uma rotina para cada página de texto, e mais cinco que tratam do visual do programa.

O programa em si tem uma estrutura bastante simples e creio que é de fácil compreensão. O código fonte vai acompanhado por explicações a cada passo do programa. Creio que nada mais tenho a dizer acerca da programação e espero que gostem.

Hugo Ferreira

“
O programa tem no início os dados técnicos dos computadores testados e onde o programa foi elaborado.”

O FANTÁSTICO MUNDO DAS

BBS

PARTE II

Então, ainda não está registado na sua BBS preferida? Bom, vamos analisar os sistemas de registo mais comuns:

O REGISTO POR «CALL BACK & VERIFY»

Este sistema é bastante seguro para o Sysop e fácil de usar por parte do utilizador do sistema. Consiste no seguinte: o utilizador (vulgo *user*) que liga para a BBS pela primeira vez é levado pelo sistema, automaticamente, a um subsistema (chamado *door* – que veremos mais adiante) que lhe vai perguntar muito simplesmente qual o seu número de telefone. Atenção: o que o sistema pretende saber é exactamente o número do telefone onde nesse preciso momento está ligado o modem.

O *user* terá que introduzir esse número, incluindo o eventual indicativo se for fora da localidade da BBS, e mais nada. O sistema faz o seguinte: primeiro desliga a chamada; segundo, verifica se esse tal número não existe na actual base de dados do sistema; finalmente, liga para esse número. Aqui está um pequeno truque: o *user* (se tudo correr bem) vai ver a palavra **RING** no seu monitor: tudo o que tem de fazer é escrever a palavra **ATA**, depois disso recebe

uma mensagem do sistema a pedir para introduzir a sua *password*: faça-o. Em seguida é levado ao sistema com o nível de utilizador registado.

Pode haver várias razões para que este sistema não funcione. Uma delas, a mais evidente, é que esse número já tenha sido verificado, o que só acontece se o *user* tenta registar-se mais do que uma vez com nomes diferentes. Como isso corresponde de facto a uma pequena fraude, não vamos considerar essa hipótese.

Outra razão para não funcionar é quando o utilizador vê a palavra **RING** no monitor, levanta o auscultador do telefone e tenta responder à chamada: isso faz cair a própria chamada e invalida o processo.

Outra razão é o modem do utilizador não «perceber» os comandos, nomeadamente o comando **ATA** que deveria colocá-lo de novo em contacto com a BBS. Essa hipótese é muito mais frequente do que se pensa pois, infelizmente, ainda existem muitos modems (nem sempre os mais baratos) que não são compatíveis com os códigos internacionais de comunicações.

Por todos esses motivos, existem várias possibilidades deste sistema não funcionar.

A primeira reacção do utilizador é tentar de novo. Essa

é, sem dúvida, uma decisão inteligente que recomendamos porque, antes de desistir, é sempre melhor experimentar pelo menos mais uma ou duas vezes... Se ainda assim não funcionar, então terá várias hipóteses:

O REGISTO POR CARTA (OU FAX)

O Sysop está bem consciente que o *call back* pode não funcionar logo à primeira e, normalmente, recebe sempre uma série de mensagens do tipo: «não consigo registar-me, o meu modem não atende» ou ainda «estou a ligar duma central telefónica e a chamada não passa» ou ainda «o meu modem deve estar mal configurado pois não há maneira de estabelecer a ligação», etc., etc. É natural que o *user* fique frustrado, mas não deve! Um Sysop inteligente e cuidadoso terá seguramente previsto uma ou mais alternativas.

Por exemplo, a coisa mais simples desse mundo é o registo via correio ou via FAX.

Durante a sua primeira ligação, o novo *user* vai ver passar à sua frente uma série de mensagens, boletins, notícias que, em geral, ele nem lê. Mal! Muito mal! Primeiro, porque o Sysop teve o trabalho de as escrever só para si e, portanto, por uma razão de respeito e educação deve ler isso tudo; segundo, porque 99,99% da vezes são essas informações que lhe permitirão registar-se e tirar o máximo partido da BBS no futuro.

É caricato, mas frequente, que as primeiras mensagens de um novo utilizador se refiram a dúvidas que quase sempre podem ser respondidas apenas lendo de novo as informações introductórias. Portanto leia bem tudo, porque é natural que nas notícias iniciais esteja devidamente indicado como fazer para registar-se pelo correio (ou FAX).

O processo é muito simples. Na maioria dos casos basta enviar um postal, ou uma pequena carta com os dados pessoais (eventualmente uma fotocópia do B.I.) e pronto. Saiba desde já que essas informações são absolutamente «sagradas» para o Sysop, que toma o compromisso de nunca as divulgar: por isso pode ficar descansado.

Logo que o Sysop receba essa sua informação irá proceder ao seu «upgrade», ou seja, ao aumento de nível, o que lhe permitirá, quase sempre, utilizar as potencialidades da BBS em cheio.

UM POUCO DE ÉTICA!

Ok, agora está registado, faz parte de um clube ou de uma «micro-sociedade», e como tal existem regras de comportamento que são apenas ditadas pelo senso comum, pela educação e pelo sentido de civismo de qualquer ambiente.

O QUE NUNCA DEVE FAZER:

- Nunca utilize uma linguagem agressiva nem para o Sysop nem para os outros utilizadores da BBS. Se tiver divergências ou opiniões diferentes dos outros (o que é o mais natural) use sempre educação e cortesia, pois só tem a ganhar com isso.
- Nunca espere que o Sysop saiba tudo sobre toda a matéria informática. O Sysop é uma pessoa como outra qualquer. Talvez tenha mais conhecimentos de telemática e até pode ser um *expert*, mas não assumo isso como facto consumado, pois existem outros utilizadores que, muito provavelmente, sabem muito mais em assuntos específicos como programação, *hardware* etc.
- Nunca insista para falar por «voz» com o Sysop, pois quase sempre a BBS é para ele um *hobby* e não um

trabalho (ainda que lhe ocupe imensas horas do dia e da noite). Sempre que o desejar pode deixar um comentário para o Sysop ou uma mensagem para este ou outros utilizadores. Pode ficar descansado que o Sysop vai SEMPRE responder à sua mensagem, nem que seja para o orientar.

- Nunca tente julgar o comportamento dos outros utilizadores escrevendo mensagens de moderação. Existe um Sysop que toma conta desses assuntos. O que poderá fazer eventualmente é chamar a atenção do Sysop para situações que lhe possam parecer menos «canónicas».
- Nunca tente ser o «Chico esperto» do seu bairro ao tentar mais do que um registo na BBS. O Sysop não nasceu ontem e apercebe-se disso, o que o levará a invalidar o seu actual nível de acesso.
- Nunca espere encontrar na BBS programas comerciais. Isso seria uma violação do direito de *copyright* e um Sysop minimamente inteligente nunca terá o «Wordstar», o «Windows» ou o «Clipper» na sua lista de ficheiros (vulgo *files*). Assim, pelo mesmo raciocínio, nunca deverá enviar para a BBS programas que não sejam *shareware*, *public domain* ou *freeware*. Veremos noutra secção o que isto significa.

O QUE DEVE SEMPRE FAZER

- Respeitar a opinião dos outros, mesmo que não concorde. O diálogo fez-se para essas situações. Aliás, uma das áreas mais interessantes da BBS é justamente a área de mensagens onde todos podem trocar opiniões e pareceres sobre os assuntos mais diversos.
- Respeitar o nível que lhe é atribuído: isto quer dizer o quê? Bom, em princípio um nível de acesso consiste num tempo limite e num número de Kbytes que pode «puxar» (chama-se *download*) por dia. Existem meios para incrementar esses limites, os quais são explicados nas regras da BBS e o Sysop tem a obrigação de os pôr em boa evidência aos seus utilizadores.
- Ler as notícias que o Sysop escreveu para si (e para os outros). Essa tarefa exige um empenhamento por parte do Sysop e ele escreve isso para si, para que esteja sempre a par das novidades e eventuais mudanças de rumo ou de configuração da BBS. Você deve ler tudo isso se quiser participar de uma maneira activa na vida da BBS.
- Ler todas as mensagens que lhe são dirigidas, não só pelo Sysop, mas também pelos outros utilizadores. Se o seu único interesse é o de ligar para a BBS para puxar *files*, bem, isso é considerado uma atitude pouco interessante e, apesar de estar no seu pleno direito de o fazer, nunca lhe será reconhecido o atributo de utilizador de pleno direito.

Haveria muito mais a dizer sobre esta matéria, mas também não quero assustar ninguém com demasiadas regras e deveres: afinal a BBS é um *hobby* e um prazer, pelo que as coisas vêm com o tempo e estes conceitos que eu expriimi resultam só do bom senso e da prática.

No próximo capítulo vamos começar a analisar os comandos do MENU PRINCIPAL. Iremos vê-los um por um e ainda que a referência seja feita a um programa específico tal como o PCBoard(tm), na esmagadora maioria das BBS os comandos e as possibilidades são equivalentes.

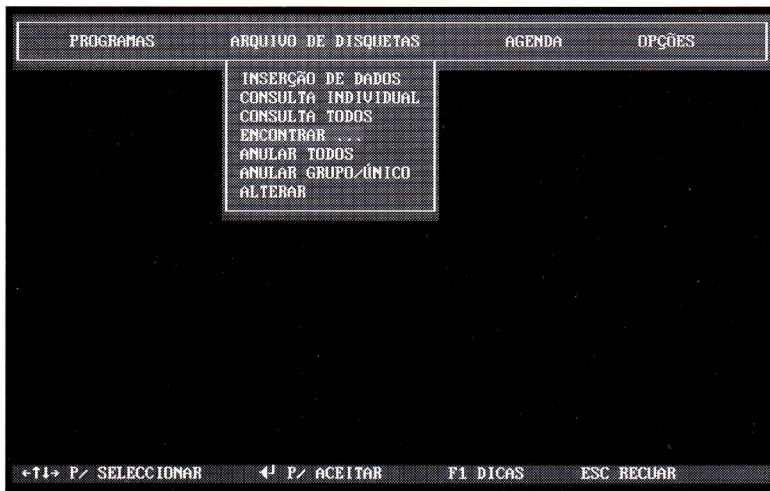
Não se esqueça de ler o próximo artigo sobre o «FANTÁSTICO MUNDO DAS BBS»!



Mário Pozzetti, o seu amigo Sysop.

BDI

BASE DE DADOS INDISPENSÁVEL



Este programa já «deambulava» há alguns meses na nossa redacção, sempre objecto de um profundo diálogo com o autor. A versão agora oferecida aos nossos leitores é o resultado desse diálogo e dos meritórios esforços do José Tomás no sentido do aperfeiçoamento. Considerámo-lo de indiscutível interesse, sobretudo pelo carácter didático que a sua construção encerra. Convidamos os entusiastas do QBasic a uma cuidada análise das fontes incluídas na disquete.

UM POUCO DE HISTÓRIA

Já lá vão alguns anos desde que adquiri um PC AT 286. Desde então tenho acumulado *software* em disquetes. Passei a ter problemas quando precisava de encontrar um programa ou uma disquete, ou até mesmo quando queria saber quantas disquetes tinha.

Decidi numerar as disquetes e desenvolver uma base de dados em dBase, mas o resultado não foi o desejado. Logo me decidi a desenvolver uma Base de Dados sobre disquetes. Mais tarde acabei por desenvolver o BDI, que reúne três «Bases de Dados Indispensáveis» para quem trabalha com PC e tem por hábito reunir bastantes disquetes com *software*.

Como a *Spooler* é uma excelente revista portuguesa de informática, eu dedico o BDI à *Spooler* no intuito de torná-lo num programa de domínio público.

CONSTITUIÇÃO DO BDI

O BDI foi feito em Quick Basic 4.5, que permite utilizar estruturas optimizadoras de funcionamento e rapidez, servindo-se de uma linguagem tão fácil de entender como é o

BASIC. O BDI é constituído pelas seguintes bases de dados:

- **Arquivo de disquetes.** É uma base de dados constituída por campos referentes às disquetes que se possui em arquivo.
- **Programas.** É uma base de dados constituída por campos referentes a programas para uma melhor gestão de *software*.
- **Agenda.** É uma base de dados elaborada de modo a armazenar telefones, moradas e algumas referências sobre pessoas que conhecemos.

O BDI é constituído pelos seguintes ficheiros indispensáveis:

- BDI.EXE – Ficheiro principal
- BDIAG.CHN – Ficheiro secção da Agenda
- BDIAR.CHN – Ficheiro secção do Arquivo de Disquetes
- BDIP.CHN – Ficheiro secção dos Programas

Todos os outros ficheiros além do BRUN45.EXE (ver Compilação), são ficheiros criados após funcionamento do BDI:

- BDCONFIG.DAT – Possui dados sobre a apresentação configurada.
- BDISHELL.DAT * – Ficheiro de comunicação entre os 4 ficheiros executáveis (ver notas)
- AGENDA[.DAT;.OLD;.BAK;.IND;.REF] ** – Possuem dados da secção da Agenda.
- ARQDISK[.DAT;.OLD;.BAK;.IND;.REF] ** – Possuem dados da secção do Arquivo de Disquetes.
- PROGRAMS[.DAT;.OLD;.BAK;.IND;.REF] ** – Possuem dados da secção de Programas.

Notas: O ficheiro assinalado com * nunca deverá surgir. Os ficheiros assinalados com ** nunca deverão surgir (ver Funcionamento) com algumas das extensões possíveis.

Cada base de dados é formada por um ficheiro executável, embora esses ficheiros tenham funcionamento dependente do ficheiro principal (BDI.EXE).

Os ficheiros de extensão CHN (Extensão CHN porque são chamados a executar tarefas por meio da instrução CHAIN do Quick Basic.), são executáveis embora quando corridos com extensão EXE aparentem não funcionar, pois não tiveram acesso ao ficheiro de comunicação BDISHELL.DAT.

FUNCIONAMENTO DO BDI

O BDI tem uma apresentação comum aos programas da Borland, onde as onze primeiras linhas se destinam a

pequenos menus. A linha 25 do ecrã informa quais as teclas e para que servem, para que o utilizador se possa orientar. As linhas intermédias às linhas 11 e 25 destinam-se a diversos menus de comandos ou serviços.

BDI.EXE

O ficheiro BDI.EXE propõe 4 quatro serviços diferentes. O 1º serviço promove um acesso cómodo às três bases de dados e ao conjunto de três outros pequenos serviços (OPÇÕES). Ao promover o acesso a qualquer uma das bases de dados cria o ficheiro BDISHELL.DAT, onde deposita os valores de variáveis necessárias ao bom funcionamento de cada base de dados. Por sua vez, as bases de dados passam o controle ao ficheiro principal (BDI.EXE), recriando o ficheiro BDISHELL.DAT com valores de variáveis que dizem ao BDI se fica no menu principal ou se passa para um dos Pull-down vizinho.

O 2º serviço – Configuração de cores – faz parte de um pequeno conjunto de três serviços. Este serviço propõe três definições possíveis para a apresentação do BDI, sendo possível testar a definição activa accionando a tecla F6 e gravar a definição activada accionando a tecla F2.

Atenção: caso não se grave a definição activada, o BDI na sua próxima chamada irá correr com a definição anterior. Por omissão o BDI arranca com a apresentação Vermelho Azul.

O 3º serviço – Sair – permite sair para o DOS e retornar digitando EXIT seguido de <ENTER>.

O 4º serviço – Comandos do DOS – permite digitar comandos sem que se tenha de sair do programa. A tecla F3 neste serviço permite ter acesso a um menu de dicas, a tecla F4 permite recuar ao Pull-down das opções, e a tecla F5 permite repetir o último comando de DOS executado.

BDIAG.CHN & BDIP.CHN

Os ficheiros BDIAG.CHN e BDIP.CHN formam as bases de dados Agenda e Programas respectivamente, possuindo serviços idênticos. Estes ficheiros promovem oito serviços diferentes. O 1º serviço – Inserção de dados – como o próprio nome diz, promove a inserção de dados em campos pré-defenidos pelo programa, com um conjunto máximo de 999 registos (cada registo é um grupo de dados/campos).

Quando se acciona este serviço, é automaticamente dado um número a seguir ao último registo, para o que bastará premir <ENTER>, mas se quiser pode mudar o número ou até mesmo apagá-lo e depois premir <ENTER> para sair deste serviço. No fim da inserção de dados poderá escolher entre o ENTER, para fazer entrar os dados e voltar a inserir mais, e o ESC para fazer entrar os dados e sair deste serviço.

O 2º serviço – Consulta individual – permite consultar registo a registo na integra. Quando se acciona este serviço, automaticamente ele pede o número do registo a consultar; caso se prima <ENTER> sem se referir qualquer número, sai-se deste serviço. Caso contrario é exibido o registo com todos os seus dados possíveis. No fim de cada exibição poderá decidir entre o ENTER para uma nova consulta e o ESC para sair deste serviço.

O 3º serviço – Consulta todos – permite consultar todos os registos embora só exiba o número de registo e o(s) campo(s) principal(ais) ([Nome Telefone]/[Programa]). No fim de cada página é pedido que se prima qualquer tecla para continuar, ou sair. Este serviço pode ser dirigido a duas saídas possíveis (seleccionáveis no início), que são impressora (I) ou ecrã (E).

O 4º serviço – Encontrar – quando se acciona este serviço é-nos pedida automaticamente uma cadeia de caracteres que será procurada no campo principal independentemente das letras serem maiúsculas ou minúsculas. Logo

de seguida são exibidos os registos que contêm essa cadeia de caracteres. Depois é pedido que se prima uma tecla para ver a próxima página se existir, caso contrário será para sair deste serviço.

O 5º serviço – Anular todos – quando se aciona este serviço o ficheiro que armazena os dados é automaticamente renomeado para extensão OLD e assim permanecerá, para o caso de alguém chamar este serviço por engano; neste caso bastará inverter o processo via DOS renomeando o mesmo ficheiro para extensão DAT. Como o ficheiro armazém deixa de existir com a extensão DAT, o programa reconhece uma anulação de dados.

O 6º serviço – Anular Grupo/Único – este serviço, como o próprio nome sugere, serve para anular um grupo de registos ou um só registo; para isso é necessário definir o primeiro e o último registo a anular (Grupo), ou qual o número do registo a anular (Único). De seguida é pedido para confirmar a definição; se confirmar é feita a anulação e sai-se do serviço; se anular sai de imediato deste serviço.

O 7º serviço – Alterar – é similar ao 1º serviço, embora seja necessário dizer qual o registo a alterar e confirmar de seguida a sua alteração; caso se confirme a alteração passa-se à inserção dos novos dados. No fim da inserção temos duas hipóteses: o ENTER para fazer entrar os dados e voltar ao serviço de alteração ou o ESC para fazer entrar os novos dados e sair.

O 8º serviço – Ordenar – é bastante simples, pois a sua função limita-se a ordenar os registos alfabética e numericamente.

Durante a ordenação este serviço cria e apaga o arquivo que armazena os dados com quatro extensões possíveis. Cria e recria o ficheiro com extensão REF que serve de referência para o SORT do DOS e posteriormente para o próprio serviço de ordenação e cria o ficheiro com extensão IND para compatibilizar o ficheiro de extensão REF com o ficheiro que sairá de extensão DAT já ordenado. Em todo este processo, um tanto complicado, sobra o ficheiro de extensão DAT já ordenado e todos os outros ficheiros são simplesmente apagados.

BDIAR.CHN

O ficheiro BDIAR.CHN forma a base de dados do Arquivo de Disquetas, prestando serviços bastante semelhante aos ficheiros anteriormente discriminados.

Nota 2: Na base de dados do Arquivo de Disquetas quando se fala em anular registos significa anular os dados num dado registo ou grupo que corresponde a uma disquete ou a um grupo de disquetas que existe(em) mas ou está(ão) vazia(s) ou estragada(s), pois seria ilógico só por causa de uma anulação, ter de se mudar o número a uma quantidade enorme de disquetas.

FONTES E SUA COMPILAÇÃO

Como se pode ver, o BDI é um gestor de dados que proporciona diversos serviços para uma melhor gestão desses mesmos dados, sendo esses serviços quase iguais para as três bases de dados.

Atenção: durante este capítulo que nos fala das fontes, toma-se como referência a linguagem do Qbasic do DOS 5 para uma facilitação de consulta. Para este programa tanto faz falar de Qbasic ou Quick Basic, pois não se utilizou qualquer função ou instrução incompatível entre os dois ambientes.

As fontes, como não poderia deixar de ser, tem extensão BAS, tendo sido compiladas com o Quick Basic Compiler 4.5. Para que as fontes funcionem em ambiente Qbasic é necessário copiar as fontes para as extensões que têm quando compiladas, pois são chamadas pela extensão CHAIN do Qbasic que não olha às extensões classificativas, mas

“

Como se pode ver, o BDI é um gestor de dados que proporciona diversos serviços para uma melhor gestão desses mesmos dados.

”

sim ao ambiente em que se está a trabalhar e à especificação do ficheiro, quer em ambiente Qbasic quer DOS. A instrução CHAIN transfere apenas o controlo do programa actual para um outro programa Qbasic ou executável, dependendo apenas do ambiente em que se está a trabalhar.

Em Qbasic a instrução CHAIN transfere o valor de todas as variáveis consideradas globais pela instrução COMMON SHARED, mas isto já não é possível em ambiente DOS, daí a existência de um ficheiro de comunicação, o BDISHELL.DAT, durante o funcionamento do BDI.

Muitos serão os que perguntam porque é que se não utilizou a tão habitual instrução SHELL. A resposta é simples: o Quick Basic Compiler 4.5 tem hipótese de criar ficheiros executáveis com pouco mais de 100K bytes mas, a partir daí, nega-se com uma falsa mensagem de erro pois os habituais 580 Kb que temos livres na memória baixa não lhe chega. E neste programa debati-me com o problema da pouca memória do meu 286. Ora bem, se eu tenho esses problemas, na certa que não sou o único; daí ter optado pela instrução CHAIN que não acumula nada em memória do programa anterior, apenas transferindo o controlo ao próximo programa, enquanto que o SHELL, digamos que «empresta» o controlo e acumula informação na memória.

Além disso, a instrução CHAIN permite ter acesso a ficheiros executáveis que ponho com extensão diferente da habitual extensão EXE, pois embora os ficheiros de extensão CHN (BDIAG; BDIAR; BDIP) sejam executáveis independentes, dependem na realidade de determinados valores para variáveis que utilizam para poderem funcionar

com a configuração de cores pré-definida. Daí a extensão CHN, pois caso contrário todos correriam os ficheiros que não vale a pena correr via DOS.

Na compilação, o Quick Basic Compiler 4.5 oferece duas hipóteses: a primeira que tem menor dimensão consiste na compilação das fontes que irão correr em conjunto com o BRUN45.EXE que deve estar presente; a segunda tem maior dimensão pois dispensa o BRUN45.EXE.

Sempre que se fizer a compilação das fontes deve-se sempre proceder à renomeação dos ficheiros de extensão EXE para CHN, tal como expliquei acima.

ADVERTÊNCIAS

- Quando utilizar o serviço «Comandos do DOS», tenha o cuidado de sair dele estando na drive e directoria em que instalou o BDI.
- A mensagem de erro «NÚMERO IVÁLIDO!» é perfeitamente desprezável quando se quer sair de um serviço não digitando qualquer número e premindo <ENTER>.
- Chama-se a atenção para o facto de ambos os ficheiros estarem compilados em formato que requiere sempre o BRUN45.EXE (Versão distribuída pela Spooler.).

 José F. Tomás

CIFRA & BCIFRA

Aqui temos mais um programa dedicado à segurança dos nossos dados, um tema de grande actualidade que está a levantar alguma polémica em certos meios. Mas enquanto os outros discutem «o sexo dos anjos», vejamos o que o Carlos Batista tem a dizer sobre o assunto:

Estes dois programas constituem exemplos de dois métodos de criptação. Um utilizado desde os primórdios da criptação e outro aparecido com a ciência da Informática. O CIFRA utiliza o método mais conhecido (e simples...) da substituição, ou seja, a substituição de um carácter ASCII por um outro, sempre na mesma sequência. Seria como se tivéssemos um alfabeto todo trocado: em vez de «abcde-fg...», teríamos «qwyurg...», onde o «a» corresponderia a «q» e assim sucessivamente.

O BCIFRA serve-se de um método exclusivamente informático: a manipulação dos bit de um carácter. Neste caso, substitui-se o carácter ASCII pelo seu complementar (símbolo ~). No entanto, poder-se-iam manipular os bit de uma forma muito mais complexa.

Os programas CIFRA e BCIFRA são sobretudo «académicos» pois, para além dos sistemas de criptação serem simples, um ficheiro codificado com o BCIFRA poderá ser descodificado por qualquer pessoa, desde que possua o BCIFRA!

Pretendem mostrar uma faceta curiosa e interessante da Informática, muitas vezes utilizada pelas Forças Armadas de muitos países: a Criptação. Alterando um pouco as

fontes torna-se possível, inclusivamente, construir um programa próprio de criptação que só o leitor possa usar. A linguagem utilizada para estas rotinas foi (na minha opinião) a mais apropriada para a elaboração deste tipo de programas – a C – por várias razões: por ser extremamente portátil entre computadores, protegendo o trabalho de desenvolvimento de rotinas próprias e, no caso do BCIFRA, por ser a única linguagem de médio nível capaz de manipular os bit com eficácia.

FUNCIONAMENTO

A sintaxe é muito simples:

BCIFRA (ou CIFRA) <ficheiro> codificar (des) <ficheiro> codificar (des) C I D

C e D são os parâmetros que indicam se queremos Codificar ou Descodificar o ficheiro. Por exemplo:

CIFRA codigo.txt texto.txt D

descodifica o ficheiro CODIGO.TXT para o ficheiro TEXTO.TXT.

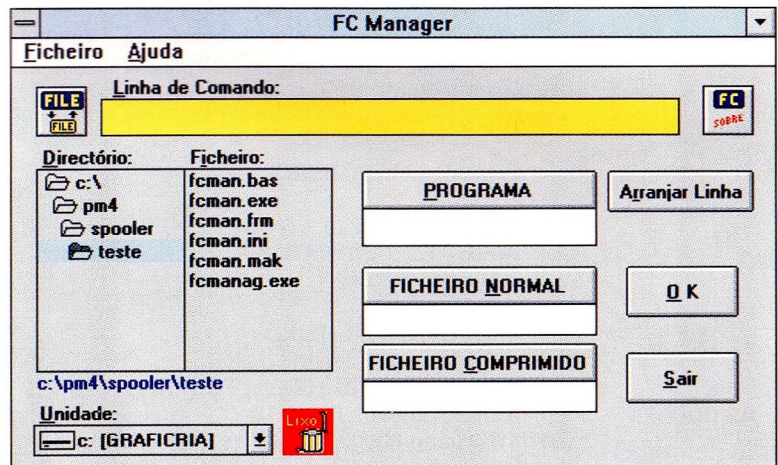
Por último um alerta: os ficheiros codificados pelo CIFRA não podem ser descodificados pelo BCIFRA e vice-versa.

 Carlos Batista

FC MANAGER

**COLABORAÇÃO
DO LEITOR**

Pedro Sinogas não quer que os leitores da Spooler se vejam confrontados com as mesmas dificuldades por que passou, ao ter adoptado a plataforma Windows. Eis um excelente utilitário com a não menos excelente apresentação facultada pelo ambiente Windows. Conforme é dito pelo autor, o programa necessita da presença, no directório Windows, do ficheiro VBRUN100.DLL. Tal ficheiro, pela sua dimensão, não é incluído na disquete deste número. Os nossos assinantes e leitores fiéis já o receberam na disquete da Spooler N.º 18 com o programa WINMENU. Os leitores que não possuírem este número da revista poderão passar pelas nossas instalações e obtê-lo, ou escrever-nos enviando uma disquete de 3 1/2": recebê-la-hão de volta com o dito ficheiro.



Trata-se de um pequeno utilitário para Windows que ajuda o utilizador a manusear os incómodos compressores de DOS e outros programas, cuja utilização de dentro do Windows requiera um «Shell» para o DOS.

HISTÓRIA E PORQUÊ DO FCMAN

É frequente a utilização que faço de compressores de ficheiros, como o PKZIP/PKUNZIP, especialmente para poupar espaço em disquetas.

Quando adoptei a plataforma Windows, vi-me confrontado com a necessidade de utilizar o DOS PROMPT e/ou o FILE/RUN para proceder à compressão dos ficheiros de que necessitava.

A ideia deste utilitário surgiu, portanto, para me facilitar esta tarefa que obrigava a escrever não só as coordenadas do compressor, como as dos ficheiros comprimidos e a comprimir, assim como as opções pretendidas. É um processo relativamente incómodo e sujeito a erro. «Pegar» nos ficheiros com o mesmo rato com que se manipula o Windows e «despejá-los» numa caixa que faça o «serviço», é bastante mais cómodo e evita erros de escrita.

Comecei então por fazer um pequeno programa em Visual Basic que me respondesse às necessidades apontadas. Adicionei-lhe depois outras possibilidades, tal como a hipótese de automaticamente escolher as opções do compressor e mesmo os nomes dos ficheiros que rotineiramente comprimo sempre da mesma forma.

O seu nome – FCMAN – surgiu da redacção inicial em inglês (File Compression Manager).

Quando surgiu a ideia de o publicar através da Spooler, houve a necessidade de o traduzir para português, juntar instruções, opções para manipulação de ficheiros e «alinhá-lo» esteticamente. O resultado é o que se apresenta.

DE QUE NECESSITA O FCMAN PARA CORRER?

- Microsoft Windows 3.00 ou posterior, como plataforma base;

- FCMAN.EXE (Programa propriamente dito);
- FCMAN.INI (Ficheiro com as opções para cada programa predefinido, não obrigatório);
- VBRUN100.DLL (Run-Time Library do Visual Basic, sobre o qual foi construído).

Durante a sua execução, o programa cria um ficheiro temporário ~FCMAN.BAT, de que vos falei mais a frente. Relativamente ao computador, já correu em 286, 386 SX e 386 DX. Também já provou em CGA monocromático e em diversas resoluções de VGA.

O QUE FAZ O FCMAN ?

O FCMAN, a mando do utilizador e de forma que julgo muito simples, executa todo o tipo de programas para DOS a partir do Windows. Terminada a execução dos programas que lança, pára com um PAUSE do DOS, para permitir ao utilizador observar o resultado da sua execução, antes que o Windows retome o controlo das operações e impeça a leitura de quaisquer mensagens e/ou erros reportados no fim pelo *software*.

O Ficheiro FCMAN.INI contém a indicação dos programas definidos e as respectivas opções que se utilizam com mais frequência. Pode ser alterado com qualquer editor de texto ASCII (NOTEPAD do Windows, por exemplo), de acordo com as instruções que se indicam adiante.

Neste utilitário, quando com o programa a executar é utilizado um ficheiro com opções pré-definidas no FCMAN.INI, aparece o item OPÇÕES na MENU BAR, disponibilizando as alternativas previamente definidas pelo utilizador e utilizáveis sem qualquer hipótese de erro, se bem definidas. Se o utilizador necessitar de opções que não se encontrem nesse menu, pode inseri-las na Linha de Comando do FCMAN ou alterar o FCMAN.INI, para as acrescentar (ver Ajuda/Como...).

O FCMAN como programa para Windows que é, pode funcionar integralmente com o rato, bem como com as teclas, como a maioria dos programas existentes para esta plataforma gráfica.

COMO O FAZ O FCMAN?

O FCMAN recolhe todas as informações dadas pelo utilizador e, quando este clica no botão OK, cria automaticamente um «Batch File» temporário: ~FCMAN.BAT em que estão contidas todas essas informações. Faz depois um «shell» para o DOS e executa o ~FCMAN.BAT, a partir da actual Unidade e Directório.

COMO SE FAZ COM O FCMAN ?

As dicas que se alinham a seguir também podem ser consultadas dentro do programa escolhendo AJUDA/COMO...

– COPIAR UM FICHEIRO

Selecione o Ficheiro, escolha FICHEIRO/COPIAR FICHEIRO SELECCIONADO e introduza o novo nome com o caminho pretendido.

– APAGAR UM FICHEIRO

Selecione o ficheiro e escolha FICHEIRO/APAGAR FICHEIRO SELECCIONADO ou selecione o ficheiro e DoubleClick no CAIXOTE DO LIXO ou selecione o ficheiro e arraste-o para o CAIXOTE DO LIXO. Por medida de segurança, exige-lhe a confirmação.

– MOVER/RENOMEAR UM FICHEIRO

Selecione o ficheiro e escolha FICHEIRO/RENOMEAR FICHEIRO SELECCIONADO, digite o novo nome e caminho do ficheiro (Se indicar a mesma unidade de disco com diferente directório, far-se-á a mudança do ficheiro).

– ACTUALIZAR A LISTA DE FICHEIROS

Porque o seu funcionamento é através da execução de programas DOS, a lista de ficheiros não é actualizada automaticamente após a compressão ou descompressão de ficheiros. A sua actualização faz-se sempre que muda de directório ou escolhendo FICHEIRO / REFRESCAR LISTA DE FICHEIROS.

– SELECCIONAR TIPO DE FICHEIROS

Escolha FICHEIRO/TIPO DE FICHEIROS e responda digitando as especificações do tipo de ficheiros a ver. Por exemplo: *.ZIP ou CARTA?.DOC

Aparecerão na lista de ficheiros apenas os que obedecem aos requisitos definidos.

– MUDAR DE UNIDADE DE DISCO

Escolha FICHEIRO/MUDAR DE UNIDADE DE DISCO e digite a primeira letra da unidade ou abra a lista de unidades com o rato e escolha a unidade desejada ou selecione a unidade com as teclas de cursor.

– MUDAR DE DIRECTÓRIO

Selecione o directório da lista de directórios com as setas ou com o rato, prima <ENTER> ou faça um DoubleClick com o rato.

– ESCOLHER O PROGRAMA

Selecione o programa da lista de ficheiros e arraste-o para o local do programa com o rato ou prima <ALT+P>, escolha o programa a correr e prima <ENTER> ou DoubleClick com o rato. Não é aceitável digitar o nome do

programa no respectivo quadro, para garantia de que só serão lançados programas que de facto existam nos discos activos.

Sempre que é seleccionado um ficheiro e o rato é mantido em baixo, o cursor transforma-se numa mão a «pegar» numa folha nos locais onde é permitido largar o ficheiro seleccionado e num NÃO onde não serve de nada largá-lo.

– ESCOLHER AS OPÇÕES DO PROGRAMA

Escolha OPÇÕES/OPÇÕES, escolha o item e prima <ENTER> ou carregue com o rato (o Menu das OPÇÕES só aparece para os programas definidos no ficheiro FCMAN.INI e são inseridas imediatamente a seguir ao nome do programa). Se necessário eliminar as opções activas, escolher OPÇÕES/OPÇÕES e LIMPAR TODAS AS OPÇÕES ou prima <CTRL+C>.

– COSTUMIZAR AS OPÇÕES

As opções são definidas pelo utilizador de acordo com as suas necessidades habituais, podendo mesmo incluir-se uma Linha de Comando com todos os elementos necessários a passar para o programa definido: ver exemplo prático a seguir. Cada opção ocupa uma linha do ficheiro FCMAN.INI imediatamente a seguir ao nome do programa respectivo (entre dois espaços e dois parênteses rectos) e antes de um sinal de =, após o que facultativamente se pode definir o significado da opção. O ficheiro FCMAN.INI será automaticamente carregado se existir no mesmo directório do FCMAN.EXE, aquando do lançamento deste.

– MODIFICAR A LINHA DE COMANDO

Escreva directamente na LINHA DE COMANDO, prima <ENTER> duas vezes ou carregue no OK. (Não carregue no ARRANJAR LINHA pois este serve para programas com opções definidas e vai buscar os elementos existentes nas respectivas caixas).

– COMPRIMIR UM FICHEIRO

Defina o programa a usar, arraste o ficheiro a comprimir para o local do FICHEIRO NORMAL com o rato ou prima <Alt+N>, <Alt+I>; escolha o ficheiro e prima <ENTER> ou «DoubleClick» com o rato, escolha o directório e/ou nome do novo ficheiro (comprimido), arrastando-o ou escrevendo-o no local do FICHEIRO COMPRIMIDO. Adicione as opções. Carregue no ARRANJAR LINHA e no OK.

– DESCOMPRIMIR UM FICHEIRO

Defina o programa a utilizar, arraste o ficheiro para local do FICHEIRO COMPRIMIDO com o rato ou prima <Alt+C>, <Alt+I>, escolha o ficheiro e prima <ENTER> ou DoubleClick com o rato. Escolha o directório para onde pretende extrair o(s) ficheiro(s), arrastando-o para o local do FICHEIRO NORMAL. Adicione as opções. Carregue no ARRANJAR LINHA e no OK.

COMO EU FAÇO COM O FCMAN !

Vamos supor que queremos comprimir todos os ficheiros de gráficos criados com o PAINTBRUSH no formato BMP em determinado directório, para fazer seguranças em disquetes. A linha de comando será qualquer coisa como:

```
C:\PKWARE\PKZIP -a -ex -& A:IMAGENS.ZIP
C:\FILES\PAINT\*.BMP
```

Em ambiente DOS, o que há a fazer é digitar uma linha

“
As opções
são definidas
pelo utilizador
de acordo
com as suas
necessidades
habituais,
podendo
mesmo
incluir-se uma
Linha de
Comando com
todos os
elementos
necessários a
passar para o
programa
definido.
”

de comando, exactamente como se indica. Em Windows, pode escolher-se FILE / RUN... e digitar os mesmos 60 caracteres.

Com o FCMAN deve seleccionar-se o ficheiro PKZIP.EXE do respectivo directório e arrastá-lo para o local do programa, arrastar um dos ficheiros BPM do directório, ou o próprio directório que se pretende comprimir, para a caixa do ficheiro normal e substituir aí o nome desse ficheiro por *.BMP. Arrastar a unidade de disquetes A: para a caixa do ficheiro comprimido e acrescentar o nome que se lhe pretenda dar. Escolher as opções indicadas e clicar OK. Em alternativa pode acrescentar no FCMAN.INI a linha de comando que se apresenta, sem o nome do programa, na área do [PKZIP.EXE], seguido do sinal =. Sempre que se pretenda repetir esta operação há apenas que arrastar o PKZIP para a caixa do programa, escolher esta opção e clicar OK.

Foi-me sugerido que considerasse a possibilidade de poder incluir ficheiros sortidos no momento da compressão ou da descompressão. De acordo com a filosofia do progra-

ma, cada ficheiro seleccionado é definido pelo seu nome, antecedido do respectivo «path» completo. Ao tentar contemplar esta sugestão, apercebi-me de que a linha de comando do DOS, que só suporta até 127 caracteres, era rapidamente saturada quando se seleccionavam vários ficheiros de subdirectórios com vários níveis para outros subdirectórios. Assim, optei por não contemplar esta possibilidade. Na minha utilização regular deste utilitário carrego, com o rato, um ficheiro de cada vez, ou utilizo *Wildcards*, suportados por todos os compressores, conforme ao exemplo apresentado no tópico «Ajuda/Como Seleccionar Tipo de Ficheiros».

Espero que o FCMAN seja útil e corresponda às expectativas que lhe criei, caro leitor. Qualquer dúvida ou sugestão será sempre bem-vinda. Não hesite em escrever para a **Spooler** ou para a morada indicada no programa.



Pedro Sinogas

CÁLCULOS MATEMÁTICOS

Paulo Caridade entrou pelas «matemáticas» e oferece-nos um «pacote» destinado a facilitar cálculos complicados. Eis o seu discurso:

Este programa permite aproximar uma lista de **m** pontos, por um polinómio de grau **n**, pelo método dos mínimos quadrados, ou seja :

– Para uma lista de **m** pontos: $(x_i, y_i) \quad i=1, 2, \dots, m$

– Pretende-se uma função aproximante
 $P(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$

tal que: $S_r = \sum_{i=1}^m (y_i - a_0 - a_1 x_i - a_2 x_i^2 - \dots - a_n x_i^n)^2$, seja mínimo.

O mínimo de **S_r** encontra-se para os zeros das derivadas parciais em ordem aos coeficientes **ak** :

$$d S_r = -2 \sum_{i=1}^m x_i^k (y_i - a_0 - a_1 x_i - a_2 x_i^2 - \dots - a_n x_i^n) = 0$$

$$d a_k \quad i=1$$

$$\text{com } k=0, 1, \dots, n$$

Obtendo-se assim um sistema de **n+1** equações, em que as **n+1** incógnitas são os coeficientes **ak** pretendidos e que podem ser determinados através da resolução do sistema de equações que têm a seguinte estrutura :

$$a_0 \cdot \sum (1) + a_1 \cdot \sum (x_i) + \dots + a_n \cdot \sum (x_i^n) = \sum y_i$$

$$a_0 \cdot \sum (x_i) + a_1 \cdot \sum (x_i^2) + \dots + a_n \cdot \sum (x_i^{n+1}) = \sum x_i \cdot y_i$$

$$a_0 \cdot \sum (x_i^2) + a_1 \cdot \sum (x_i^3) + \dots + a_n \cdot \sum (x_i^{n+2}) = \sum x_i^2 \cdot y_i$$

$$\dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \sum = \sum_{i=1}^m$$

$$a_0 \cdot \sum (x_i^n) + a_1 \cdot \sum (x_i^{n+1}) + \dots + a_n \cdot \sum (x_i^{2n}) = \sum x_i^n \cdot y_i$$

Em relação ao programa, este é constituído por 6 ficheiros :

- MAIN.C : Menu principal (Opções)
- MENU.C : Menu Principal
- FILE.C : Entrada de dados (Ficheiro, Utilizador)
- LEASTSQ.C : Aproximação polinomial pelo método dos mínimos quadrados
- GAUSS.C : Resolução do sistema de equações
- GRAPH.C : Gráficos

Todas as funções encontram-se devidamente comentadas, com especial interesse para as funções **least_square** e **gauss** que podem ser utilizadas em outros programas, para aproximação polinomial pelo método dos mínimos quadrados ou então para a resolução de sistemas de equações.



Paulo H. L. Caridade

CONNECTIVIDADE

O (RESTO DE TUDO O QUE) VOCÊ SEMPRE QUIS SABER SOBRE AS «LÃS» (E NÃO OUSOU PERGUNTAR)

Olá leitores.

Eis que, quando menos se esperava, o tema das LAN volta à baila, até porque o prometido em Agosto (**Spooler N.º 19**) é devido. Falámos então dos prós e contras das redes, apresentámos algumas sugestões alternativas à instalação de uma rede e referimos o protocolo ETHERNET. Para este artigo reservámos uma abordagem idêntica aos muito falados TOKEN-RING e NETWARE.

O NOVELL NETWARE

“

A filosofia do NETWARE baseia-se na existência de um *server* dedicado, onde esta máquina efectua a gestão dos equipamentos partilhados.

”

A NOVELL é a empresa líder mundial no segmento de redes de computadores, detendo uma quota de cerca de 70% do mercado mundial. O NOVELL NETWARE é um sistema operativo de rede, e não uma rede propriamente dita. Como tal, trata-se de *software*, que funciona nas estações de trabalho e no *file-server*, efectuando a gestão de dispositivos partilhados entre os diversos utilizadores e das comunicações entre eles.

A filosofia do NETWARE baseia-se na existência de um *server* dedicado, onde esta máquina efectua a gestão dos equipamentos partilhados – discos de grande capacidade, impressoras, discos ópticos, *modems*, etc. O disco do *server* onde é instalado o NETWARE é formatado com um *file-system* proprietário, incompatível com o *file-system* do DOS, mas oferecendo grandes vantagens ao nível da *performance* e segurança dos dados.

Os postos de trabalho (as estações da rede) são PCs vulgares correndo um sistema operativo que tanto pode ser DOS como OS/2 e onde o acesso à rede é garantido por um conjunto de programas residentes, que tratam das transferências de dados entre as estações e o *file-server*. Todos os dispositivos que estiverem ligados ao *file-server* podem ser partilhados: discos, impressoras, *scanners*, etc... As estações podem possuir dispositivos próprios (impressoras ou discos) que, no entanto, serão sempre locais, não sendo possível a sua partilha.

Apesar do NETWARE permitir que o *file-server* seja usado simultaneamente como estação de trabalho («abrindo» uma sessão de DOS), esta dualidade de funções é desaconselhada por questões de segurança e quebra de *performance* global da rede.

Aos olhos do utilizador de um dado posto de trabalho, a rede aparece como um conjunto de discos extra (F:, G:, H:, etc...) – estes discos não existem na realidade, pois são parcelas do disco do *file-server*, com o conteúdo e dimensão determinados pelas permissões concedidas a esse utilizador. Apesar do disco do *server* não estar formatado em

DOS, as máquinas «vêem-no» como um disco DOS, já que o NETWARE se encarrega das necessárias conversões.

De todo o *software* que está instalado na rede, as estações apenas podem executar aquele que «vêem» nos discos, ou seja, aquele para o qual têm permissões. Quando as aplicações estão instaladas nas suas versões de rede, os diferentes utilizadores podem partilhar os dados (por exemplo, uma mesma base de dados ou folha de cálculo).

Ao nível da segurança, o NETWARE é extremamente versátil. As permissões de acesso podem ser dadas a um utilizador individual ou a um grupo de utilizadores (para facilitar a gestão, o administrador da rede pode agrupar os utilizadores conforme os níveis de acesso que pretende conceder a cada um, criando assim os GRUPOS). Consideremos um dado programa que tem permissão de *execute-only* (apenas para execução) para um dado grupo de utilizadores. Assim sendo, esses utilizadores não podem editar, copiar, apagar ou visualizar o conteúdo dos ficheiros em causa, podendo apenas executar os programas!

Como se pode verificar, o NETWARE é um sistema bastante poderoso ao nível da segurança, oferecendo facilidades que são apenas encontradas ao nível dos *mainframes*; por outro lado, é bastante rápido, de forma que qualquer máquina 386 de baixa gama pode ser *file-server* de uma rede de 10 postos de trabalho com total fiabilidade.

Devido à popularidade do NETWARE, existem centenas de aplicações para rede e dispositivos *hardware* que são NETWARE-COMPATIBLE, o que reforça ainda mais a posição de liderança da NOVELL no mercado das redes de computadores. As contrapartidas de todas estas vantagens são, infelizmente, o elevado preço do produto.

Além de ser necessário (ou «conveniente», o que aqui significa o mesmo!) afectar uma máquina à função de *file-server*, o NETWARE propriamente dito é vendido a preços bastante elevados:

sistema para 5 utilizadores	– 115.000\$00
sistema para 10 utilizadores	– 260.000\$00
sistema para 20 utilizadores	– 360.000\$00
sistema para 50 utilizadores	– 578.400\$00
sistema para 100 utilizadores	– 746.200\$00
sistema para 250 utilizadores	– 1.249.600\$00

Os preços apresentados são os que um conceituado revendedor da marca aconselha como preços de venda ao público do NETWARE 386 v3.1.1, o que significa que será possível adquirir este *software* por valores mais baixos. Mas atenção: estes valores não incluem a instalação (nem o mui amado IVA).

A instalação do NETWARE não é propriamente tarefa para principiantes e só a formatação do disco do *file-server* é tarefa que pode demorar 24 a 48 horas!

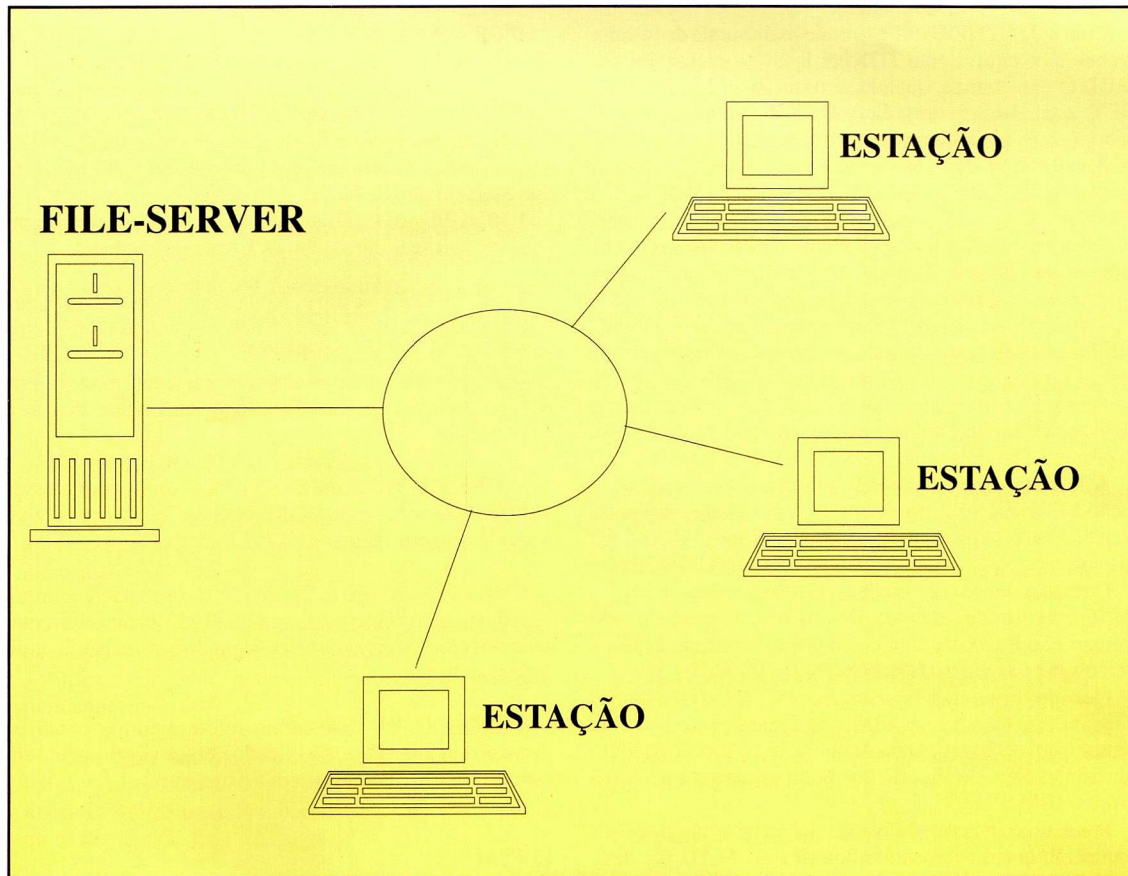
A TOPOLOGIA EM ANEL (RING)

Numa rede com topologia em anel, cada estação está ligada à estação adjacente (em fila), estando ainda a última ligada à primeira.

cumprir as suas funções de receber e passar adiante os pacotes, o tráfego no *ring* pára e a rede deixa de funcionar.

O TOKEN-RING

A rede TOKEN-RING começou por ser desenvolvida pela parceria IBM / TEXAS INSTRUMENTS, tendo apresentado em 1985 a primeira versão, funcionando a 4Mbit por segundo. Segundo os projectistas, a TOKEN-RING



“

Numa rede
TOKEN-RING
os dados são
transferidos
para o ring em
unidades
designadas
por frames.

”

Em cada estação, o adaptador de rede regenera o sinal à medida que recolhe e retransmite o pacote de dados que circula no *ring*. O protocolo mais popular de arbitragem dos acessos ao *ring* nesta topologia é o protocolo do *token*, em que os acessos ao meio de transmissão só são permitidos à estação que tem o *token*.

O *token* é o equivalente ao testemunho numa estafeta 4x400m do atletismo: só corre quem tem o testemunho e um corredor, quando termina a sua prestação, passa o testemunho ao próximo atleta a entrar em cena. Este «testemunho» informático é, na realidade, uma sequência específica de 24 *bits*.

Numa rede de topologia em anel, quando nenhuma estação está a enviar dados, a única coisa que está em circulação na rede é o testemunho (o *token*); como ninguém precisa de transmitir dados, quando o *token* atinge a sua estação, regenera-o e envia-o para a estação adjacente; assim que uma dada estação necessita de transmitir, ao receber o *token* mantém a sua posse, sem o passar adiante; ao ficar com ele, passou a controlar o *ring*; então, envia os seus pacotes de dados; mesmo que outra estação necessite de comunicar, como não tem o *token* não o pode fazer (tem de esperar por ele).

A grande desvantagem desta topologia é a sua dependência do bom estado do anel – se uma ligação entre duas estações se interrompe ou um qualquer adaptador deixa de

tem vantagens de rapidez e fiabilidade em relação à ETHERNET, nomeadamente em redes de grande porte. A TOKEN-RING beneficiou ainda do facto de ser o *standard* IBM para ligações entre PC, MAINFRAMES, AS/400 e sistemas RISC/6000.

Numa rede TOKEN-RING (definida pelo *standard* IEEE 802.5), os dados são transferidos para o *ring* em unidades designadas por *frames*. As *frames* são enviadas sequencialmente de uma estação para a seguinte, e assim sucessivamente, onde cada estação repete a *frame* (o teste de erros é feito em cada uma destas repetições). Quando o endereço do destinatário é o seu próprio endereço, a estação copia os dados para si mas passa-os igualmente adiante. Depois de ter dado a volta ao *ring*, quando a *frame* atinge de novo a estação emissora, esta retira a *frame* de circulação.

Quando o adaptador de rede é inicializado (ao ligar a máquina e aceder à rede), o processo de inicialização garante que não vão coexistir dois endereços iguais no *ring*. A rede TOKEN-ring usa uma transmissão em banda-base (*baseband*), operando a uma velocidade de 4 ou 16 Megabit por segundo (4Mbps ou 16Mbps). As redes TOKEN-RING podem também usar um meio de transmissão em fibra óptica, onde a velocidade típica é de 100Mbps.

Para a criação do *ring* físico, uma vez que não é cómoda a existência de um cabo em anel ligando todas as máquinas, utiliza-se uma caixa designada pela sigla MAU (*Multiple*

Access Unit), à qual todas as máquinas individuais são ligadas por um único cabo (contendo internamente 4 condutores).

A «ETHERNA» COMPETIÇÃO ETHERNET/TOKEN-RING

As vantagens ou desvantagens de uma determinada rede só podem ser pesadas por comparação com outra. E o termo de comparação mais vulgar para uma rede ETHERNET é a TOKEN-RING (e vice-versa).

Uma rede ETHERNET é sempre mais barata de instalar do que uma equivalente TOKEN-RING – eventualmente MUITO mais barata, quando se tratar de THIN ETHERNET, a tal que usa cabo fino. E «mais barata» significa cabo, fichas, placas e restante *hardware* mais barato!

A velocidade de transmissão da ETHERNET é actualmente de 10Mbit por segundo (está aí a ETHERNET a 100Mbit...). Para um número de utilizadores até à meia centena, as colisões não degradam significativamente os tempos de resposta da rede; no entanto, para números superiores, os atrasos podem ser significativos.

A rede TOKEN-RING é apresentada em dois sabores (4Mbit e 16Mbit por segundo) e apresenta a vantagem do tempo de resposta não sofrer alteração sensível com o aumento do número de estações em rede. No entanto, a IBM entrou também na guerra das velocidades e declara-se já pronta a fornecer redes TOKEN-RING a 100Mb.

A TOKEN-RING apresenta algumas vantagens em relação à ETHERNET: por exemplo, é possível acrescentar ou retirar estações da rede sem ser necessário interromper o seu funcionamento; isto não pode ser feito com a ETHERNET.

Quanto à questão da melhor performance da TOKEN-RING em casos de redes com muitos utilizadores e elevado tráfego não é tão pacífica e sempre foi contestada pelos defensores das redes ETHERNET.

Quando se pretende ligar em rede PC, AS/400, sistemas RISC/6000 e mainframes IBM, esta é igualmente a escolha sensata, pois existe garantia de que as ligações vão de facto funcionar, além do suporte que pode sempre ser pedido à própria IBM.

Por último, na contabilização dos custos não pode ser esquecido que é necessário adquirir uma MAU por cada lote de 8 utilizadores (a MAU é indispensável para a formação do *server* e o seu custo é de cerca de cem mil escudos por unidade).

EXPLICAÇÃO DE ALGUM «CALÃO» TÉCNICO

SISTEMA OPERATIVO DE REDE – é o *software* que faz funcionar a rede («correndo» no *server* e nas estações) tratando das comunicações e tornando viável a partilha de um mesmo recurso por vários computadores.

BASEBAND ou BANDA-BASE – é um tipo de transmissão de informação pelo meio físico sem envolver a modulação de um sinal portador. Em cada instante, a informação ocupa toda a largura de banda disponível no meio. A partilha de um meio físico em banda-base é feita pela técnica de *Time Division Multiplexing* (TDM), que é o mesmo que dizer que o tempo de utilização do meio é dividido em parcelas pelos diferentes utilizadores. A tecnologia envolvida numa transmissão em banda-base é relativamente barata.

CARD DE REDE ou ADAPTADOR DE REDE – define a placa de comunicações que é instalada num computador e que permite que este se insira numa rede e, dessa forma, possa comunicar com outros computadores,

seja para enviar dados, produzir listagens em impressoras de rede ou aceder a ficheiros/programas residentes noutros computadores (da rede).

TOKEN-RING ou IEEE 802.5 – é um sistema *hardware* de rede baseado na topologia de anel (*ring*), desenvolvido fundamentalmente pela IBM.

ETHERNET – é um sistema *hardware* de rede baseado na topologia de «barramento» (*bus*), desenvolvido fundamentalmente pela NOVELL.

IEEE – Sigla do *Institute of Electrical and Electronic Engineers*, que se dedica à elaboração de normas (*standards*) para manter estáveis as definições funcionais de diversos produtos, nomeadamente dos sistemas de redes de computadores, o que permite que empresas distintas possam desenvolver produtos para as redes normalizadas sem correrem o risco de que as suas definições se alterem.

BRIDGES ou GATEWAYS – são estações especializadas em interligar duas ou mais LAN ou sistemas.

FILE-SERVER – é um computador que dispõe de recursos que partilha com outras máquinas; como os recursos são seus, é o FILE-SERVER que coordena a distribuição das *benesses* por quem necessita (as estações); resta referir que os recursos partilhados são normalmente discos e impressoras.

WORKSTATION ou STATION – é um computador vulgar, funcionando em DOS ou OS/2 e ligado a uma rede, podendo utilizar recursos do FILE-SERVER.


CROSSTALK – é o termo que define a interferência que um sinal, num dado cabo, produz num cabo adjacente, criando um sinal parasita neste segundo cabo e danificando o sinal original.

TOPOLOGIA – este termo refere a forma como os dispositivos de rede estão ligados entre si (em anel, em estrela, em «barramento», em árvore, etc).

O FIM

Sem a pretensão de que estes artigos vos tenham tornado especialistas em redes de computadores (até porque foram omitidas aqui e além algumas partes mais complicadas da história) esperamos, no entanto, que tenham servido para aumentar a vossa CGI (CGI = Cultura Geral Informática) e assim permitir-vos ter uma ideia do funcionamento geral das «lãs». E, sem dar por isso, chegámos ao FIM (e do FIM não passaremos, como aliás convém).

“
Uma rede ETHERNET é sempre mais barata de instalar do que uma equivalente TOKEN-RING eventualmente MUITO mais barata, quando se tratar de THIN ETHERNET.
”



Paulo Bernardo

EDIC

COLABORAÇÃO
DO LEITOR

EDIC é um editor de caracteres para a placa VGA, modo de texto 80x25. Permite redefinir os caracteres do código ASCII e gravar essa definição em ficheiros fonte. Estes novos caracteres podem ser utilizados através das rotinas em Pascal que se encontram na disquete ou executando o programa LEFNT. A definição de caracteres permite mudar radicalmente a apresentação de um programa.

UTILIZANDO EDIC

Para utilizar EDIC é necessário um rato. Depois da apresentação aparece o ecrã de trabalho com os seguintes elementos:

Ficheiro corrente (em cima): Este nome será utilizado quando a opção Gravar for executada.

O nome do ficheiro pode ser alterado com a opção Nome do menu.

Menu (em baixo): A última linha do ecrã contém o menu. Cada palavra constitui uma das opções que pode ser executada pressionando o botão esquerdo com o rato em cima da palavra.

Tabela do código ASCII (à esquerda): Utilizada para escolher o carácter a definir utilizando a opção Carácter do menu. Sempre que o rato aponta para a tabela é indicado por baixo desta o código do carácter apontado.

Grelha de edição (ao centro): É neste espaço que editamos o carácter. Em volta da grelha estão referenciadas as linhas e colunas através de letras e números. Em baixo encontram-se: a coluna, a linha, o carácter em edição e o código desse carácter. Para desenhar e apagar prima os botões esquerdo e direito do rato.

Quadro de testes (à direita): Este espaço é utilizado para ver os caracteres definidos. O utilizador pode desenhar caixas, fundos etc..

Restaurar vídeo (sinal °° no canto superior direito): Para restaurar o modo de vídeo basta premir o botão esquerdo com o rato em cima do sinal °°.

OPÇÕES DO MENU

Carácter – Através desta opção será escolhido a carácter a editar; para tal basta premir o botão esquerdo do rato no carácter desejado. Pode utilizar a tabela de códigos ASCII do lado esquerdo do ecrã ou qualquer outro ponto do ecrã.

Restaurar – Se desejar voltar atrás depois de uma alteração pode fazê-lo utilizando esta opção.

Original – Esta opção copia a definição existente na ROM para a tabela de EDICão.

Limpar – Limpa o carácter...

Copiar – Copia um carácter para outro.

Nome – Atribui um novo nome. Este nome será utilizado pela opção Gravar.

Gravar – Grava a fonte corrente com o nome corrente.

Abrir – Lê uma fonte previamente gravada. A extensão por omissão é .FNT .

Teste – Coloca o cursor no quadro de testes e permite a sua alteração.

UTILIZANDO AS FONTES GRAVADAS

Para utilizar as fontes gravadas basta incluir o ficheiro «Rotina.PAS» no seu programa e chamar a rotina «CarregaFNT» da seguinte forma:

```
if CarregaFNT(<fich>) then { Erro ! } ...
```

Esta rotina irá carregar a fonte indicada por <fich> e se tudo correr bem devolve FALSE; caso contrário devolve TRUE.

Veja os exemplos: "EXEMPLO.PAS" e "LEFNT.PAS" na disquete.

CORRENDO OS EXEMPLOS

O ficheiro LEFNT.EXE carrega uma fonte cujo nome deve ser passado como parâmetro. Exemplo:

```
LEFNT medieval.fnt
```

Este programa pode ser utilizado por exemplo no AUTOEXEC.BAT para que o utilizador possa arrancar o DOS com a sua fonte preferida. O ficheiro EXEMPLO.EXE utiliza a fonte DEMO.FNT para exemplificar a definição de novas caixas e fundos.

MEMÓRIA ESTENDIDA

PARTE II

No número anterior da nossa revista iniciámos esta nova série de artigos versando a temática da **memória estendida** e do **modo protegido de 32 bits**.

E tivemos nessa altura oportunidade de apresentar um método de acesso a dados da **memória estendida**, designado normalmente por método INT 15H, ou mais correctamente por serviço do BIOS *Protected-Mode Data Move*. E porque este serviço é providenciado pelo BIOS de todos os computadores de modelos compatíveis com os IBM AT e PS/2 ele está, por conseguinte, implicitamente ao inteiro dispor de todos nós.

Esse método contudo não é de utilização muito fácil para efeitos da transferência de dados, o que até poderia ser um mal menor, não fosse o facto de apresentar também vários outros inconvenientes, como se terão apercebido os leitores que analisaram o código fonte do programa EXTVIEW.EXE que ilustrava o artigo. Apesar disso é um método utilizado ainda por muitos programas nos nossos dias. Tudo dependerá por conseguinte do objectivo que se pretenda atingir.

No artigo de hoje e no próximo vamos tentar explicar um pouco aos leitores interessados nestas matérias, mas ainda desconhecedores, os princípios de funcionamento dos processadores Intel de 32 bits em **modo protegido**, modo esse que é, como se sabe, o único modo de funcionamento do processador em que é possível o acesso a dados e a execução de programas acima do primeiro Megabyte de memória (mais correcto será afirmar, como estarão lembrados, 64 KB menos 16 bytes acima do primeiro Megabyte).

Como este assunto não é assim muito trivial, alguns pequenos conhecimentos de linguagem *Assembly*, embora não absolutamente essenciais, ajudarão um pouco na compreensão geral do conteúdo do texto que se seguirá.

Mas a compreensão do funcionamento e análise do pequeno programa que preparámos para ilustrar o tema do artigo e que consta da disquete **Spooler** desta edição, o PROTECT.EXE, exigirá contudo um conhecimento razoável de linguagem *Assembly* (e também algumas luzes de linguagem C já que o prólogo e o epílogo foram construídos nessa linguagem por razões de produtividade).

O PROTECT.EXE é em si mesmo um programa sem utilidade prática e que não faz nada de especial, excepto enviar duas mensagens do lado de lá (isto é do segundo Megabyte) informando-nos o que está a fazer e onde. Contudo, o estudo do seu funcionamento a partir do código fonte ajudará bastante, segundo cremos, a solidificar algumas conceitos base que aqui iremos expor.

MEMÓRIA SEGMENTADA

Em **modo protegido** a palavra **segmentação** não tem o mesmo sentido que se dá a esse termo em **modo real**. Em **modo real**, como se sabe, a memória acedível por um

programa em dado momento é dada por um «dueto» que consiste no valor existente num registo de segmento e por um deslocamento (ou *offset*) relativo ao valor existente no mesmo registo de segmento.

Os registos de segmento percorrem a memória a «passadas» de 16 bytes (designado por um parágrafo) de cada vez e o deslocamento por sua vez é limitado a um número máximo de 16 bits, isto é pode ir de 0 até 65535 (ou 0FFFFh em hexadecimal).

Considere-se por exemplo a instrução Assembly: MOV AL, ES:[1234h]

Após a sua execução, o registo AL do processador ficará carregado com o valor do byte existente na localização de memória identificada pelo endereço 1234h dentro do segmento ES.

Se nesse momento o segmento ES contivesse o valor 2000h, o **endereço físico** (endereço absoluto a partir do início da memória) do byte na memória seria:

$$2000h \times 10h + 1234h = 21234h$$

Mas os registos de segmento são também registos de 16 bits, portanto o máximo endereço físico de memória que é possível aceder em **modo real** é, seguindo o mesmo raciocínio:

$$0FFFFh \times 10h + 0FFFFh = 10FFEFh$$

(ou seja 1114095 em decimal)

Essa posição de memória corresponde precisamente a 64 KB menos 16 bytes acima do primeiro Megabyte. Esta é a grande limitação do funcionamento do processador em **modo real**.

Em **modo protegido** o processador faz a gestão do endereçamento à memória de modo significativamente diferente.

Os programas não endereçam directamente a memória, mas sim um modelo designado por **memória virtual** (não confundir com o modo de trabalho designado por virtual 86). Cabe a dois mecanismos internos ao processador desfazerem a ambiguidade. Um deles, designado por **mecanismo de segmentação**, possibilita a existência de múltiplos espaços de endereçamento independentes; o outro designado por **mecanismo de paging** permite a existência de um grande espaço de endereçamento em memória dispondo-se de relativamente pouca memória RAM mas em compensação suficiente espaço em disco. Cada um dos mecanismos isoladamente, ou ambos simultaneamente, podem ser activados por certos programas (normalmente pertencendo ao sistema operativo ou então que se substituem ao sistema operativo nessas missões).

Um endereço emitido dentro de um programa é um **endereço lógico** e compete, em primeira instância, ao mecanismo de segmentação a oportunidade de traduzir

“
No artigo de hoje e no próximo vamos tentar explicar um pouco aos leitores interessados nestas matérias, mas ainda desconhecedores, os princípios de funcionamento dos processadores Intel de 32 bits em modo protegido.”

esse endereço lógico num endereço absoluto designado por **endereço linear**.

Se o mecanismo de *paging* não estiver activo, o endereço linear corresponde de imediato ao endereço físico. Se estiver activo, será o mecanismo de *paging* que determinará a partir do endereço linear qual é o endereço físico.

O mecanismo de segmentação permite gerir segmentos de qualquer tamanho, deste 1 byte até 4 Gigabytes de tamanho e não, tal como acontecia em modo real, apenas segmentos de 64 KB.

Dentro de um dado programa é normal criarem-se vários segmentos independentes com características muito bem definidas. Haverá segmentos reservados ao código executável do programa, haverá outros reservados aos seus dados e um ou mais às suas pilhas (ou *stacks*).

Dentro de cada segmento e um pouco à semelhança do que aconteceria em modo real, o endereço linear é determinado pelo deslocamento dentro de um segmento, mas com uma diferença muito importante: os registos de segmento não contêm valores em parágrafos correspondentes directamente a um certo endereço físico de memória, tal como aconteceria em modo real, mas sim valores designados por **selectores** (veja por favor a figura seguinte).

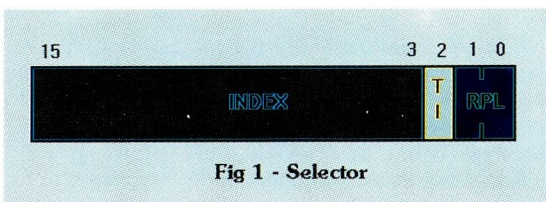


Fig 1 - Selector

Os selectores contêm índices (cujo valor é dado pelos bits 3 a 15) relativos a tabelas em RAM, designadas por *Descriptor Tables*. As *Descriptor Tables* contêm *Segment Descriptors* ou simplesmente *Descriptors* (que serão detalhados mais à frente).

Um programa terá sempre de efectuar a sua escolha entre duas alternativas de *Descriptor Tables*, a qual é formalizada através do bit 2 do selector (designado por *Table Indicator* ou **TI**).

Há que optar entre a *Global Descriptor Table* ou **GDT** (seleccionada com **TI=0**) e a *Local Descriptor Table* ou **LDT** (seleccionada com **TI=1**).

No sistema está disponível uma única GDT para todos os programas, e uma LDT para cada programa em execução. Contudo, podem ser concebidos sistemas operativos em que todos os programas compartilhem uma única LDT (o Windows 3.xx comportando-se como uma extensão ao sistema operativo MS-DOS actua desse modo). Pode também ser concebido um sistema sem LDT, em que todos os programas utilizem apenas a GDT.

O selector contém ainda um campo constituído pelos bits 0 e 1 e que é designado por *Request Privilege Level* ou **RPL**.

E antes de continuarmos vamos analisar um pouco o que se entende por «privilégios», pois daí deriva precisamente um pouco da explicação (mas não toda) do nome «modo protegido».

O processador dispõe de um dispositivo de protecção que reconhece 4 níveis (ou *rings*) de privilégio numerados de 0 a 3. Quanto maior o número, menor o nível de privilégio.

Uma «General Protection Exception» (*Exceptions* ou «Excepções» como se queira, serão tratadas no próximo número) é gerada sempre que um programa tente aceder a um segmento utilizando um nível de privilégio inferior ao que se aplica a esse segmento. O controle do nível de privilégio é efectuado por três estruturas:

- 1) Os bits 0 e 1 dos segmento CS (ou SS). Esse valor indica ao processador o *Current Privilege Level* ou

CPL. O CPL contém o nível de privilégio do programa em execução. Normalmente o CPL é igual ao nível de privilégio do segmento de código (isto é, o segmento em que estão a ser lidas as instruções do programa). Se o programa transferir controlo para um segmento de código menos privilegiado, o CPL é ajustado para esse nível. O programa normalmente não pode transferir controlo para segmentos de código mais privilegiados, a menos que esses segmentos sejam do tipo *conforming*. Um segmento de código *conforming* é executado com o nível de privilégio da rotina que o chamou, mas aqui o CPL não será alterado (nos dois primeiros bits do registo CS), mesmo que explicitamente se altere o RPL na execução de instruções **JMP FAR** ou **CALL FAR**.

- 2) Os bits 45 e 46 do *Segment Descriptor* contêm um campo designado por *Descriptor Privilege Level* ou **DPL**, o qual representa o nível de privilégio do segmento.
- 3) Finalmente, o RPL que vimos acima e que é coincidente com o CPL nos segmentos de código e pilha. Se o RPL para um selector de dados for menos privilegiado (valor numérico maior) que o CPL, o acesso à memória é efectuado ao nível de privilégio do segmento de dados. Por conseguinte, um programa só pode aceder a um segmento se o DPL do segmento for o mesmo ou menos privilegiado que o menos privilegiado dos CPL e RPL.

Vimos atrás que os selectores indexam *Descriptors* em *Descriptor Tables*.

Pela Figura 2, vemos que cada *Descriptor* consta de 64 bits divididos por vários campos, os quais dizem não só da localização e tamanho dos segmentos, mas também contêm várias informação de controlo e *status*. Vamos ver esses campos um a um, pedindo desde já as nossas mais sentidas desculpas pelo amplo uso que iremos fazer (e que fizemos também até aqui) dos termos originais em inglês, mas efectivamente só iríamos aumentar a confusão presente (e especialmente futura) do leitor se nos atrevêssemos a inventar traduções para termos já internacionalmente bem firmados.

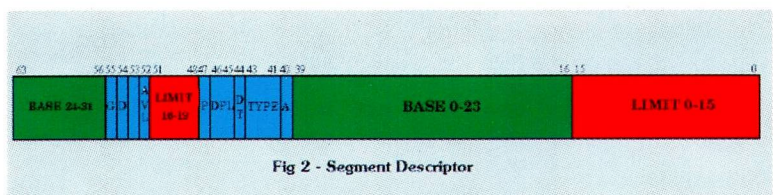


Fig 2 - Segment Descriptor

LIMIT – Define o tamanho do segmento. O processador concatena os dois campos de Limite e forma um valor de 20 bits. Esse valor pode ser interpretado de 2 modos:

- Se o bit de Granularidade **G** (bit 55) for 0 (*Byte Granularity*) o segmento terá um tamanho compreendido entre 1 byte e 1 megabyte (2^{20} bytes).
- Se o bit de Granularidade for 1 (*Page Granularity*) o segmento terá um valor compreendido entre 4 KB e 4 Gigabytes. Por conseguinte, aqui o incremento processa-se 4 KB (designado por uma página) de cada vez..

Quase sempre um endereço lógico terá de ter um deslocamento compreendido entre 0 e o valor do Limite. Programas que gerem deslocamentos fora do limite dão origem a *Exceptions*.

Podem contudo definir-se segmentos do tipo *Expand-Down* (ver mais à frente) em que, pelo contrário, os endereços podem ser feitos com qualquer deslocamento excepto valores compreendidos entre 0 e o Limite. Este tipo de segmento é algumas vezes utilizado para *stacks* que contemplem a possibilidade de ser expandidos (para baixo claro).

Nota: A razão pela qual o Limite é formado dentro do *Descriptor* por dois campos em posições diferentes tem a ver com a necessidade de se manter compatibilidade com os processadores 80286. Até ao bit 47 os *Descriptors* são idênticos tanto nos 80286 como nos modelos de 32 bits, mas os bits 48 a 63 são «reservados» nos 80286.

BASE – Define a localização do segmento no espaço de endereçamento de 32 bits (4 Gigabytes). O processador concatena os três campos que definem a Base para formar o valor requerido de 32 bits.

ACCESSED (A) – Se este bit for 0 isso significará que o selector para o *Descriptor* não foi transferido para um registo de segmento. Se for 1 acontece o contrário. Os *Descriptors* inicialmente referem todos os segmentos como *Accessed*, mas se num certo momento se colocarem todos a 0, então será possível a partir daí controlar os segmentos que foram acedidos.

Existe uma certa analogia com o conhecido *Archive* bit dos ficheiros.

TYPE – Este campo define as operações permitidas no segmento. Se se tratar de um segmento de memória a interpretação a dar é a seguinte:

- 000 – Segmento de Dados, *Read-Only, Expand-Up*
- 001 – Segmento de Dados, *Read/Write, Expand-Up*
- 010 – Segmento de Dados, *Read-Only, Expand-Down*
- 011 – Segmento de Dados, *Read/Write, Expand-Down*
- 100 – Segmento de Código, *Execute-Only, Nonconforming*
- 101 – Segmento de Código, *Execute-Read, Nonconforming*
- 110 – Segmento de Código, *Execute-Only, Conforming*
- 111 – Segmento de Código, *Execute-Read, Conforming*

Nota: Se não se tratar de um segmento de memória (e se trata de um segmento de sistema ou *task gate*) a interpretação será algo diferente mas inibimo-nos de a apresentar apenas por ultrapassar o âmbito introdutório deste artigo.

Descriptor Type (DT) – Os *Descriptors* de segmentos de memória recebem 1 neste campo. Os *Descriptors* de segmentos de sistema e *gates* recebem 0. Ver por favor a Nota anterior.

Descriptor Privilege Level (DPL) – Já nos referimos a ele mais atrás. Este campo é utilizado, como se explicou, para o processador controlar o acesso ao segmento.

Segment Present Bit (P) – Se este bit estiver a 0 o processador gerará uma *Segment-Not-Present Exception* se um selector para o *Descriptor* for carregado num registo de segmento. Essa *Exception* tem por objectivo alertar o sistema operativo sobre o acesso a segmentos indisponíveis (que podem por exemplo ter sido guardados no disco rígido). Como resultado dessa *Exception* o sistema operativo tem então a oportunidade de providenciar para que esse segmento fique disponível de novo (por exemplo recarregando-o do disco para a memória) ao programa de aplicações e de um modo totalmente transparente para este. Vê-se assim a importância deste bit na gestão da memória virtual.

Available Bit (AVL) – Como a tradução indica «Disponível» este bit pode ser utilizado pelo *software* do sistema para qualquer tipo de controle que entenda ser conveniente.

Bit 53 – Este campo é reservado pela Intel para futuros microprocessadores. Deverá ser mantido a zero, até se saber mais pormenores dessas intenções.

Default Bit (D) – Este bit determina qual é por omissão o tamanho do operando e do endereço nas instruções executadas. Se o segmento de programa for de 32 bits, o valor deste bit deve ser 1. Para os segmentos de dados e pilha este bit recebe o nome de «Big Bit» (B).

Granularity Bit (G) – Já foi referido atrás quando tratamento do campo de Limite.

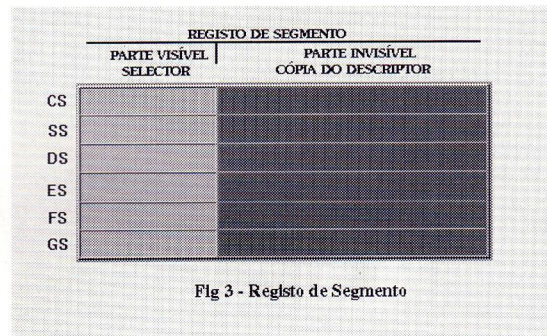


Fig 3 - Registo de Segmento

Os registos de segmento são constituídos por uma parte visível e acessível por *software* e uma parte invisível (Fig 3). Quando um selector é carregado num registo de segmento, o processador carregará automaticamente a parte invisível do mesmo registo de segmento com a informação constante do *Descriptor*. Este pormenor é importante, pois significa que o processador a partir daí procederá exactamente em conformidade com o conteúdo do *Descriptor* e sem qualquer perda de desempenho.

Resta-nos referir ainda que a Base e Limite da GDT constam de um registo do processador de nome GDTR (*Global Descriptor Table Register*). No caso de LDT, a sua Base, Limite e selector da GDT que contém o *Descriptor* do segmento onde se encontra a LDT constam de um registo do processador de nome LDTR (*Local Descriptor Table Register*).

Para visualizar melhor como toda esta engrenagem joga, os leitores que pela primeira vez lêem sobre este assunto, deverão investir algum tempo na análise do conteúdo da Figura 4.

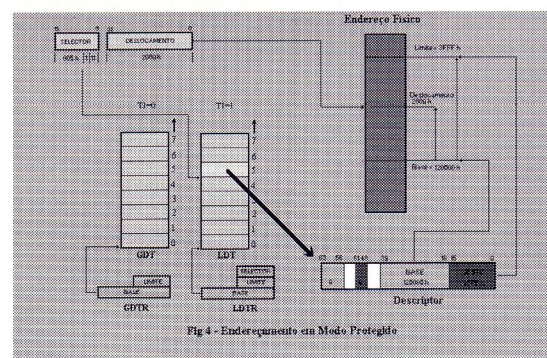


Fig 4 - Endereçamento em Modo Protegido

MODELO «FLAT»

O modelo «FLAT» (ou de Base 0) de memória é um caso particular do modelo de memória segmentada, mas em que (parece contraditório mas não é) pura e simplesmente a segmentação desapareceu. Esse efeito consegue-se mapeando todos os segmentos para o mesmo endereço físico de memória. A Base dos segmentos será 0 e o Limite 4 Gigabytes. Este modelo é interessante para sistemas operativos como o Unix que não suportam segmentação mas podem suportar *paging*.

O modelo «FLAT» pode também ser «protegido» se os Limites forem estabelecidos para se fixarem apenas em zonas para as quais existem efectivamente endereços físicos. Esta situação designa-se por modelo *Protected FLAT*.

E no próximo número cá estaremos de novo para completar a introdução a esta interessante mas indiscutivelmente delicada matéria. E naturalmente contamos aqui com os nossos leitores, em particular o núcleo duro dos que não desistem à primeira.

TESTE C – A SOLUÇÃO

Antes de mais, queria agradecer a todos aqueles que colaboraram com a **Spooler** participando no teste de C. No entanto, é de lamentar o escasso número de participantes neste teste/concurso, pois apenas 10 leitores participaram. Quanto ao teste, este era constituído por 6 perguntas que de seguida iremos analisar:

- 1 – Nesta primeira questão começava-se logo por testar a atenção do leitor. Tratava-se de uma cadeia de **if** onde os **else** se encontravam desalinados (na vertical) com os respectivos **if**, o que poderia iludir o leitor menos atento. Como resposta aceitava-se (por exemplo): **O programa não tem output**. Resta ainda salientar que todos os 10 participantes responderam acertadamente a esta questão.
- 2 – Ao contrário da questão anterior em que todos os leitores acertaram, esta foi onde mais participantes erraram: 60%. Era constituída por um problema de inicialização de variáveis locais. Como foi dito na lição N.º 3 do curso de C (pag.37), qualquer variável local não inicializada poderá ter no seu conteúdo qualquer valor. No caso específico desta questão era impossível determinar qual dos dois ponteiros **p1** ou **p2** (declarados dentro da função «abc») apontava para um endereço mais baixo. Assim, poderia ser aceite como resposta: **Impossível de determinar**.
- 3 – Esta questão era talvez a mais simples do teste, onde mais uma vez todos os participantes responderam correctamente. Pretendia-se saber se a declaração de uma variável estava ou não correcta. Esta estava errada, pois o nome da variável tinha como primeiro carácter um número, o que não é permitido no nome de um identificador.
- 4 – Nesta questão era proposta ao leitor a resolução de uma expressão matemática contendo vários operadores, muitos parêntesis e um número em base 8 (octal). Segue-se a resolução:

$$(34 + 56 * (34 + (2 \& 9)) / (20 - (26 \wedge 016))) - 224$$

$$(34 + 56 * (34 + 0) / (20 - (26 \wedge 14))) - 224$$

$$(34 + 1904 / (20 - 20)) - 224$$

$$1938 / 0 - 224$$

Tal como se pode ver, a simplificação da expressão leva-nos à detecção de um erro de divisão por zero. Nesta expressão existiam alguns elementos que poderiam induzir o leitor em erro:

- **016** é um número octal em C que corresponde a **14** em decimal
- **2 & 9** é zero.

(2)	0010
(9)	1001 &
(0)	0000
- $26 \wedge 14$ é 20.

(26)	11010
(14)	01110 ^

(20)	10100

Como resposta a esta questão aceitava-se: **Erro de divisão por zero**.

- 5 – Esta questão poderia ter «mil e uma» soluções diferentes, no entanto foi dada prioridade aos programas que continham algoritmos mais engenhosos e simples, sendo logo à partida excluídos aqueles que utilizavam funções da *Standard Library*. A solução apresentada é uma das possíveis:

```
#include <stdio.h>      /* Inclui os protótipos das funções utilizadas */

/*
  Esta função lê caracteres do teclado até que o utilizador pressione
  ENTER, escrevendo de seguida todos os caracteres por ordem contrária à
  de leitura. Como facilmente se pode ver é utilizado um algoritmo
  recursivo, isto é, sempre que é lido um carácter a função chama-se a si
  própria para efectuar a leitura de novo carácter. Quando o utilizador
  pressionar ENTER, a recursividade termina, escrevendo todos os
  caracteres começando pelo último e terminando no primeiro.
*/
void invert(void)
{
    char ch;           /* para guardar o carácter a ler */

    ch = getchar();   /* lê o carácter */
    if (ch != '\n') { /* se não for um ENTER */
        invert();     /* repete toda a operação */
        putchar(ch); /* escreve o carácter lido atrás */
    }
}

main(void)
{
    printf("\nIntroduza a string a inverter:\n");
    invert();         /* lê os caracteres e escreve-os começando pelo
último */
    return 0;        /* retoma 0 para o sistema operativo */
}
```

- 6 – Como resposta a esta questão pretendia-se que a versão final da calculadora desenvolvida ao longo do curso, interpretasse mais uma função. Esta deveria calcular o inverso de um número e ter como nome **INV**. Segue-se uma lista das alterações efectuadas ao longo do ficheiro **CALC.C**:

– Inserção do protótipo da nova função **inv()**

```
double inv(double);
```

- Alteração do número máximo de funções de **8** para **9**

```
#define MAX_FUNC 9
```

– Adicionar no *array f* a função **INV**

```
struct ftab f[MAX_FUNC] = {
    ...
    {"INV", inv } /* retoma o inverso de um número
*/
};
```

– Definição da função **inv()**

```
/* retoma o inverso de um
número. Será sinalizado erro
de divisão por zero */
/* se o número tiver como valor
0. */

double inv(double d)
{
    if (d == 0.0) erro(DIV_ERR); /* se o número for zero, erro */
    return 1/d; /* retoma o inverso do número */
}
```

– Alterar o texto de ajuda de forma a dar conhecimento ao utilizador da existência da função **INV**

```
char *ajuda =
```

```
“
    _____ INV(x) calcula o inverso de x\n”
“
    <>,<FIM> - Termina a execução do programa.\n”;
```

NOTA FINAL

Resta agora atribuir o prémio ao concorrente cujo teste foi o melhor classificado. Tratou-se de um trabalho de equipa realizado por **Artur Miguel Santos Gaspar** e **Mário Augusto Godinho**, ambos estudantes de Engenharia. O teste desta equipa salientou-se de entre os quatro que responderam a todas as questões acertadamente, porque foi o que no geral apresentou as melhores respostas às perguntas 5 e 6. Na disquete deste número, o ficheiro TESTEC.EXE contém as fontes das soluções e o ficheiro TESTE.TXT contém as soluções enviadas pela equipa. Para finalizar, espero que este mini-curso tenha contribuído para a valorização pessoal (relativamente à linguagem C) não só dos 10 leitores que nos enviaram o teste, como também de todos os que tiveram a paciência de nos ler.

Carlos Ladeira

**COLABORAÇÃO
DO LEITOR**

CONFIG

O nosso amigo Hugo Meneses oferece-nos um interessante conjunto de utilitários destinados a permitir uma configuração do sistema adaptada a casos particulares. Os exemplos que apresenta são típicos das dificuldades de um 286 em lidar com aplicações exigentes. Mas o «engenho e a arte» conseguem colmatar muitas dessas limitações – foi o que o Hugo fez, e bem. Mas conttenham-se: não mexam em nada sem uma leitura atenta do artigo e um estudo não menos apurado dos ficheiros de comandos que seguem na disquete.

No local onde trabalho, tenho um PC, um IBM Personal System/2 Colour Display com um processador 286, 1 Mega de memória e um disco rígido de 30 Megs (não, não estou a fazer publicidade). Nesse PC tenho o Windows 3.1 (onde uso o Write), o Quattro Pro Versão 4 (onde uso folhas de cálculo «pequenas», algumas de 100 000 a 200 000 bytes quando gravadas) e o Clipper 5.01 (onde programo). Como podem imaginar, este PC é «muitíssimo rápido» para trabalhar com qualquer destas aplicações...

Como seria de prever, fiquei com um pequeno problema: o Windows quer memória estendida, o Quattro quer expandida e o Clipper quanto mais convencional melhor. Para resolver este problema tinha de ter uma disquete com o MS-DOS 5 quando queria trabalhar com o Windows, alterar o CONFIG para usar o EMS40.SYS quando queria o Quattro e assim por diante. Fartei-me e resolvi o problema através dos programas que seguem na disquete deste número. Uma *batch file*, um programa GETOPT.EXE (com o respectivo código fonte em C, já agora aviso que não fui eu que o fiz e não sei quem o fez), o já publicado RESET.COM e um programa chamado ESC.COM (que já vos tinha enviado aquando do meu artigo sobre *batch files*).

O funcionamento é simples: cria-se um directório CONFIG (com o comando MD CONFIG) depois copiam-se os ficheiros CONFIG.TXT, CONFIG.BAT, GETOPT.EXE, RESET.COM e ESC.COM e depois copiam-se os CONFIG.* e os AUTOEXEC.*. Já agora aviso que os comandos são do MS-DOS 3.30 visto que é o sistema operativo utilizado neste PC e que não posso mudar porque a emulação ao IBM SYSTEM/36 não funciona com outro sistema operativo, conforme me foi dito. Ao correrem o programa CONFIG verão aparecer um menu no ecrã (se o ecrã de fundo não estiver a cores é porque não têm o ANSI.SYS instalado e para isso deverão introduzir a linha DEVICE=C:\DOS\ANSI.SYS no vosso CONFIG.SYS). Depois de escolherem a configuração desejada, o programa copia o CONFIG.XXX e o AUTOEXEC.XXX para a raiz e depois pergunta se querem fazer um *reboot* ao PC: se quiserem o computador arranca com o novo CONFIG e AUTOEXEC para a configuração desejada.

Os CONFIGS e os AUTOEXECs (lindos plurais...) enviados devem ser alterados de acordo com os vossos casos particulares, assim como a *batch file* CONFIG.BAT. Verão que é simples com os exemplos que lá estão. O modo de usar o programa GETOPT.EXE também é fácil conforme verão na *batch file*, bastando apenas usar **getopt** “Nome do MENU” “Opcao 1” “Opcao 2” ... “Sair”. Já agora aviso que também trabalha com o rato.

Espero que gostem da ideia e até à próxima.

NOTA: Não se esqueçam de fazer uma cópia de segurança dos vossos CONFIG.SYS e AUTOEXEC.BAT actuais para um local seguro antes de correrem o CONFIG.BAT, pois este copia logo as novas configurações para a raiz do disco, alterando os anteriores.

Hugo Meneses

DISQUETAS BULK (sem marca)

10 DISQUETAS 3,5" DS/DD
Ref. D040 Preço: 1 956\$00

10 DISQUETAS 3,5" DS/HD
Ref.. D041 Preço: 1 243\$00

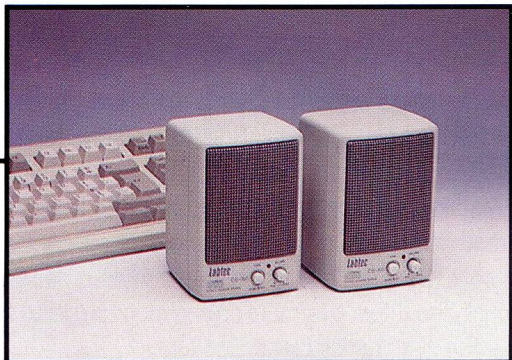
«RATO» SEM FIOS Genius HiMouse



- **Liberdade de movimentos:** pode usá-lo até 2 metros do computador sem quaisquer ligações com cabos.
- **Tecnologia de infra-vermelhos:** sem interferência em sinais rádio
- **Poupa energia:** Desliga-se automaticamente quando não em uso. A unidade receptora possui sempre 2 baterias adicionais recarregadas e prontas a usar.
- **Software de primeira:** inclui o extraordinário ZSoft Paintbrush IV
- **Inclui:** Adaptador 9/25 pinos, unidade receptora, 3 baterias recarregáveis, disquete de instalação e ZSoft Paintbrush IV.

Ref. RSF1

Preço: 14 600\$00

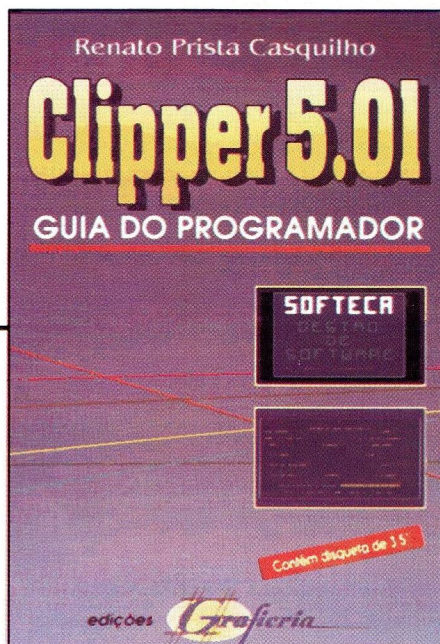


SISTEMA DE AMPLIFICAÇÃO COM ALTIFALANTES PARA COMPUTADOR

10% de desconto
para assinantes em todos os produtos da Montra*

TODOS OS PREÇOS APRESENTADOS JÁ INCLUEM O IVA.
(Livros: 5%, Restantes produtos: 16%)

* Para calcular o desconto de assinante retirar primeiro o IVA (dividir o preço indicado por 1,16 ou 1,05 nos livros), depois retirar 10% e tornar, finalmente, a adicionar o IVA.



Clipper 5.01 – Guia do Programador Renato Prista Casquilho

Este livro pretende ir ao encontro das preocupações e dificuldades encontradas por quantos pretendem iniciar-se em qualquer linguagem de programação. Destina-se, essencialmente, aos programadores de dBASE que pretendem dar o «salto» qualitativo para o poderoso ambiente facultado pelo Clipper 5.01, capaz de gerar aplicações sofisticadas, elegantes, rápidas e autónomas. Nele são analisados e exemplificados todos os comandos, funções e declarações, incluindo as novidades e correcções introduzidas pela versão 5.01.

A obra é acompanhada por uma disquete de 3 1/2", cujo conteúdo constitui uma demonstração das inúmeras capacidades desta linguagem.

Ref. GR03

Preço: 3 600\$00

O melhor som estéreo para o seu PC

- Sistema de amplificação para Computador.
- Estereofónico com controlo individual de volume
- Alimentação interna por baterias 4 «C» (não incluídas) ou externa por DC 6V.
- Funciona com ou sem placas de som..
- Rated Output Power: 7,2 W 3,6 W x 2 P.O.P.M.
- Frequência resposta: 100-15 000Hz
- Peso (sem baterias): 355 g x 2
- Dimensões: 85 x 120 x 90 mm

Ref. CL01
Preço: 6 000\$00

FREESOFT/SHAREWARE

Spooler

MAGAZINE DOS UTILIZADORES PC

O ficheiro FREE-S23.TXT na disquete que acompanha esta revista contém a descrição de todo o software disponível.

TAPETE PARA MOUSE

Tapete para Mouse de elevada qualidade. Anti-estático. Permite-lhe uma maior precisão e facilidade no seu trabalho.

Ref. G05R TAPETE RÍGIDO PREÇO: 415\$00
Ref. G05F TAPETE FLEXÍVEL PREÇO: 456\$00

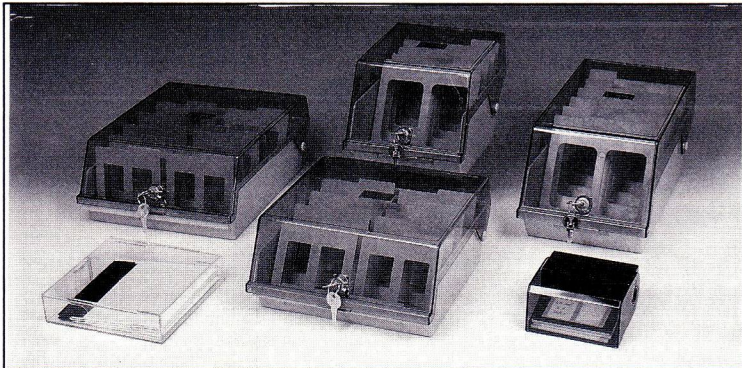
LIVROS EDITORIAL O LIVRO (Em português)

- INTRODUÇÃO À INFORMÁTICA E COMPUTADORES
REF.: L001 PREÇO: 2 000\$00

- TÉCNICAS DE PROGRAMAÇÃO
REF.: L002 PREÇO: 1 750\$00

- INTRODUÇÃO AO BASIC
REF.: L003 PREÇO: 2 600\$00

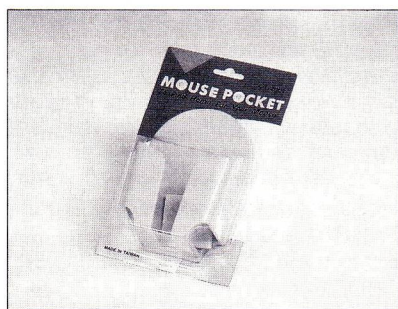
CAIXAS ARQUIVADORAS PARA DISQUETAS



- 10 disquetes 3,5"	Ref.: H004	Preço: 214\$00
- 10 disquetes 5,25"	Ref.: H005	Preço: 320\$00
- 50 disquetes 5,25" c/ chave	Ref.: H006	Preço: 986\$00
- 100 disquetes 5,25" c/ chave	Ref.: H007	Preço: 1 177\$00
- 50 disquetes. 3,5" c/ chave	Ref.: H010	Preço: 804\$00
- 100 disquetes. 3,5" c/ chave	Ref.: H012	Preço: 1 011\$00

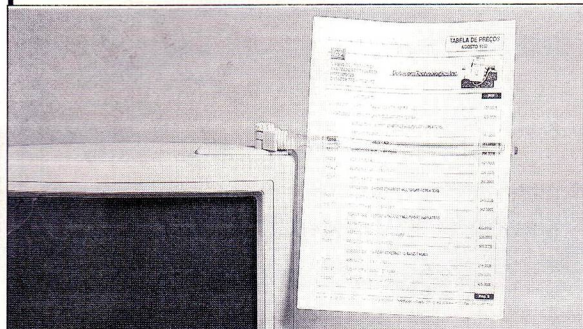
SUORTE PARA RATO

Não abandone o seu rato quando não precisa dele. Esta caixa foi especialmente concebida para acondicionar a «fera». Colocação com pastilha autocolante.



Ref. PR01
PREÇO: 750\$00

PORTA RASCUNHOS



Útil dispositivo para segurar documentos perto do seu campo de visão.

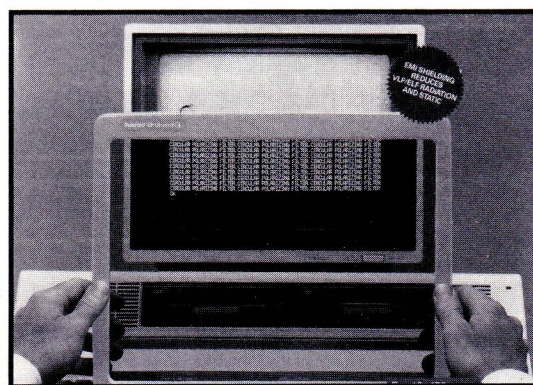
Ref. PR01 PREÇO: 1 160\$00

DBase III Plus PROGRAMAÇÃO AVANÇADA Renato Casquilho

Ref. GR01 COM DISQUETA DE 5,25" 1 800\$00
Ref. GR02 COM DISQUETA DE 3,5" 1 850\$00

INVISUAIS: Quando solicitado forneceremos em disquete suplementar o texto integral do livro.

FILTROS POLAROID PARA MONITORES



Evite a fadiga visual utilizando um filtro de alta qualidade no seu monitor. Os filtros **Polaroid** foram especialmente estudados para proteger a sua saúde ocular sem prejudicar a nitidez e definição da imagem do seu monitor. Os filtros **Polaroid** obedecem à norma 90/270/CEE. No cupão de encomenda não se esqueça de referir o formato do seu monitor (de 10 a 21"). Em vidro óptico ou acrílico.

Ref. FM01 Vidro óptico PREÇO: 22 375\$00
Ref. FM02 Acrílico PREÇO: 13 260\$00

MODEMS BEST DATA



Modems de alta qualidade a preços notáveis.

Ref. BD01	Modem interno – 2400 bps W/MNP 2-7 por software V.22 e V.22 bis)	Preço: 16 423\$00
Ref. BD02	Modem externo – 2400 bps W/MNP 2-7 por software V.22 e V.22 bis)	Preço: 23 363\$00
Ref. BD03	Fax/Modem interno – 9600 bps Fax – 2400 bps Modem W/MNP 7 por software V.22 e V.22 bis)	Preço: 27 360\$00
Ref. BD04	Fax/Modem externo – 9600 bps Fax – 2400 bps Modem W/MNP 7 por software V.22 e V.22 bis)	Preço: 41 694\$00

Nota: estes Modems aguardam registo de homologação

SOFTWARE IBER

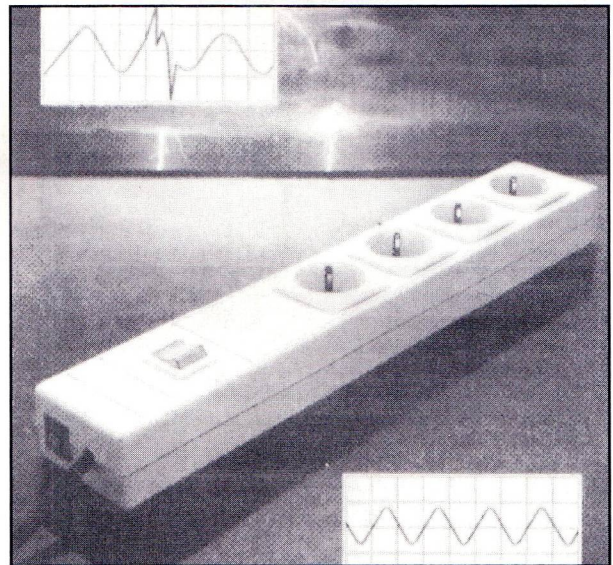
Programas e soluções profissionais para compatíveis MS-DOS.
Não requerem conhecimentos de Lotus 1-2-3 ou de MS-DOS.

Refª IB01*	CAPITAL-P – FINANÇAS PESSOAIS.
Refª IB02*	CAPITAL-E – TESOUREARIA DE PEQUENA E MÉDIA EMPRESA.
Refª IB03	CORREIO-E – REGISTO E CONTROLO DE CORRESPONDÊNCIA..
Refª IB04*	BALANÇOS – ANÁLISE DE BALANÇOS.
Refª IB05	BIBLOS – GESTÃO DE LIVROS, VIDEOS E DISCOS.
Refª IB06	CATÁLOGO – APOIO A COMPRAS – PRODUTOS – FORNECEDORES – PREÇOS
Refª IB07	CAPITAL-W – GESTÃO DE TESOUREARIA VERSÃO PARA WINDOWS

Em disquetes de 3,5" (720 Kb) PREÇO: 4 806\$00

*Versões para Lotus 1-2-3 – Rev 2.10 ou mais recente

PROTECTOR DE TENSÃO PRIMAX-5734

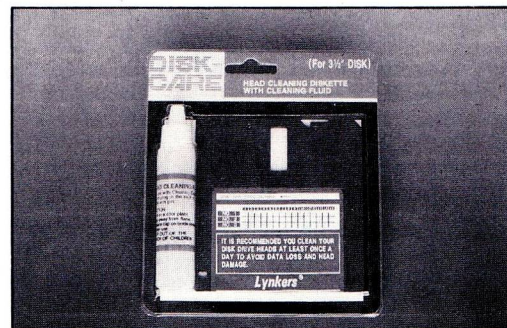


Este é o primeiro e mais importante passo para a protecção do seu equipamento. Esta «caixa de tomadas» elimina os picos temporários da alimentação e os ruídos dos circuitos de luz eléctrica (filtro EMI/RFI).

Ref. PR01

Preço: 11 352\$00

KITS PARA LIMPEZA DE DRIVES



Kits de limpeza de drives incluindo disquete e líquido de limpeza. Qualidade comprovada.

Ref. G001	Disquete 3 ½	365\$00
Ref. G002	Disquete 5 ¼	365\$00

PAPEL CONTÍNUO

– 500 folhas A4 (zebra americana) 12x9,5"		
Refª. P020	Preço:	862\$00
– 500 folhas A4 – 3 vias – 12x9,5"		
Refª. P021	Preço:	18 229\$00
– 1000 folhas A4 – 2 vias – 12x9,5"		
Refª. P022	Preço:	11 766\$00
– 2000 folhas A3 – 12x15"		
Refª. P023	Preço:	6 546\$00

QUICKJOY M-5



Compatível para PC/XT/AT este joystick de *design* ergonómico é o ideal para os seus jogos de acção. A placa pode ser adquirida em separado.

Refª G004

Preço: 3 812\$00

JOYSTICK + PLACA SV 203



Joystick para compatíveis PC/XT/AT e placa. Ótimo desempenho e *design* atractivo. Um novo dinamismo nos seus jogos predilectos. Um preço incrível!!

Ref. G003

Preço: 6 960\$00

DISQUETAS ATLANTIC MAGNETICS

Ref. AM01 Disquetas 3,5" DS/DD – Cx. 10 unid. 1 823\$00

Ref. AM02 Disquetas 3,5" HD – Cx. 10 unid. 2 438\$00

PLACA QS-120 PARA JOYSTICK

Placa para joystick

Ref. G04A

Preço: 3 646\$00

RATO MIC - MI-260



O rato MIC é um *best-seller!* Tão bom como os melhores e o mais barato de todos!

Características: 260-2600 DPI, Compatível Microsoft e Mouse System, com comutador no corpo do rato. 3 teclas.

Inclui: Software de instalação e teste, ficheiros de ajuda (em inglês).

Refª MIC1

Preço: 2 652\$00

FITAS, TONNERS E TINTEIROS PARA IMPRESSORAS

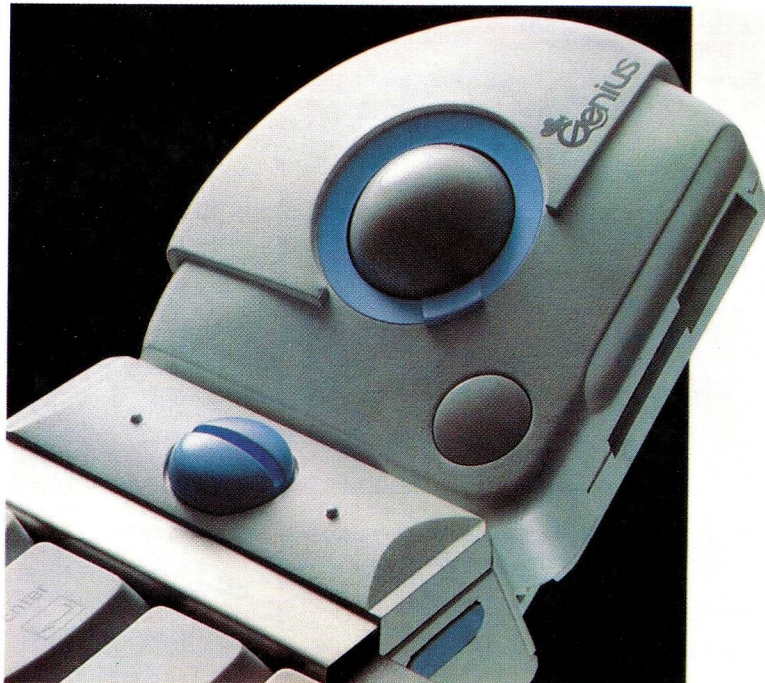
PARA TODOS OS TIPOS DE IMPRESSORAS (E MÁQ. DE ESCRIVER)
MARCAS ORIGINAIS E COMPATÍVEIS.

Indique no cupão de encomenda a referência que pretende (ou a marca e modelo da impressora)

TEMOS OS MELHORES PREÇOS.

HIPPOINT MOUSE

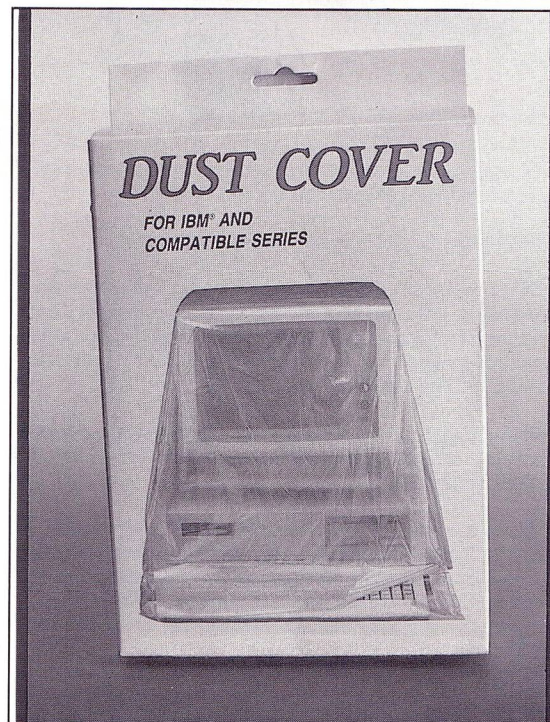
O «rato» ideal para portáteis e «notebook»



HiPoint foi especialmente concebido para equipar todos os modelos de computadores portáteis e «notebook». Com instalação extremamente fácil, o HiPoint oferece total liberdade relativamente à sua colocação no corpo do portátil e à orientação dos botões. Para todos os sistemas compatíveis IBM e PS/2 e todas as aplicações incluindo Windows. Claro que o HiPoint também funciona em computadores não portáteis.

Ref.: HP01 13 423\$00

CAPA PROTECTORA PARA PCs E IMPRESSORAS



Capa protectora para computadores em plástico especial reforçado para a completa protecção do seu equipamento contra fumos, poeiras, humidade e sujidades de toda a espécie. Adaptável a qualquer tipo de PC e impressoras.

Ref.: G006 – Capa para PC 1 823\$00
 Ref.: G007 – Capa p/ imp. 80 c. 995\$00
 Ref.: G008 – Capa p/ imp. 132 c. 995\$00



Recorte ou fotocopie o cupão e remeta-o para:

SPOOLER MAGAZINE
 Rua Duarte Pacheco Pereira, 69-A – Damaia
 2700 AMADORA

ESTIMADO LEITOR:
 Tentamos entregar as suas encomendas o mais rápido possível. Quando um número da Spooler sai ao público, somos «bombardeados» com muitos milhares de pedidos. Não desespere se não receber a sua encomenda no dia seguinte e **por favor não a devolva**. Os preços por nós praticados (quase isentos de lucro) não permitem o encargo adicional de ter que pagar os portes do retorno da encomenda. Comunique-nos se pretender, à posteriori, desistir da encomenda.

CUPÃO DE ENCOMENDA – MONTRA SPOOLER

Cliente n.º: _____

A PREENCHER PELOS NOSSOS SERVIÇOS

Desejo receber à cobrança (com os custos acrescidos de portes e taxas) os seguintes produtos:

Quant.	Ref.º	Designação	Preço

TOTAL _____

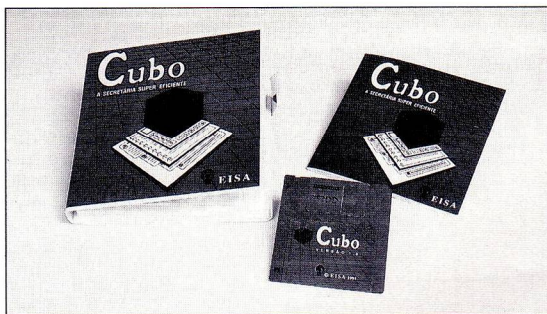
NOME _____

MORADA _____

C. POSTAL _____ LOCALIDADE _____

TELEFONE _____ NÃO SOU ASSINANTE SOU ASSINANTE N.º _____

CUBO



CUBO pretende ser um produto simples e funcional vocacionado para uso pessoal ou utilização em escritório.

O **CUBO** é constituído por três módulos principais:

- **Gestor de Informação.** Para armazenar qualquer tipo de informação, pessoal, da empresa, de colecções, de arquivos, normas, dados técnicos, permitindo a sua classificação (catalogação) por temas, secções e subsecções. Facilmente poderá proceder a pesquisas, modificar, inserir ou apagar dados.
- **Mailing.** Esta opção possibilita a produção de documentos, utilizando um processador de texto, e remete-los a destinatários constantes da opção **Endereços**.
- **Gestão de contas.** A sua conta bancária, o movimento do cartão de crédito, a contabilidade doméstica ou o caixa da sua empresa, podem ser geridos em simultâneo utilizando esta opção. Extractos das contas, configuráveis pelo utilizador, e emissão de cheques em formulário, com o respectivo lançamento em contabilidade, são outras das várias opções disponíveis.
- **Utilitários.** Para completar, o utilizador pode dispor dos seguintes utilitários: Calculadora, Calendário, Agenda, Bloco de Notas e Endereços.

Com Manual. Tudo em português.

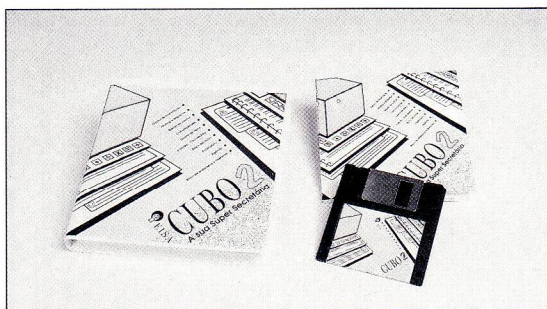
Requisitos mínimos de hardware: Processador 8086; Placas CGA, Hércules, EGA ou VGA; 640 Kb de RAM, disco de 20 Mb; MS-DOS 3.2.

CUBO e **CUBO 2** foram integralmente concebidos em Portugal pela EISA, Lda. Mais de 14 500 cópias vendidas!

Ref. CB01

Preço: 6 960\$00

CUBO 2



O **CUBO 2** é uma versão avançada e ainda mais poderosa e versátil do **CUBO**.

Além de todas as potencialidades descritas para o **CUBO**, o **CUBO 2**, que corre em modo residente (TSR), apresenta, para além de vários melhoramentos gerais, as seguintes características:

- Capacidade para operar com tantas janelas no «Gestor de Informação» quanto a capacidade do seu sistema o permitir;
- Possibilidade de emitir etiquetas, de formatos pré-configurados ou a parametrizar pelo utilizador;
- Módulo de Comunicações. Através de um *Modem* você poderá transmitir/receber dados para ou de outros sistemas, podendo mesmo fazer «chat» ou colocar-se em modo «host»;
- Gestor de disco, para todas as operações de manipulação de ficheiros do seu disco.

Ref. CB02

Preço: 21 750\$00

PROGRAMA PARA FACTURAÇÃO «FACTURA»

Este programa de fácil utilização, instalação simples, textos de ajuda claros e precisos, menus *pull-down*, obedece a todos os requisitos legais em vigor e permite-lhe:

- Criar e manter actualizados dados sobre clientes e artigos;
- Imprimir listagens de clientes e artigos de acordo com variadas opções de selecção;
- Manter actualizado o movimento e inventário de stocks;
- Produzir e imprimir propostas, facturas, notas de crédito, notas de débito e guias de remessa.
- Contas correntes, tabelas de preços por artigos e clientes.

OBEDECE A TODAS AS DISPOSIÇÕES LEGAIS EM VIGOR

Com garantia e assistência técnica ON-LINE. Produzido por XT-Software, Lda.

Ref. FT01

Preço: 17 400\$00

PROGRAMA DE GESTÃO CONTABILÍSTICA «MAXICONTA»

MAXICONTA é um programa para Contabilidade, projectado para o utilizador que necessite de aplicações simples e eficazes.

PRINCIPAIS CARACTERÍSTICAS:

- Criação e actualização de contas por centros de custo, diários, contas e tipos de lançamento.
- Introdução de lançamentos e listagens por código; data; extractos; centro custo; tipo lançamento; diário; n.º de documento.
- Balançetes do Razão; Geral; Centro Custo; Tipo Lançamento.
- 12 meses em aberto.
- Balanço geral.
- Abertura e fecho de ano.
- Utilitários diversos incluindo formatação de disquetes, cópias de segurança, reposições, indexação recálculo e meses abertos.

OBEDECE A TODAS AS DISPOSIÇÕES LEGAIS EM VIGOR

Com garantia e assistência técnica ON-LINE. Produzido por XT-Software, Lda.

Ref. MX01

Preço: 23 200\$00

«AGENDA»

AGENDA é a solução para quem pretender organizar e rentabilizar o seu tempo.

PRINCIPAIS CARACTERÍSTICAS:

- **Lista de endereços e telefones:** Número ilimitado de fichas contendo informações diversas sobre empresas ou personalidades.
- **Agenda:** Marcação e visualização de entrevistas, chamadas telefónicas e tarefas.
- **Contas bancárias:** Simples mas eficiente controlo de todas as suas contas bancárias, incluindo gestão de cheques pré-datados e letras.
- **Editor de texto:** Faça os seus próprios documentos, sem necessidade de processadores de texto convencionais. Ideal para mailings, cartas, etc.
- **Etiquetas:** Emissão de etiquetas a partir da base de dados de endereço e com selecções variadas.
- **Plano de tesouraria:** Controlo de pagamentos e recebimentos.
- **Utilitários:** Calculadora, calendário, bases de dados de utilidade pública.

Com garantia e assistência técnica ON-LINE. Produzido por XT-Software, Lda.

Ref. AG01

Preço: 13 920\$00

ASSINATURAS

Spooler

MAGAZINE DOS UTILIZADORES PC

Recorte ou fotocopie o cupão, preencha-o com letra bem legível e remeta-o, acompanhado do respectivo pagamento em cheque ou vale de correio, para:

SPOOLER MAGAZINE

R. Duarte Pacheco Pereira, 69-A
Damaia - 2700 AMADORA

Os números anteriores da Spooler podem ser encomendados à cobrança utilizando o cupão da Montra Spooler. Ao preço de capa serão acrescidos os custos dos portes e taxas de cobrança. Se preferir disquetes de 3,5", isto custar-lhe-á mais 140\$00 por exemplar.

NOTA: Os números 1 a 6 e o 10 e 11 da Spooler encontram-se esgotados.

CUPÃO DE ASSINATURA

Assinante n.º:

A PREENCHER PELOS NOSSOS SERVIÇOS

NOME

DATA DE NASCIMENTO* ____/____/____

PROFISSÃO*

TRABALHA COM COMPUTADORES?* SIM NÃO

SE SIM ESPECIFIQUE A MARCA E MODELO*

ENDEREÇO

C. POSTAL LOCALIDADE TELEF(S)

DESEJO ASSINAR A SPOOLER A PARTIR DO N.º 24 (inclusive) E DURANTE 12 NÚMEROS.

- MODALIDADE COM DISQ. DE 5 1/4" 6 000\$00
- MODALIDADE COM DISQ. DE 3 1/2" 6 000\$00

PARA PAGAMENTO DA ASSINATURA ENVIO:

- O CHEQUE N.º S/ O BANCO
- O VALE DE CORREIO N.º

* A resposta aos itens assinalados com um (*) é opcional.

REEDIÇÃO DOS NÚMEROS ESGOTADOS (1 a 9)

Mais tarde ou mais cedo, reeditaremos todos os números anteriores da Spooler
– a única revista portuguesa de informática* que esgota as suas edições, reedições e (re)reedições.
Reserve já o seu exemplar.

* Em boaverdade não conhecemos mais nenhum caso de revista, portuguesa ou estrangeira, de informática ou de qualquer outro assunto, que proceda a reedições. Mas também não conhecemos tudo...

Depois de esgotada a reedição dos números 1, 2 e 3 da Spooler e para satisfazer a procura manifestada por inúmeros leitores tornámos a imprimir os números 1, 2 e 3, agrupados num só volume. Esta edição teve uma tiragem reduzida e está quase, quase esgotada.

A reedição 4-5-6 esgotou. Inscreva-se para futura re(edição).

O único modo de possuir um exemplar destas reedições é reservá-las, desde já, preenchendo o cupão anexo (ou fotocópia) e remete-lo para os nossos serviços:

SPOOLER MAGAZINE

Rua Duarte Pacheco Pereira,
69-A - Damaia-de Baixo
2700 AMADORA

CUPÃO DE RESERVA DAS EDIÇÕES ESPECIAIS SPOOLER 1-2-3, 4-5-6, 7-8-9

NOME

ENDEREÇO

C. POSTAL LOCALIDADE TELEF.

DESEJO RECEBER A EDIÇÃO ESPECIAL

- «SPOOLER 1 - 2 - 3»
- «SPOOLER 4 - 5 - 6»
- «SPOOLER 7 - 8 - 9»

- COM 3 DISQUETAS DE 5 1/4" 1 200\$00
- COM 2 DISQUETAS DE 3 1/2" 1 200\$00

Não mande qualquer pagamento agora. A edição especial ser-lhe-á enviada à cobrança.
Aos preços indicados serão acrescidos os portes de envio e taxa de cobrança dos C.T.T.

Spooler-24

Assinatura

TINTEIRO + 3 RECARGAS PARA COMMODORE MPS 1270



Tinteiro + 3 recargas Mills. Produto de alta qualidade e grande economia para o utilizador.

Ref.: TN01

8 785\$00

Disponíveis tinteiros para todas as marcas e modelos de impressoras. Diga-nos o modelo da sua impressora. O preço será sempre o melhor possível.

GS-FC60 – Color Scanner



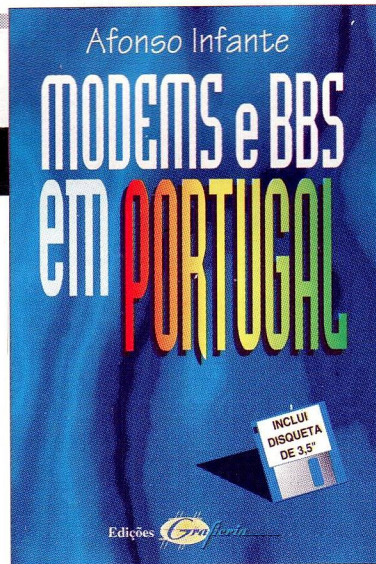
Detectando mais de 16 milhões de tonalidades de cor, este scanner de alta qualidade a um preço incrivelmente baixo, permitir-lhe-á «recolher» imagens até 8,5 x 14" (216x356 mm) a cores, em tons de cinzentos ou a preto e branco. O conjunto inclui o software para recolha e tratamento de imagem **iPhoto Deluxe** e também um programa de reconhecimento óptico de caracteres (**OCR**), o **WordScan** que converte imagens de texto em caracteres utilizáveis.

Principais características:

- **Resolução:** até 600 dpi
- **Velocidade de scanning:** Cor – 120 seg./A4; Gama cinzentos – 40 seg./A4; P/B – 9,0 seg. A/4

Ref.: HP02

279 125\$00



Com Disquete de 3,5" com o programa de comunicações shareware **TELEMATE V4.00**

BBS? Modem? Programas de Comunicações? Estes são os temas que o autor desenvolve ao longo das 128 páginas do livro. Temas da maior actualidade dada a popularização do computador pessoal e do interesse dos utilizadores em alargar as capacidades das suas máquinas.

Na sua introdução, o autor começa por colocar uma questão bem pertinente: «E, já agora, o que é uma BBS?» Pertinente, porque para muitos a resposta será uma total revelação; para outros um manancial de esclarecimentos; para alguns, um desfazer de dúvidas ou de conceitos totalmente errados.

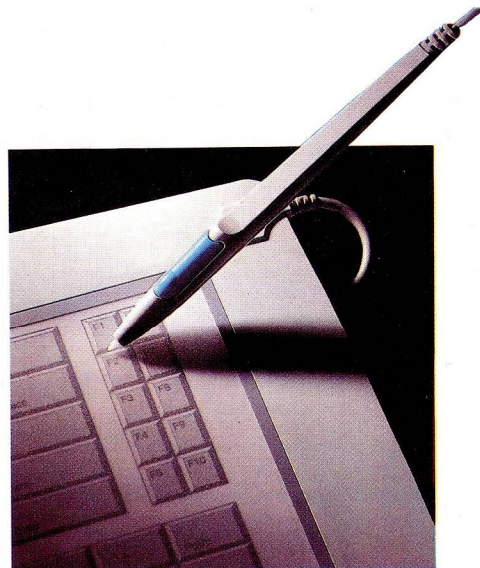
O leitor menos experiente será confrontado com uma terminologia e um «calão» totalmente desconhecidos. Um completo Glossário ajuda-lo-á a familiarizar-se com esta nova «linguagem». Ser-lhe-hão apresentados os diferentes tipos e classes de modems, o que lhe permitirá fazer uma escolha judiciosa. A configuração do modem é tratada com a necessária profundidade, sendo facultados inúmeros exemplos práticos. A partir de uma lista das principais BBS portuguesas e de uma aprendizagem «passo-a-passo», aprenderá a fazer a sua primeira ligação com a máxima garantia de sucesso. Os principais programas de comunicações são analisados em pormenor. Os «Protocolos» de transferência de ficheiros, os compressores, as «Transferências Avançadas» e o «Correio Electrónico» merecem especial destaque em capítulos próprios.

Finalmente, encontrará na disquete que acompanha o livro um excelente programa de comunicações shareware: o **TELEMATE V4.00**, ao qual é dedicado um extenso parágrafo do Capítulo 7 – «Os primeiros passos».

Ref.: GR04

2 400\$00

HiPen – A melhor invenção depois do «Mouse»



De facto, **HiPen** é um produto revolucionário. Este dispositivo permite-lhe trabalhar como sempre o fez: caneta sobre o papel. Substituindo ao mesmo tempo o teclado e o rato, **HiPen** funciona como uma vulgar caneta usando a sua letra manuscrita! Você pode «ensinar» ao **HiPen** o seu estilo de escrita que este será reproduzido como texto, como se os caracteres provenissem do teclado! Um adaptador de tinta permitir-lhe-á mesmo escrever sobre papel, possibilitando um maior rigor no «traçamento» de imagens. A embalagem do **HiPen** inclui software para o Windows 3.1, base para escrita e o fantástico **ZSoft PC Paintbrush IV!!!**

Ref.: HP03

61 066\$00

J&L Designers



**FACILITA A VIDA
NO SEU ESCRITÓRIO**



systemtronic

contacte: 937 78 40 • 937 79 40

Graficaria

Publicidade e Artes Gráficas, Lda.

Rua Duarte Pacheco Pereira, 69-A 2700 AMADORA
Telef.: 474 60 11/474 60 41 Fax: 475 22 89

Consumíveis para Informática

Somos especialistas em:

- **Duplicação de Software**
(Possuimos equipamentos específicos)
- Diskettes personalizadas
- Fitas para impressoras
(originais ou compatíveis)
- Tinteiros para impressoras
de Jacto de Tinta
- Tapetes e suportes para
«ratos»
- Suportes para documentos
no monitor
- Caixas arquivadoras
para diskettes
- Etiquetas em contínuo ou
folhas soltas para laser
- Rolos para Fax
- Toners para Lasers
- Toners para fotocopiadoras
- Capas para CPU, teclados
e impressoras
- Consumíveis para **Hewlett Packard**
- Filtros **Polaroid** para ecrãs
- Data, Tape Cartridges
- Bandas Magnéticas



LM ELECTRÓNICA DE
CONSUMO, LDA.

Rua Barão Sabrosa, nº 31 - C - 1900 LISBOA
Telef.: 814 63 74 - Fax: 814 17 96