

Departamento de Engenharia Informática  
Faculdade de Ciências e Tecnologia  
Universidade de Coimbra



# Introdução à Informática

Paulo Fernando Pereira de Carvalho

Jorge Sá Silva

# Índice

<b>CAPÍTULO I INTRODUÇÃO À TECNOLOGIA INFORMÁTICA .....</b>	<b>1</b>
1. INTRODUÇÃO .....	2
1.1 <i>Eventos Históricos no Desenvolvimento do Computador</i> .....	2
1.2 <i>As Primeiras Gerações de Computadores</i> .....	6
1.3 <i>Os PCs</i> .....	9
1.4 <i>A Evolução Futura</i> .....	10
2. NOÇÕES DE INFORMAÇÃO BINÁRIA.....	10
2.1 <i>Utilização da Codificação Binária</i> .....	11
2.2 <i>A Codificação Binária de Números Inteiros</i> .....	13
2.3 <i>O Código ASCII</i> .....	15
2.4 <i>Noções de Palavra e de Octeto</i> .....	16
3. ORGANIZAÇÃO DE UM COMPUTADOR DIGITAL .....	16
3.1 <i>A Memória Central</i> .....	18
3.1.1 <i>Tipos de Memória</i> .....	19
3.1.2 <i>As Memórias Cache</i> .....	22
3.2 <i>O Processador Central: A Unidade de Controlo</i> .....	23
3.2.1 <i>A Evolução dos processadores do IBM PC</i> .....	30
4. HARDWARE E SOFTWARE .....	34
5. BIBLIOGRAFIA .....	37
<b>CAPÍTULO II INTRODUÇÃO AO AMBIENTE WINDOWS .....</b>	<b>38</b>
1. INTRODUÇÃO .....	38
2. INÍCIO DA SESSÃO DE TRABALHO .....	39
2.1 <i>A Área de Trabalho</i> .....	39
3. EXECUÇÃO DE PROGRAMAS .....	41
4. COMUTAÇÃO ENTRE PROGRAMAS .....	44
5. UTILIZAÇÃO DO RATO EM WINDOWS.....	45
6. AS JANELAS.....	46
6.1 <i>Redimensionamento de Janelas</i> .....	49
6.2 <i>Deslocamento de Janelas</i> .....	50
6.3 <i>Os Menus</i> .....	51
6.4 <i>Os Comandos</i> .....	53
6.5 <i>Os MENUs Auxiliares</i> .....	54

6.6	<i>A Área de Trabalho</i> .....	55
6.7	<i>Os Diálogos</i> .....	57
6.7.1	<i>As Caixas de Texto</i> .....	57
6.7.2	<i>As Caixas de Listagem</i> .....	58
6.7.3	<i>Os Botões de Selecção (Radio Buttons)</i> .....	60
6.7.4	<i>Os Botões de Selecção (Check Box)</i> .....	61
6.7.5	<i>Outros Elementos de Comunicação</i> .....	62
7.	<i>PERSONALIZAÇÃO DO AMBIENTE DE TRABALHO</i> .....	63
8.	<i>FECHAR UMA SESSÃO WINDOWS</i> .....	69
9.	<i>PESQUISAR NA AJUDA</i> .....	70
<b>CAPÍTULO III CONFIGURAÇÃO DO AMBIENTE DE TRABALHO WINDOWS 2000</b> .....		<b>73</b>
1.	<i>INTRODUÇÃO</i> .....	73
2.	<i>O CONTROL PANEL</i> .....	73
2.1	<i>Execução de uma Ferramenta do Control Panel</i> .....	77
3.	<i>ALTERAÇÃO DA DATA E DA HORA DO SISTEMA</i> .....	77
4.	<i>O TECLADO</i> .....	78
4.1	<i>Definição do Modelo e Língua do Teclado</i> .....	80
5.	<i>CONFIGURAÇÃO DO RATO</i> .....	82
5.1	<i>Definir a Função dos Botões do Rato</i> .....	82
5.2	<i>Definição da Forma dos Apontadores</i> .....	83
5.3	<i>Configuração do Movimento do Rato</i> .....	85
6.	<i>CONFIGURAÇÕES REGIONAIS</i> .....	86
6.1	<i>Definição do País</i> .....	87
6.2	<i>Configuração do formato dos números</i> .....	87
6.3	<i>Configuração dos valores Monetários</i> .....	90
6.4	<i>Configuração da Data</i> .....	90
6.5	<i>Configuração do Formato da Hora</i> .....	91
7.	<i>CONFIGURAÇÃO DE ASSOCIAÇÃO DE SONS</i> .....	92
8.	<i>GESTÃO DE IMPRESSORAS</i> .....	95
8.1	<i>Instalação de uma Nova Impressora</i> .....	96
8.2	<i>Instalação de Impressoras em Rede</i> .....	101
9.	<i>INSTALAÇÃO DE SOFTWARE NOVO</i> .....	101
9.1	<i>Remoção/Alteração de um Programa Instalado</i> .....	102

# CAPÍTULO I

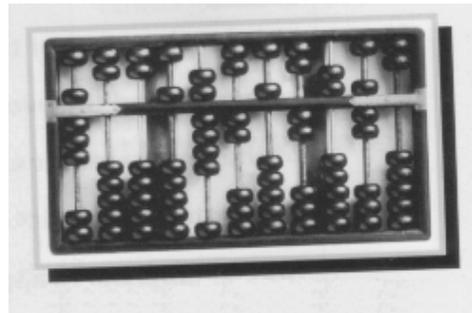
## Introdução à Tecnologia Informática

*Neste capítulo serão introduzidos alguns conceitos básicos de tecnologia informática. Estes conceitos serão complementados por descrições do estado da arte em algumas áreas dos componentes dos computadores, facultando, deste modo, ao leitor alguma informação fundamental para a escolha de um sistema informática.*

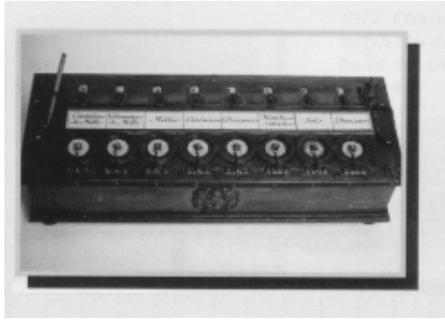
---



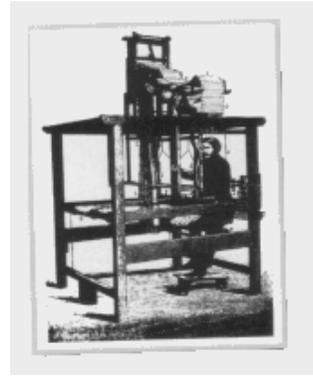
*4000-1200 A.C.: Tabelas de barro para registar transacções comerciais.*



*3000 A.C.: Os ábacos são inventados na Babilónia.*



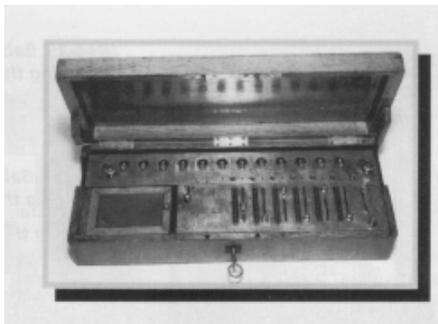
1642: Blaise Pascal inventa uma máquina mecânica para realizar adições: a "Pascalene".



1801: Um conjunto de cartões perfurados controlam os padrões na máquina de tecelagem de Joseph-Marie Jacquard.

## 1. Introdução

O computador moderno representa seguramente um avanço tão significativo como a introdução do zero no sistema de numeração e a descoberta do cálculo. O computador é simultaneamente o resultado e um dos maiores contribuintes da explosão tecnológica a que assistimos hoje em dia, verificando-se que praticamente ninguém "escapa" na sua vida quotidiana ao contacto, de uma forma directa - muitos postos de trabalho exigem conhecimentos de informática na óptica do utilizador, ou indirecta - desde as simples facturas processadas por computador, passando pelo sistema MultiBanco até às mais sofisticadas análises clínicas, com este tipo de equipamentos. Dado que o presente e o futuro muitas vezes só se entendem pela análise do passado, ir-se-á nas próximas secções deste capítulo realizar uma breve descrição histórica da evolução do computador.



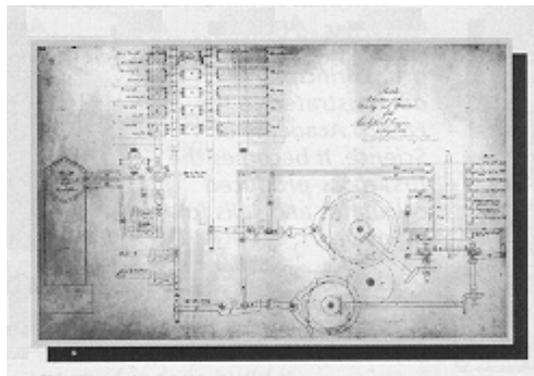
1820: Thomas desenvolve a primeira máquina de calcular mecânica a ser comercializada.

## ***Desenvolvimento do Computador***

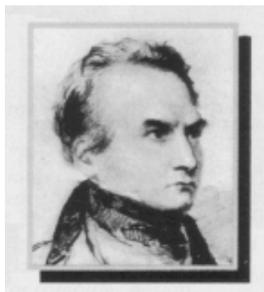
Foram vários os desenvolvimentos científicos que precederam o surgimento do computador moderno na década de 40. Em 1812, Charles Babbage, professor de matemática na Universidade de Cambridge, desenvolveu uma máquina (*difference engine*) capaz de realizar de forma automática computações simples necessárias à elaboração de tabelas logarítmicas e trigonométricas. O *difference engine* de Babbage executava uma sequência de operações, uma de cada vez, como acontece na maioria dos computadores actuais. Babbage projectou ainda uma máquina analítica, o *analytical engine*, mais complexa que deveria permitir realizar uma sequência arbitrária de operações e com capacidade para armazenar dados. Contudo, esta máquina nunca chegou a ser construída por falta de financiamento. O conceito de salvaguarda de dados introduzidos na máquina analítica de Babbage assemelha-se ao modo como os programas são armazenados nos computadores modernos.

Os computadores digitais são compostos por redes de comutação extremamente complexas. A lógica booleana, nomeada segundo o matemático George Bool, pioneiro na área da lógica simbólica, constitui um método sistemático de representação, análise e projecto destes circuitos. O

livro "*An Investigation of The Laws of Thought*", publicado em 1854 por George Bool,



1834-35: Babbage inicia o projecto do "*Analytical Engine*".



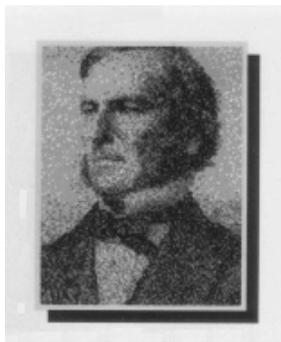
1822: Charles Babbage inicia a concepção do "*Differential Engine*".

representa a lógica de uma forma matemática por símbolos e descreve o conjunto de regras necessárias para aferir da veracidade ou falsidade de proposições. Um pequeno exemplo ilustrará a importância da lógica matemática na concepção dos computadores. Se um interruptor electrónico (os que controlam a rede de comutação) estiver aberto, é-lhe

atribuído o valor 0, caso contrário é-lhe atribuído o valor 1. Estes dois estados do interruptor podem ser interpretados respectivamente como os valores falso (0) e verdade (1) da lógica booleana. Imagine que dispõe de dois interruptores independentes, A e B, e que pretende realizar uma rede de comutação em que a saída (C) seja 1 somente quando ambos interruptores estão accionados. Este problema é facilmente solucionado pela expressão booleana:  $C=A \wedge B$  ( $C=A E B$ ), isto é, a saída só é verdadeira (1) quando o interruptor A E o interruptor B estão fechados em simultâneo. Os operadores básicos da lógica booleana são a negação

(ex: *NÃO A*), a conjunção (ex: *A E B*) e a disjunção ((ex: *A OU B*)).

O desenvolvimento de circuitos electrónicos capazes de implementar estes operadores permitiu o surgimento do computador digital. Como deverá imaginar, as expressões booleanas necessárias para a elaboração de um computador são extremamente complexas, cujo tratamento só é possível graças à existência de leis matemáticas bem definidas introduzidas por Bool, DeMorgan, Moore e outros matemáticos.



1854: George Boole publica o livro "An Investigation of the Laws of Thought" que descreve a lógica matemática.

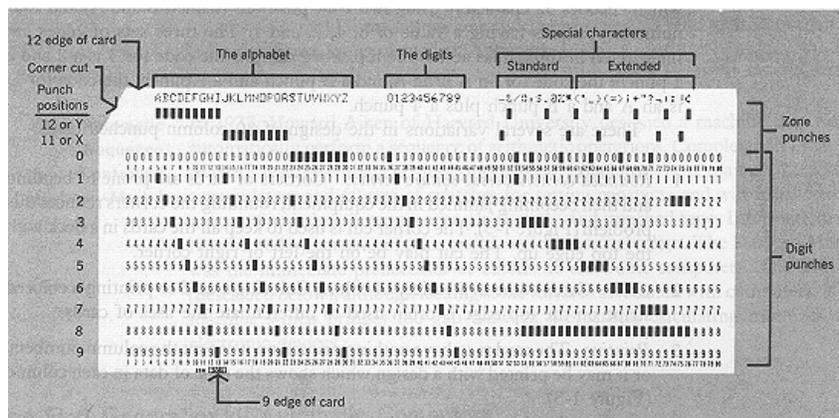


Figura 1: Um cartão perfurado de Hollerith.

Embora já não sejam elementos em uso nos computadores modernos, os cartões perfurados (vide figura 1) merecem aqui referência por duas razões: primeiro porque foram durante muito tempo o dispositivo privilegiado para a introdução de dados nos computadores e, por último, porque o seu desenvolvimento é em grande medida responsável pela fundação da empresa que orientou o desenvolvimento informático, a IBM. Em 1745, o Francês Joseph Jacquard desenvolveu um método para controlar máquinas de tecelagem baseado em cartões perfurados. Babbage propôs seguidamente a utilização dessa filosofia para a introdução de dados na evolução da sua “*difference engine*”, que nunca chegou a ser construída. Contudo, foi só em 1880 que Herman Hollerith desenvolveu um mecanismo de leitura de cartões perfurados eficaz, tendo a sua invenção, a “*census machine*”, sido aplicada com sucesso no censo Norte Americano de 1890. Em 1896 Hollerith funda a *Tabulating Machine Company* para produzir e comercializar a sua invenção. Esta empresa foi sucessivamente fundida com outras empresas, tendo dado origem à *International Business Machines Corporation* (IBM). O método desenvolvido por Hollerith foi seguramente uma das principais técnicas de introdução de dados nos computadores da primeira geração.



1889: O sistema eléctrico de contagem desenvolvido por Herman Hollerith afigura-se como o sistema ideal para o tratamento do *Census*.

## **1.2 As Primeiras Gerações de Computadores**

Em 1937, o Professor Howard Aiken da Universidade de Harvard projectou uma máquina que permitia, de forma automática, realizar uma sequência de operações aritméticas. Completada em 1944, a *Mark I* era essencialmente uma enorme calculadora mecânica composta por interruptores e fita perfurada para a introdução dos dados que eram representados internamente por um conjunto de relés mecânicos. O *Mark I* foi seguramente o predecessor imediato do computador automático electrónico.

O ENIAC (*Electronic Numerical Integrator and Calculator*), desenvolvido por J. Presper Eckert e John W. Mauchly da Universidade da Pennsylvania e completado em 1947, estabeleceu o próximo passo no desenvolvimento do computador electrónico automático representando a transição entre as calculadoras mecânicas e os dispositivos electrónicos. Durante o desenvolvimento do ENIAC, Eckert e Mauchly desenvolveram uma máquina maior, a EDVAC (completada em 1952), que apresentava duas diferenças fundamentais em relação ao ENIAC: a utilização de números binários nas operações aritméticas e a codificação das instruções internas do computador de forma digital. Esta máquina pode ser considerada o protótipo do computador sequencial (computador que executa somente uma instrução de cada vez) moderno. Durante este período desenvolvia-se na Universidade de Manchester o EDSAC, que é considerado o primeiro computador capaz de armazenar o programa a executar. Por volta de 1944, John Von Neumann, matemático Húngaro da Universidade de Princeton, inicia o desenvolvimento do computador IAS em colaboração com a Universidade de Pennsylvania. Embora só tenha sido terminado em 1952, o IAS introduziu os princípios arquitecturais (arquitectura de Von Neumann) do computador sequencial moderno. Ainda hoje são facilmente identificáveis nos PCs as componentes definidas por Von Neumann.

O primeiro computador comercializado foi o UNIVAC I produzido pela empresa Eckert-Mauchly, fundada em 1946 por Eckert e Mauchly. O primeiro UNIVAC foi instalado no *Bureau of the Census* em 1951. A primeira instalação UNIVAC para serviços comerciais foi realizada em 1954 para a *General Electric Appliance Park* de *Kentucky*, EUA. O nome UNIVAC foi durante muito tempo sinónimo de computador até a IBM, que inicialmente mostrou pouco interesse por esta tecnologia, ter decidido entrar no mercado da informática.

Os computadores acima mencionados representaram avanços importantes, quer ao nível conceptual, quer ao nível do projecto de computadores. Contudo, não foram os únicos equipamentos a serem projectados e montados durante este período. Após o ENIAC de 1947 surgiram em 1949 o EDSAC e o BINAC, em 1950 foram projectadas quatro novas máquinas, e em 1951, ano da primeira comercialização de um computador, seis novas máquinas. Em 1953 a IBM instalou o seu primeiro computador, o IBM 701. Mais tarde em 1954, a IBM instalou o primeiro computador 650 que foi o computador mais popular até 1959. Os computadores deste período são regra geral denominados “computadores da primeira geração”. Estes equipamentos caracterizam-se pela utilização de válvulas de vácuo, e por serem equipamentos gigantescos de baixas capacidades de cálculo (velocidade) e armazenamento.

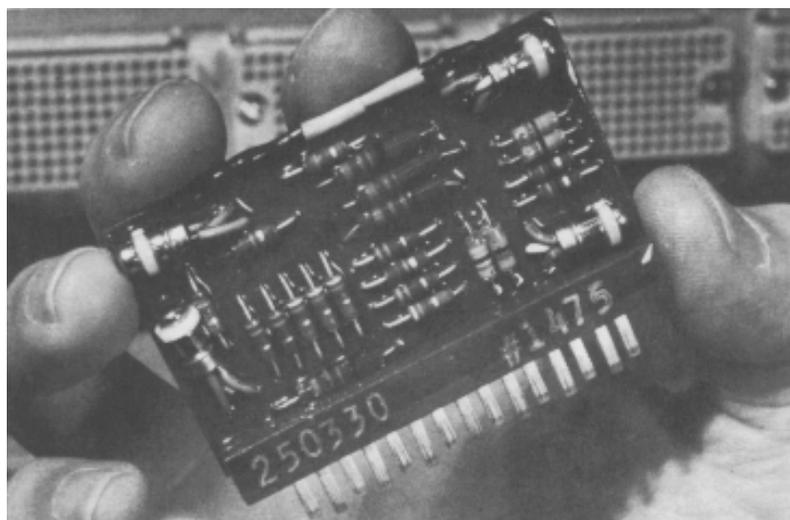


Figura 2: Circuito com transístores aplicado na construção de computadores (1960).

A invenção e aperfeiçoamento do transístor impulsionou significativamente o desenvolvimento do computador. Os transístores (vide figura 2) desempenham as mesmas funções que as válvulas de vácuo, com a vantagem de serem bastante mais pequenos, baratos, libertarem menos calor - os primeiros computadores necessitavam de poderosos sistemas de ar condicionado para funcionar, e necessitarem de menos energia para funcionarem. A transição de válvulas de vácuo para transístores iniciou-se nos computadores militares em 1956 e nos

computadores comerciais em 1959. Os computadores deste período (1959-1965) são regra geral denominados por computadores da segunda geração.

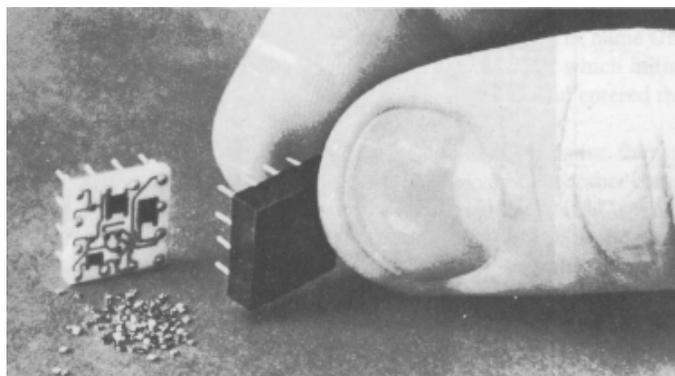


Figura 3: Transístor montado sobre uma base cerâmica com circuito impresso (1965).

A miniaturização dos componentes conduziu ao surgimento dos computadores da terceira geração (1965-1970). O início desta geração de computadores coincide com a introdução da família de computadores System /360 da IBM em 1965. Estes computadores caracterizam-se pela progressiva integração e miniaturização dos circuitos que os compõem (vide figuras 3 e 4).

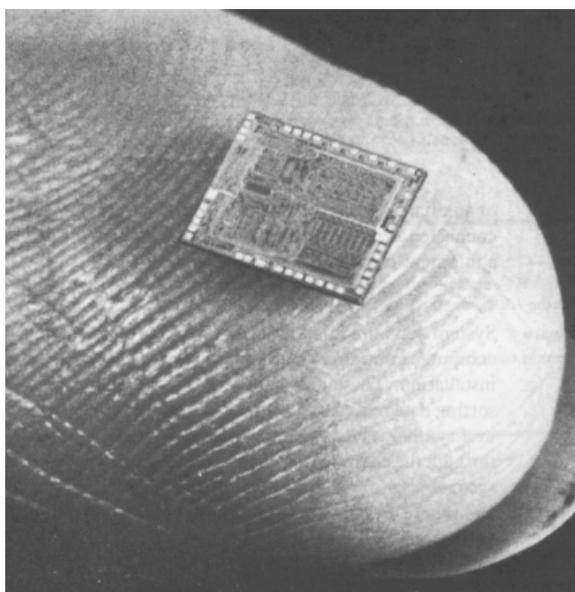


Figura 4: Circuito integrado aplicado na construção dos computadores da década de 1970.

### 1.3 Os PCs

Entre 1961 e 1971 o poder de integração do número de transístores duplica praticamente todos os anos. Em 1967 a capacidade de miniaturização permitiu à Fairchild o lançamento de integrados com 3800 transístores. Em 1971 a Intel introduz o seu primeiro microprocessador (processador constituído por um único integrado), o 4004, composto por 2300 transístores. O 4004 foi o primeiro processador comercialmente disponível. Em 1972 é introduzido o primeiro microprocessador de 8 *bits*: o 8008 da Intel composto por 3300 transístores. Pouco tempo depois surge o Intel 8080, o predecessor directo da família x86 - processadores utilizados no fabrico de computadores pessoais compatíveis IBM. Estes processadores não se destinavam inicialmente ao fabrico de computadores. Contudo, rapidamente deram origem a uma nova indústria que se dedicava ao fabrico de *kits* completos que permitiam, em casa, montar os primeiros computadores pessoais (computadores bastante rudimentares).

O primeiro computador pessoal completamente montado e funcional foi o Apple II que foi introduzido em 1977. Este sistema dispunha já de um ecrã, de um teclado e de uma unidade de leitura de disquetes e estabeleceu o padrão da indústria dos computadores pessoais até ao surgimento do IBM PC baseado no processador 8088 da Intel, em 1981. Em 1984 a Apple lança o Apple Macintosh baseado no processador MC68000, um processador bastante mais evoluído que o Intel 8088 e 8086 que equiparam os primeiros PCs da IBM. O Macintosh foi o primeiro computador pessoal a apresentar um sistema operativo com interface gráfica e baseado em janelas.

Se os computadores da Apple eram bastante mais evoluídos que os da IBM, como se explica que tenha sido a arquitectura proposta por esta última a estabelecer o padrão dos computadores pessoais? A IBM e Apple seguiram estratégias distintas na disputa do mercado dos computadores pessoais. Enquanto a IBM usou normas abertas na elaboração da arquitectura IBM PC, a Apple optou por circuitos e normas proprietárias. A opção da IBM conduziu a que centenas de fabricantes produzissem computadores compatíveis baratos baseados no processador da família x86 da Intel. A influência destes fabricantes no mercado é de tal ordem que quando a IBM tentou inverter a situação com o lançamento da linha PS/2 baseado num barramento proprietário, o MCA - *MicroChannel Architecture*, perdeu completamente o controlo sobre o mercado e sobre a evolução da arquitectura PC por si proposta (nessa altura

um consórcio liderado pelo maior fabricante de *clones* (computadores compatíveis), a Compaq, reagiu com a especificação da evolução do barramento ISA - barramento usado até a altura - para o EISA). Até muito recentemente a Apple não licenciava a terceiros nem a BIOS (*Basic In Out System* - componente fundamental do sistema operativo que permite gerir a comunicação com o sistema), nem o sistema operativo do Macintosh, nem tão pouco fornecia dados sobre as normas de *hardware* deste sistema. Este facto conduziu a uma situação monopolista da Apple, sendo a própria empresa a estabelecer as margens de comercialização. O elevado custo, mantido artificialmente pela Apple, destes equipamentos, conduziu gradualmente a supressão da quota de mercado.

#### **1.4 A Evolução Futura**

Fazer futurologia em informática é extremamente difícil e arriscado. Contudo, dadas as evoluções recentes, é crível que a tendência para a duplicação da performance dos processadores em períodos de 12 a 18 meses se mantenha. A duplicação do número de transístores ocorre actualmente em períodos de dois anos. Estes factos irão certamente contribuir para que a industria informática mantenha a competitividade que a tem caracterizado, dado que os equipamentos tendem a “envelhecer” rapidamente. Isto é, o melhor computador de hoje não passará, daqui a dois anos, de uma simples máquina de jogos sem qualquer utilidade para realizar tarefas produtivas.

## **2. Noções de Informação Binária**

Devido a razões económicas os computadores são construídos com elementos que à semelhança de um interruptor só permitem representar dois estados: ligado ou desligado. Por esta razão diz-se que os computadores digitais (capazes de representar somente um número finito de estados) representam informação binária, que constitui a forma mais elementar de informação. Os possíveis estados traduzem-se por alternativas “sim ou não”, “verdade ou falso”, “ligado ou desligado”. Por questões de simplificação dos fenómenos binários, convencionou-se traduzir os estados binários por “1 ou 0”. O dígito binário, denominado por *bit* (*Binary Information digiT*), só poderá conter um dos seguintes valores: 0 ou 1.

## 2.1 Utilização da Codificação Binária

Como vimos no ponto anterior um dígito binário ou *bit* só permite representar dois estados possíveis. Porém, se agruparmos diversos *bits* já se torna possível representar informação mais complexa. Por exemplo, considere que pretendia representar os estados de um semáforo por codificação binária. Se os estados do semáforo forem vermelho, verde, amarelo e amarelo intermitente poderemos, naturalmente, atribuir um bit para descrever cada um dos estados (vide figura 5 - 1: ligado; 0: desligado), necessitando, nesse caso de 4 *bits*.

Vermelho	Verde	Amarelo	Amarelo Intermitente
0	0	0	1
0	0	1	0
0	1	0	0
1	0	0	0

Figura 5: Representação dos estados de um semáforo.

Como terá reparado, o sistema sugerido não esgota todas as combinações possíveis de *bits* que é possível realizar com 4 dígitos. Este sistema resulta num desperdício de informação. Uma forma mais optimizada de resolver esta tarefa será a da figura 6 . Neste tipo de codificação cada *bit* não representa nenhum estado em concreto. Este só é definido considerando o agrupamento de *bits* como um todo.

Combinação	2º Bit	1º Bit	Estado
1ª Combinação	0	0	Vermelho
2ª Combinação	0	1	Verde
3ª Combinação	1	0	Amarelo
4ª Combinação	1	1	Amarelo Intermitente

Figura 6: Codificação binária eficiente.

Com **2 bits** foi possível representar **4** estados, com **3 bits** é possível representar **8** estados distintos: 000,001,010,011,100,101,110 e 111. Generalizando por recorrência, facilmente se verifica que com ***n bits*** se podem representar  **$2^n$**  estados distintos.

Do exemplo anterior podemos concluir que um grupo de dígitos binários permitem representar qualquer tipo de informação discreta. A lei que define uma determinada correspondência entre um código binário único e cada informação que se pretende representar no computador é chamado de código. A operação pela qual se estabelece a correspondência chama-se de codificação. A operação inversa, isto é, a operação pela qual se obtém uma informação não binária a partir da representação que lhe corresponde um código binário, chama-se descodificação. Assim, pela tabela da figura 6, o código definido para o estado vermelho do semáforo é o 00. À semelhança dos números decimais, o *bit* mais à esquerda de um código binário é chamada o *bit mais significativo* e o *bit* mais à direita o *bit menos significativo*.

A memória de um computador contém exclusivamente 0s e 1s. O seu significado depende do código que lhes é associado em cada momento. Assim, o código binário 01000001 poderá ser interpretado pelo computador como o número inteiro 65 (vide secção sobre codificação de números inteiros) se for usado numa operação de soma, como o carácter “A” se for interpretado como uma entrada do teclado (vide tabela 1), como a cor vermelha de uma figura que se exibe no ecrã se se tratar da representação de um *Bitmap*, ou como a instrução de multiplicação se for interpretado como uma instrução de um programa. Isto é, um determinado código binário só é interpretável se lhe associarmos um determinado contexto.

## 2.2 A Codificação Binária de Números Inteiros

O computador foi inicialmente concebido como uma poderosa máquina de calcular. Qual será então o código usado para representar valores numéricos num computador? Na verdade não existe um único código, mas sim diversos esquemas para realizar essa codificação. Contudo, a codificação mais usada na representação de números inteiros positivos e negativos é a representação em complementos de dois.

Neste código, o bit mais significativo representa o sinal do número. Se este *bit* for 0, o número é positivo, se for 1, o número é negativo. Seja,  $a_{n-1}a_{n-2}a_{n-3}\dots a_2a_1a_0$  ( $a_i \in \{0,1\}, i \in \{0,1,\dots,n-1\}$ ) uma determinada combinação de *bits*. Se o *bit* mais significativo for 0 então o número é positivo e o seu valor é determinado por:

$$\text{Valor} = a_{n-1}2^{n-1} + a_{n-2}2^{n-2} + a_{n-3}2^{n-3} + \dots + a_22^2 + a_12^1 + a_02^0 \quad (1)$$

Código	Sim.	Código	Sim.	Código	Sim.	Código	Sim.
0000000	NUL	0100000	SP	1000000	@	1100000	`
0000001	SOH	0100001	!	1000001	A	1100001	a
0000010	STX	0100010	"	1000010	B	1100010	b
0000011	ETX	0100011	#	1000011	C	1100011	c
0000100	EOT	0100100	\$	1000100	D	1100100	d
0000101	ENQ	0100101	%	1000101	E	1100101	e
0000110	ACK	0100110	&	1000110	F	1100110	f
0000111	BEL	0100111	'	1000111	G	1100111	g
0001000	BS	0101000	(	1001000	H	1101000	h
0001001	HT	0101001	)	1001001	I	1101001	i
0001010	LF	0101010	*	1001010	J	1101010	j
0001011	VT	0101011	+	1001011	K	1101011	k
0001100	FF	0101100	,	1001100	L	1101100	l
0001101	CR	0101101	-	1001101	M	1101101	m
0001110	SO	0101110	.	1001110	N	1101110	n
0001111	SI	0101111	/	1001111	O	1101111	o
0010000	DLE	0110000	0	1010000	P	1110000	p
0010001	DC1	0110001	1	1010001	Q	1110001	q
0010010	DC2	0110010	2	1010010	R	1110010	r
0010011	DC3	0110011	3	1010011	S	1110011	s
0010100	DC4	0110100	4	1010100	T	1110100	t
0010101	NAK	0110101	5	1010101	U	1110101	u
0010110	SYN	0110110	6	1010110	V	1110110	v
0010111	ETB	0110111	7	1010111	W	1110111	w
0011000	CAN	0111000	8	1011000	X	1111000	x
0011001	EM	0111001	9	1011001	Y	1111001	y
0011010	SUB	0111010	:	1011010	Z	1111010	z
0011011	ESC	0111011	;	1011011	[	1111011	{
0011100	FS	0111100	<	1011100	\	1111100	
0011101	GS	0111101	=	1011101	]	1111101	}
0011110	RS	0111110	>	1011110	^	1111110	~
0011111	US	0111111	?	1011111	_	1111111	DEL

Tabela 1: Tabela ASCII.

Por exemplo, considere o código 01101. Repare que o *bit* mais significativo é 0, logo o número é positivo e o seu valor decimal pode ser obtido pela expressão  $1$ , isto é (note que  $n = 5$ ),

$$\text{Valor} = 0 * 2^4 + 1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0$$

ou ainda,

$$\text{Valor} = 0 * 16 + 1 * 8 + 1 * 4 + 0 * 2 + 1 * 1 = 13$$

No caso do número ser negativo, isto é, o *bit* mais significativo ser 1, copia-se o código binário da direita para a esquerda até surgir o primeiro 1 (que também é copiado). Todos os dígitos binário subsequentes são complementados, isto é, se for 0 copia-se um 1 e se for 1 copia-se um 0. O valor decimal correspondente é obtido aplicando a expressão 1 ao código copiado. Considere o código 10010. Usando a regra descrita, obtém-se o novo código 01110, o que aplicando a expressão 1 dá o valor 14. Ou seja, o código original representa o número -14.

Existem ainda códigos especiais para representar números fraccionários. A representação de números nesses códigos chama-se representação em vírgula flutuante.

### 2.3 O Código ASCII

O código ASCII (*American Standart Code for Information Interchange*) é um código que permite representar não apenas grandezas numéricas, mas também caracteres literais e sinais de pontuação. Este código foi desenvolvido para ser usado na transferência de informação entre um computador e os seus dispositivos periféricos (teclado, monitor, impressora, etc.). O código ASCII é formado por grupos de 7 bits (posteriormente foi estendido para 8 bits devido à necessidade de representar caracteres especiais como as vogais acentuadas, o “ç” e outros), permitindo representar os dígitos de 0 a 9, todos os caracteres do alfabeto, sinais de pontuação, sinais de operações aritméticas e de igualdade e uma série de caracteres de controlo, num total de 128 ( $2^7 = 128$ ) símbolos distintos. A tabela 1 ilustra a correspondência entre os diversos símbolos e as respectivas representações binárias.

Pelo facto deste código representar os símbolos habituais de escrita, o código ASCII constitui o código privilegiado de comunicação homem/máquina. Assim, quando introduz a mensagem pelo teclado

Calcula  $A=3*9$

este dispositivo envia ao computador os seguintes códigos ASCII (o símbolo SP “SPACE” representa um espaço):

Caracter	Código
C	1000011
a	1100001
l	1101100
c	1100011
u	1110101
l	1101100
a	1100001
SP	0100000
A	1000001
=	0111101
3	0110011
*	0101010
9	0111001

## 2.4 Noções de Palavra e de Octeto

Um aspecto fundamental que define a tecnologia e as capacidades de um computador é o comprimento, expresso em número de *bits*, da informação binária que o equipamento é capaz de manipular. Por razões tecnológicas verificou-se ser conveniente projectar as unidades que constituem um computador de tal forma que operem sobre grupos de *bits* com o mesmo comprimento. Esse facto conduziu ao conceito de palavra do computador, ou grupo de *bits* suficientes para representar nesse computador uma informação binária. Por exemplo, os *IBM PCs* construídos com processadores até ao *Intel 80386* eram máquinas de 16 *bits*, isto é, só permitiam tratar, em simultâneo, informação codificada até um máximo de 16 *bits*. A partir do *Intel 80386* até ao *Pentium II*, o comprimento de palavra é de 32 *bits*, daí se falar frequentemente em programas de 32 *bits* (programas que aproveitam esta potencialidade).

Além do conceito de palavra, outro conceito que se tornou muito vulgar na representação de informação binária à custa de número fixos de *bits* é o de octeto ou mais vulgarmente conhecido por *byte*. Um *byte* representa qualquer entidade independente representada à custa de 8 *bits*. O *byte* tornou-se ao longo dos tempos a medida padrão para expressar quantidades de *bits*. Os valores mais usados são o *Kbyte* (1024 *bytes*  $\approx$  mil *bytes*), o *Mbyte* (1*Kbyte* \* 1*Kbyte*, ou seja,  $1024 * 1024 = 1048576$  *bytes*  $\approx$  um milhão de *bytes*) - Mega *byte* - e o *Gbyte* (1*Mbyte*\*1*Kbyte*, ou seja,  $1048576 * 1024 = 1073741824$  *bytes*  $\approx$  um bilião de *bytes*) - Giga *byte*.

## 3. Organização de um Computador Digital

Os computadores digitais sequenciais - aqueles que executam uma instrução de cada vez, seguem ainda hoje a arquitectura introduzida pelo matemático John Von Neumann em 1944. As unidades básicas definidas por Von Neumann para o computador digital convencional são as que se apresentam na figura 7. Como pode ser observado, os elementos da arquitectura são: unidades de entrada e saída, memória central e a unidade de controlo (CU - *Control Unit*) e a unidade aritmética e lógica (ALU - *Arithmetic and Logical Unit*) que em conjunto constituem o processador central (CPU - *Central Processing Unit*).

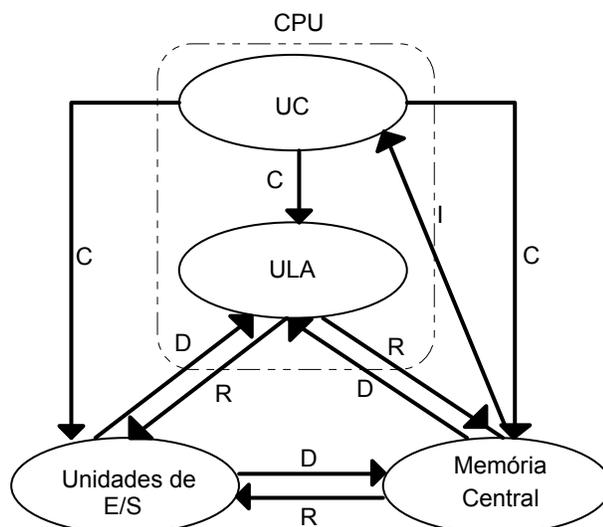


Figura 7: Organização de um computador digital convencional. C - controlo, D - dados, R - resultados, I - instruções.

A memória central constitui o elemento da arquitectura onde são armazenados os **programas** que estipulam ao computador as tarefas elementares a efectuar e a ordem pela qual devem ser realizadas, os **dados** sobre os quais as tarefas elementares serão efectuadas e os **resultados** das operações realizadas sobre os dados.

A **unidade de processamento central** é composta pela **unidade lógica e aritmética**, onde se realizam todas as operações lógicas (por exemplo a comparação entre valores numéricos) e aritméticas (somas, multiplicações, etc.) elementares estipuladas pelos programas, e pela **unidade de controlo**, unidade que controla todo o funcionamento do computador. Esta unidade é responsável pela extracção da memória central das instruções do programa, tendo ainda a seu cargo a respectiva descodificação, com base na qual distribui e gere as tarefas necessárias à sua realização pelas restantes unidades da CPU.

A terceira unidade do modelo de Von Neumann é a unidade de entrada e saída. Esta unidade estabelece a ponte entre o computador e o mundo exterior, permitindo que diversos periféricos (teclado, rato, discos, placas multimédia, ecrã, etc.) sejam ligados e possam comunicar com o sistema central.

Pelo descrito, facilmente se pode estabelecer a analogia entre um computador e os seres vivos. Assim, a unidade de entrada e saída funciona como o sistema sensorial (visão, audição, etc.) e de actuação (mãos, dedos, pés) dado que permite realizar a

troca de informação com o exterior. A memória central assemelha-se à componente do cérebro que nos permite memorizar experiências (programas), que no conjunto condicionam o comportamento. A unidade de processamento central corresponde, por sua vez, à componente do cérebro que nos permite pensar e tomar decisões.

Do modelo de Von Neuman acima apresentado facilmente se distinguem duas categorias de instruções: as instruções de processamento, aquelas que implicam a transferência de dados da memória para a ULA, a realização de operações sobre estes dados e a salvaguarda dos resultados na memória. Outro tipo de instruções são as instruções de entrada e saída que permitem ao sistema receber e enviar dados para o exterior.

### **3.1 A Memória Central**

Como já foi referido o computador armazena na sua memória central os dados a tratar, os resultados do tratamento e os programas que especificam o conjunto de operações a realizar sobre os dados. Embora existam diversas tecnologias de memória, partilhando todas elas os seguintes princípios de funcionamento.

De um modo geral pode-se considerar uma memória como sendo composta por um conjunto de células, permitindo cada uma das quais armazenar uma palavra ou *byte*. A cada célula está associado um número que a identifica unicamente. Esse número é chamado de endereço de memória da célula. Para identificar uma determinada célula, quer em operações de escrita de resultados, quer em operações de leitura de dados, envia-se o seu endereço (geralmente é a CPU que gera o endereço após ter descodificado a instrução que necessita do dado contido na célula), que é temporariamente armazenado no registo (memória bastante rápida que também se encontra no interior das CPUs) de endereços da memória (vide figura 8). O endereço é então utilizado para descodificar de modo a activar a célula pretendida. Consoante o comando seja de leitura ou escrita, a célula visada é preparada para enviar ou receber os dados do barramento de dados (linhas que transportam os dados para as outras unidades do computador). Se o comando for de escrita, os dados que circulam no barramento de dados são inicialmente copiados para o registo de palavra da memória, sendo posteriormente transferidos para a célula visada. Se o comando for de leitura, os dados da célula endereçada são inicialmente transferidos para o registo de palavra e posteriormente para o barramento de dados do computador.

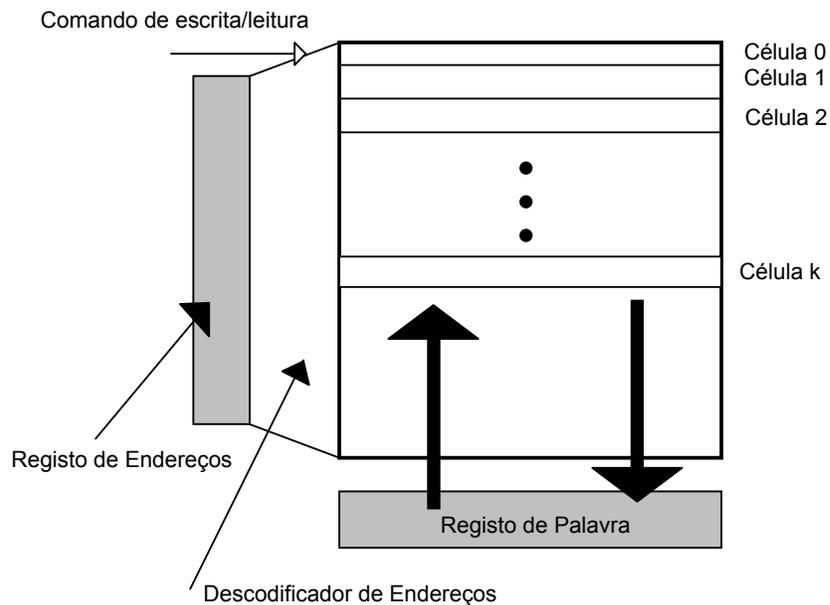


Figura 8: Diagrama simplificado de uma memória.

A existência dos dois registos associados à memória permite desacoplar as velocidades de reacção da memória e do barramento (ou *bus*) de transporte de dados e de endereços.

### 3.1.1 Tipos de Memória

As memórias existentes para computador podem ser classificadas em duas categorias genéricas: as memórias voláteis - aquelas que não mantêm a informação quando são desligadas da corrente - e as não voláteis - aquelas que mantêm a informação mesmo quando são desligadas da corrente.

A memória não volátil existente na memória central do computador é a ROM (*Read Only Memory*). Esta memória, como o nome indica, uma vez escrita - operação realizada na fábrica - nunca mais poderá ser apagada. A sua utilização na construção de computadores serve para guardar o programa básico necessário para que o equipamento, uma vez ligado, carregue o sistema operativo e realize o conjunto de testes necessários à verificação da sua operacionalidade. Outras tecnologias de ROMs são as PROM (*Programmable Read Only Memory*) - idêntica à ROM, embora possa ser programada uma única vez, e a EPROM (*Erasable*

*Programmable Read Only Memory*) - memória com característica de ROM que pode ser programada diversas vezes usando equipamento especial.

As memórias voláteis mais conhecidas são as RAM (*Random Access Memory*), assim denominadas por permitirem acessos aleatórios ao seu conteúdo. De entre estas memórias distinguem-se ainda duas categorias genéricas: as memórias estáticas SRAM (*Static Random Access Memory*) e as memórias dinâmicas DRAM (*Dynamic Random Access Memory*).

A SRAM é tecnologicamente mais evoluída, sendo cada *bit* armazenado com base num circuito composto por 4 ou 6 transístores. Este facto permite que uma vez gravada a informação, a SRAM não perca o seu conteúdo, desde que seja devidamente alimentada electricamente. A DRAM, tecnologicamente menos complexa - cada bit requer somente a existência de 1 transístor - necessita periodicamente de ser refrescada para não perder o seu conteúdo. Isto é, periodicamente a memória terá que ser completamente lida sendo a informação nela reescrita. Se esta operação não se realizar a memória DRAM perde o seu conteúdo em poucos ms. Pelo exposto, facilmente se compreende que as SRAM são memórias bastante mais rápidas que as DRAM. Contudo, dada a complexidade tecnológica do seu fabrico e o número elevado de transístores por *bit*, verifica-se que não é possível produzir memórias SRAM com capacidades de armazenamento tão elevadas como acontece nas DRAM, sendo a diferença de preço entre as duas tecnologias bastante substancial. Assim, é comum encontrar-se a DRAM como a tecnologia empregue na memória central de um computador e a SRAM como a memória destinada a funções muito específicas como na construção de *caches* (vide secção sobre *caches*).

A evolução tecnológica, nomeadamente ao nível da velocidade, das RAM não tem sido tão acentuada com a que se tem verificada com os processadores. Este facto tem conduzido ao surgimento de diversas tecnologias de RAM, que são aplicadas na construção dos computadores. Dadas as diferentes performances (e preços), é necessário que o potencial comprador de um sistema informático tenha a consciência que a RAM que compõe o seu sistema poderá condicionar fortemente o desempenho do mesmo. As principais optimizações ao nível das RAMs têm-se centrado em quatro domínios:

1. Aproveitamento de características eléctricas e adição de lógica adicional para aumentar a velocidade de transferência em leituras de células adjacentes. De

entre as RAMs que aplicam este princípio as mais utilizadas são as EDORAM (*Extended Data Out Random Access Memory*) - equipam geralmente os computadores de topo de gama dada a sua elevada eficiência - e as FPM RAM (*Fast Page Mode Random Access Memory*) - estas memórias tentam prever o próximo acesso iniciando-o antes que este seja realizado.

2. Inclusão de *caches* (vide secção sobre *caches*), construídas com SRAM, internas ao integrado de memória para melhorar o desempenho das transferências. Estas memórias são regra geral designadas de CDRAM (*Cached Dynamic Random Access Memory*) ou EDRAM (*Enhanced Dynamic Random Access Memory*). A sua utilização é pouco frequente dado o seu elevado preço.

3. Extensão do número de portos de acesso simultâneo permitindo servir entidades distintas ao mesmo tempo. Esta tecnologia tem sido particularmente aplicada com sucesso na construção de memórias de vídeo, sendo estas RAMs designadas de VRAM (*Video Random Access Memory*). É também possível encontrar VRAM com *cache* incorporada, designando-se, nesse caso, a memória de CVRAM (*Cached Video Random Access Memory*). Embora a CVRAM seja pouco frequente, é possível encontrar a VRAM nas melhores placas gráficas. A sua utilização é especialmente aconselhada quando se pretende trabalhar em modos de ecrã de elevada resolução e de elevado número de cores. Em modos de resolução menos exigentes a utilização de placas construídas com DRAM normal é suficiente para permitir actualizações de ecrã suficientemente rápidas.

4. Incorporação de capacidades elementares de processamento nos integrados de RAM libertando desse modo a CPU. Esta tecnologia tem sido aplicada no fabrico de integrados de memória vídeo para sistemas de elevado desempenho. Nesta tecnologia de RAMs destacam-se as WRAM (*Window Random Access Memory*), especialmente concebidas para implementarem operações de animação, e a 3D RAM, que foram desenvolvidas pela Matsushita para equiparem placas gráficas especialmente concebidas para animações 3D.

Na tabela 2 encontra-se um guia com algumas das abreviaturas mais usuais das RAM.

Abreviatura	Significado
CDRAM	Cached DRAM
CVRAM	Cached VRAM
DRAM	Dynamic RAM
EDRAM	Enhanced RAM
EDORAM	Extended Data Out RAM
EDOSRAM	Extended Data Out SRAM
EDOVRAM	Extended Data Out VRAM
FPM RAM	Fast Page Mode RAM
FRAM	Ferroelectric RAM
RDRAM	Rambus RAM
SDRAM	Synchronous RAM
3D RAM	Integrado produzido pela Matsushita para visualizações 3D
VRAM	Video RAM
WRAM	Window RAM

Tabela 2: Designações de RAM.

### 3.1.2 As Memórias *Cache*

Uma memória *cache* é simplesmente uma memória geralmente pequena e extremamente rápida (tipicamente SRAM) que se localiza entre um dispositivo lento, como por exemplo a memória central (DRAM), e um dispositivo rápido como o processador. O objectivo da *cache* é acelerar a resposta do dispositivo lento. Para conseguir este objectivo, é mantida na *cache* uma cópia dos dados e fragmentos de programa mais recentemente usados. De facto, verifica-se que a maioria dos programas exibem a característica da localização temporal e espacial. Isto é, a maioria dos programas executam repetidamente o mesmo código numa certa vizinhança e sobre dados contíguos durante um certo período de tempo. Assim, se o fragmento de código em causa e uma certa vizinhança dos dados a tratar forem

mantidos na memória rápida, o sistema responde como se o equipamento mais lento funcionasse à velocidade da memória *cache*.

A filosofia de *cache* pode ser encontrada em diversas componentes de um computador. Por exemplo, a CDRAM e CVRAM usam este princípio para acelerarem o acesso aos seus dados; os processadores INTEL incorporam desde o i486, uma *cache* interna que tem vindo a aumentar de geração em geração. Só para se ter uma ideia da importância desta tecnologia, o processador Pentium Pro da Intel deve, em grande medida, a sua elevada eficiência a uma *cache* de 512 *Kbytes* bastante rápida implementada no próprio integrado.

É frequente encontrarem-se sistemas com diversos níveis de *cache*. O primeiro nível de *cache* é, regra geral, denominado L1 (Level 1), o segundo nível L2 (Level 2), e assim sucessivamente. No Pentium Pro, os 512 *Kbytes* de *cache* são do tipo L2, uma vez que o próprio processador já contém 8 *Kbytes* de *cache* interna. Na tabela 3 são listados alguns dados sobre a performance de um sistema com diversas configurações de *caches*. É de salientar que com somente 16 *Kbytes* de *cache* o sistema sofre um aumento de eficiência de 35%.

Tamanho da <i>Cache</i>	Performance relativamente ao acesso em DRAM
1K	0.91
8K	1.25
16K	1.35
32K	1.39

Tabela 3: Configurações de *caches* e respectivas performances.

### **3.2 O Processador Central: A Unidade de Controlo**

Como já se referiu, a unidade de controlo de um processador desempenha as funções de:

1. extrair uma a uma as instruções do programa a executar;
2. analisar as instruções e

3. gerar os sinais de comando que são enviados às diversas unidades do computador garantindo desta forma que a instrução seja executada.

A primeira função da unidade de controlo é basicamente implementada à custa de dois registos que existem na UC (Unidade de Controlo): o registo de contador de programa (PC - *Program Counter*) ou registo contador de instruções (IC - *Instruction Counter*) que envia para a memória o endereço da célula que contém a próxima instrução a ser executada; o registo de instrução (IR - *Instruction Register*) que recebe da memória cada instrução guardando-a o tempo necessário à sua interpretação. Assim, para executar um determinado programa, é fornecido ao registo contador de programa o endereço da sua primeira instrução. Esse endereço é enviado à memória com a indicação de que se pretende realizar uma leitura. A memória responde, colocando o conteúdo da célula visada no barramento (*bus*) de dados, sendo essa informação captada pelo registo de instrução. Após a análise da instrução, o registo contador de programa é automaticamente incrementado de uma unidade, ficando, deste modo, a apontar para a próxima instrução do programa. Este processo é repetido até ao final do programa ou até que surja uma instrução que quebre esta sequência (instrução de salto).

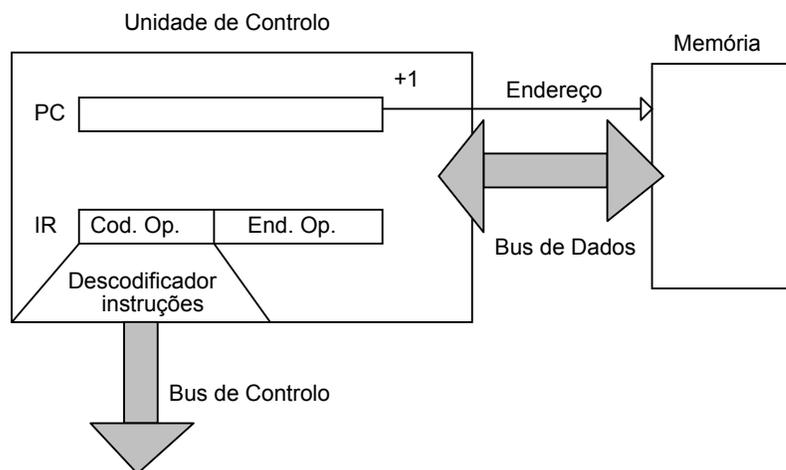


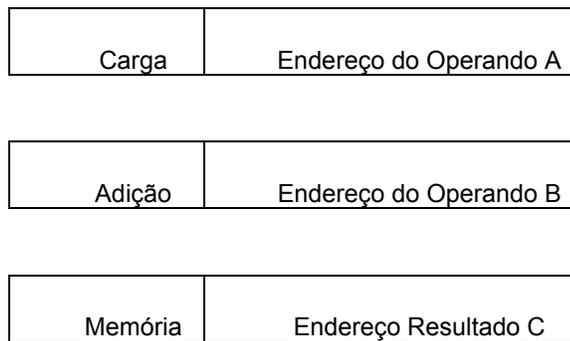
Figura 9: Componentes da unidade de controlo.

O registo de instrução é composto por duas secções (vide figura 9): a primeira onde se armazena o código da instrução (código binário) e a segunda que contém o endereço do operando.

Imagine que pretendia realizar a seguinte operação de soma (A, B e C são variáveis que se encontram armazenadas na memória):

$$C = A + B$$

O programa para executar esta tarefa poderá ser constituído pelas seguintes instruções:



Inicialmente o programa (o conjunto de instruções que o compõem) e os seus dados encontram-se em memória à espera de serem tratados (vide figura 10).

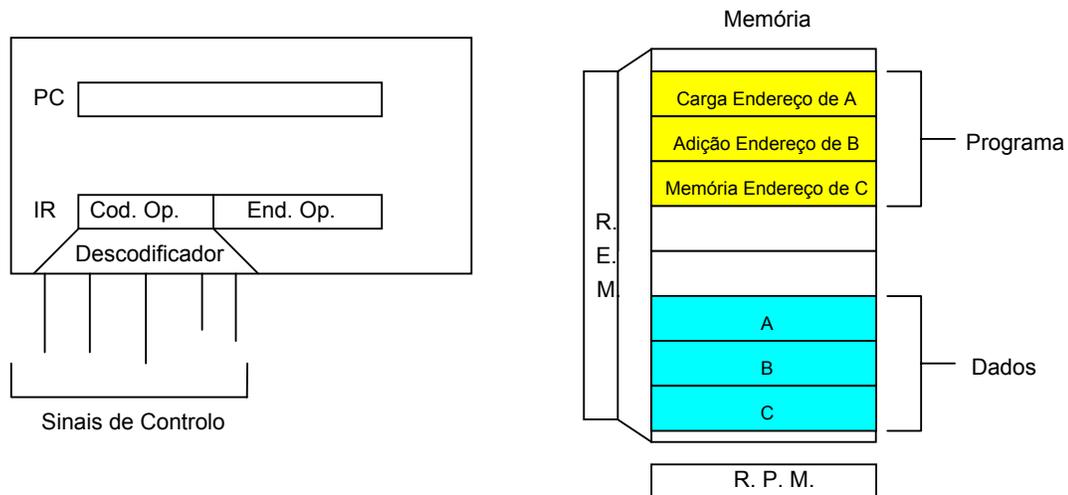


Figura 10: Estado inicial do computador.

Quando o sistema operativo inicia a execução do programa, coloca no registo PC o endereço da primeira instrução, o que por sua vez irá originar a sua leitura para o registo de instrução (IR). Para o efeito, a unidade de controlo comunica o endereço da célula pretendida à memória e activa o comando de leitura. A memória por sua vez reage a estas ordens colocando a instrução no registo de palavra, sendo a sua transferência para o barramento de dados comandada pela UC. Em simultâneo, a

UC comanda o registo IR para que este copie os dados que circulam no barramento de dados. O procedimento atrás descrito encontra-se ilustrado na figura 11.

Esta fase existe para todas as instruções de um computador, visto que para que uma instrução possa ser executada é necessário copiá-la para o registo da unidade de controlo onde esta será analisada.

Uma vez no registo IR, a instrução é descodificada pelo descodificador de instruções, o qual reconhece que se trata de uma instrução para ler uma dada célula de memória e inicia o procedimento para a realização dessa tarefa. O descodificador de instrução primeiro comanda a transferência do endereço contido na instrução para o registo de endereço de memória. Para o efeito, activa os sinais de leitura da memória e coloca o endereço no barramento de endereços. Seguidamente, a memória disponibiliza o dado A e a unidade de controlo activa os sinais necessários para que o dado lido seja guardado no registo acumulador da unidade lógica e aritmética (vide figura 12).

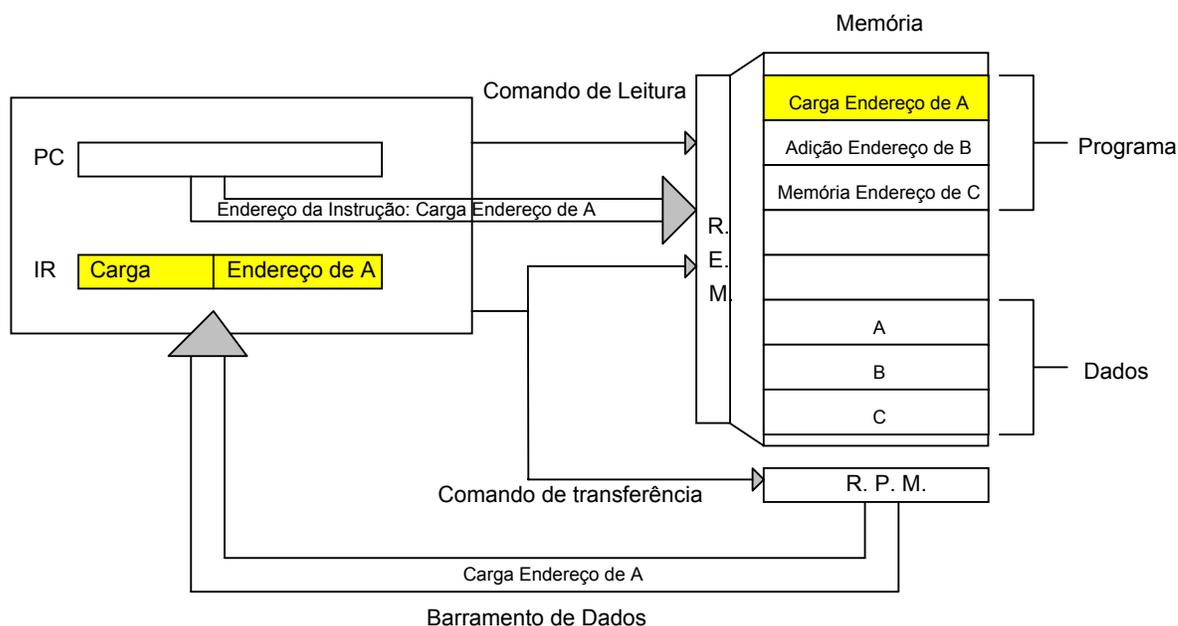


Figura 11: Leitura da primeira instrução.

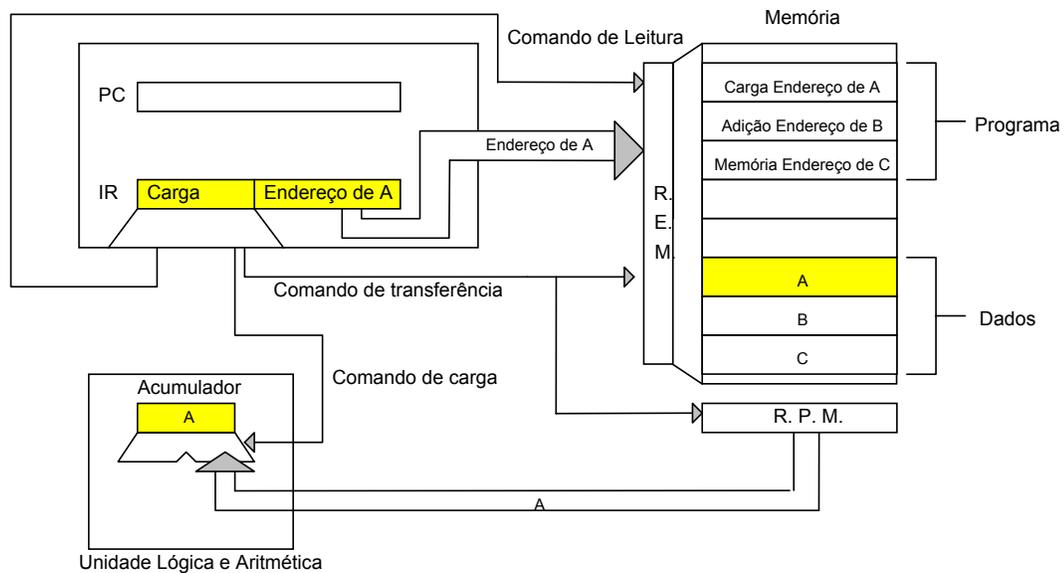


Figura 12: Transferência do primeiro operando A para o acumulador.

Finda a execução da instrução “Carga Endereço A” a unidade de controlo prepara a execução da próxima instrução. Para o efeito o contador do programa é incrementado de uma unidade, passando a conter o endereço da célula de memória com a segunda instrução do programa: “Adição Endereço de B”. Esta fase, tal como a fase de leitura da instrução, existe sempre para todas as instruções do computador e desenrola-se, normalmente, em simultâneo com a primeira fase (busca da instrução) ou segunda fase (descodificação e execução da instrução) da execução de uma instrução.

A execução da segunda instrução do programa inicia-se de forma idêntica à da primeira instrução (vide figura 11). Uma vez no registo de instrução, o descodificador reconhece a segunda instrução do programa como sendo um comando para adicionar o conteúdo do acumulador com o conteúdo da célula, cujo endereço é incluído na instrução. Assim, a UC comanda a transferência do endereço para a memória, juntamente com o comando de leitura, que responde colocando o dado B no barramento de dados. Posteriormente, a unidade de controlo activa as linhas de comando necessárias para que a unidade lógica e aritmética realize a adição do dado que circula no barramento de dados com o valor que se encontra no acumulador. O resultado da operação é guardada novamente no acumulador (vide figura 13).

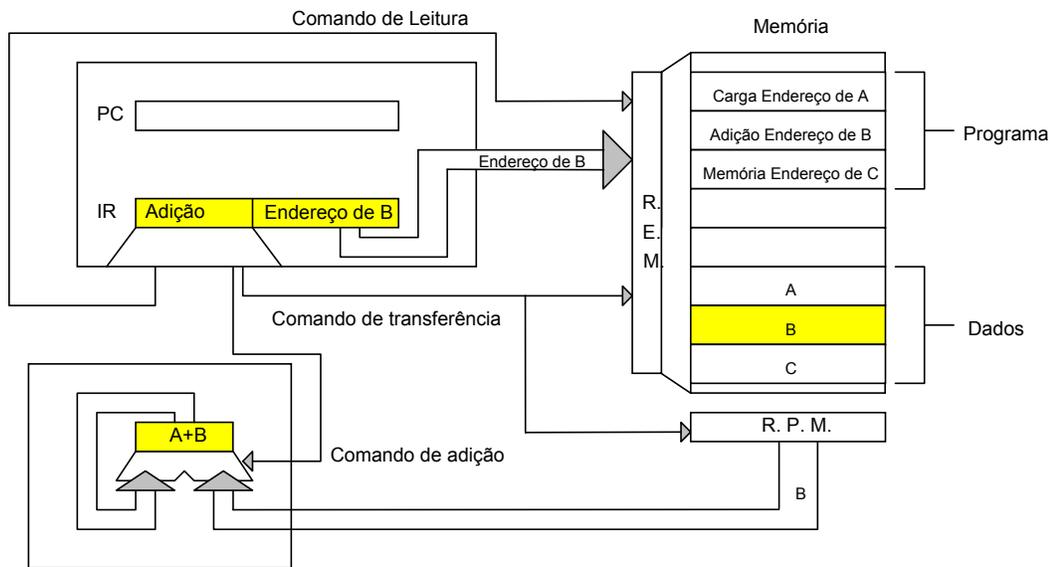


Figura 13: Realização da operação de adição.

Uma vez realizada a operação de adição, a unidade de controlo prepara o registo PC para a próxima instrução, executando o procedimento já descrito para carregar a terceira instrução no registo de instrução. Uma vez no IR, o decodificador identifica o código da instrução como sendo o de uma instrução para transferir o conteúdo do acumulador para a célula cujo endereço consta na instrução. Assim, coloca o endereço no barramento de endereços e comanda a memória para uma operação de escrita. Simultaneamente, comanda a transferência do dado do acumulador para o barramento de dados. Este dado é lido pelo registo de palavra da memória que, usando o endereço entretanto transferido para o registo de endereços da memória, copia o seu valor para a célula pretendida (vide figura 14). No final, a UC prepara novamente o registo PC para a próxima instrução.



de controlo, nomeadamente o seu descodificador, com base numa ROM, chamando-se aos programas aí residentes microcódigo.

Os processadores RISC implementam, regra geral, um conjunto de instruções bastante reduzido (poucas dezenas), o que permite que o descodificador seja implementado em lógica pura (circuitos básicos de lógica booleana). Este facto resulta em processadores bastante mais eficientes e capazes de funcionar a frequências de relógio bastante mais elevadas (geralmente os processadores RISC são aplicados em estações de trabalho).

Actualmente assiste-se a uma fusão de ambas as tecnologias. Por exemplo, o Pentium Pro da Intel é um processador CISC em que cada instrução complexa é internamente traduzida para um conjunto de instruções simplificadas que são executadas por tecnologia RISC. Deste modo, conseguiu-se manter as vantagens da arquitectura CISC, aproveitando, em simultâneo, as vantagens da arquitectura RISC.

### 3.2.1 A Evolução dos processadores do IBM PC

O primeiro PC da IBM foi construído com base no processador 8088 da Intel (um processador externamente de 8 bits), tendo rapidamente evoluído para o processador 8086 de 16 bits da mesma empresa (XT da IBM). Este processador só permitia, contudo, endereçar até 1Mbyte de memória, o que, devido a razões de compatibilidade, veio a constituir o maior *handicap* do MS-DOS. De facto, o DOS (*Disk Operating System*), o primeiro sistema operativo da era PC, nunca foi capaz de lidar com espaços de memória superiores a 1Mbyte.

Com a introdução do IBM AT surgiram os processadores 80286 da Intel. Estes processadores já permitiam o funcionamento em modo protegido, um modo em que é possível aceder a mais do que 1Mbyte de memória. Foi ainda integrada neste processador a possibilidade de gerir memória virtual - neste modo o disco passa a funcionar como uma extensão da memória central do computador, ou por outras palavras, a memória central passa a ser o disco, sendo a RAM uma espécie de *cache* dessa memória. Teoricamente, esta técnica permite que o computador disponha de uma quantidade de memória praticamente ilimitada.

A primeira grande evolução da arquitectura 80x86, também muitas vezes designada por arquitectura x86, sucedeu com a introdução do Intel 80386. Este foi o primeiro processador de 32 bits a ser integrado no PC. Contudo, dada a baixa aceitação do

sistema de 32 *bits* - só muito recentemente (desde a introdução do Windows 95) é que esta propriedade tem sido devidamente aproveitada - a Intel desenvolveu a partir do 80386 um processador cujo barramento externo se mantinha nos 16 bits, o 80386 SX.

A próxima evolução dos processadores Intel foi o 80486, posteriormente renomeado para i486. Este processador, poucas melhorias introduziu na arquitectura 80386. Basicamente, um i486 é um 80386 com uma *cache* (4Kbytes para instruções e 4Kbytes para dados) L1 integrada no processador, bem como dispondo no mesmo integrado do chamado coprocessador, ou processador de vírgula flutuante (FPU - *Floating Point Unit*) - unidade otimizada para o cálculo com números reais (números fraccionários). Do i486 a Intel lançou uma versão (muitos especularam de que se tratava de processadores i486 normais com defeitos de fabrico) sem coprocessador, o i486 SX.

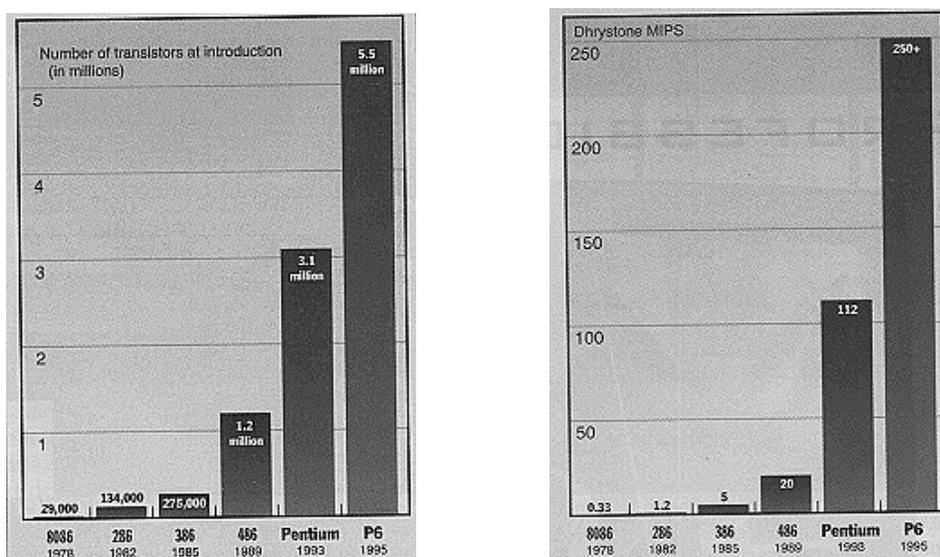


Figura 15: A evolução dos processadores. Esquerda - evolução do número de transístores; direita - evolução do desempenho em número de MIPS (*Mega Instruction Per Second*).

Com a introdução do Pentium (nome de código P54C), a quinta geração de processadores da arquitectura x86, a designação 80x86 foi abandonada, o que se deveu ao facto de não ser possível patentear o nome de um processador constituído unicamente por um número. Até à introdução do Pentium, a escolha de um processador era relativamente trivial. O sistema era tanto mais rápido quanto o fosse o seu processador. Após a introdução do Pentium, os concorrentes da Intel (a AMD,

a NexGen - adquirida em 1995 pela AMD, e a Cyrix), deixaram de se limitar a copiar a tecnologia Intel, passando a introduzir alterações significativas nos processadores que produzem.

Tem sido interessante a guerra dos microprocessadores.

A tecnologia MMX foi desenvolvida pela Intel e é formada por 57 novas instruções especialmente projectadas para a manipulação de imagem e som. Deste modo a Intel integrou nos seus processadores capacidades de processamento até aí só possíveis pela utilização de equipamento dedicado baseado em processadores DSPs (*Digital Signal Processing* - processadores especialmente concebidos para manipularem sinais digitais como a voz, a imagem, etc.). O Pentium MMX (nome de código P55C) é basicamente um Pentium P54C ao qual foram acrescentadas as potencialidades MMX. Dadas algumas ligeiras alterações introduzidas ao nível da arquitectura (duplicação das *caches* L1 e melhorias no *pipeline*) este processador estima-se em cerca de 15% mais rápido que o seu equivalente P54C em tarefas não MMX.

O Pentium Pro, nome de código P6, surgiu no final de 1995 e foi o primeiro processador de sexta geração da Intel. Este processador introduziu a utilização de tecnologia RISC nos processadores Intel, e dispunha de uma *cache* L2 de 512Kbytes integrada no mesmo integrado do processador (o que contribui significativamente para o seu preço elevado). Este facto permitiu à *cache* funcionar à mesma velocidade do processador. As alterações arquitectónicas permitiram ao Pentium Pro funcionar cerca de 33% mais rápido que um Pentium P54C equivalente. O Pentium Pro sofreu, contudo, de um grande *handicap*: a execução de código de 16 *bits* não era eficiente, ficando muito aquém do Pentium P55C. Este facto condicionou a utilização deste processador em computadores Windows 95.

Em 1997 a Intel lançou o Pentium II, nome de código Klamath. O Pentium II derivou directamente do Pentium Pro, sendo, portanto, um processador com tecnologia da sexta geração. Em relação ao seu antecessor, o Pentium II não dispunha da *cache* L2 interna (o que contribui para o seu preço relativamente baixo), funcionando esta fora do integrado do processador a metade da frequência deste. Em contrapartida, a *cache* L1 foi duplicada, dispondo o Pentium II de 16Kbytes de *cache* para os programas e de mais 16 Kbytes de *cache* para os dados. Esta duplicação atenuou significativamente a diminuição da velocidade de troca de dados com a *cache* L2. Outra melhoria introduzida no Pentium II foi a correcção dos problemas na execução

de código 16 *bits* que o Pentium Pro apresentava. Este processador incluiu, ainda, o suporte MMX. A grande desvantagem do Pentium II residiu na sua construção. O processador era embalado num integrado SEC (*Single Edge Contact*) não sendo compatível com o *pin-out* (distribuição de pinos para a ligação à placa mãe) dos integrados precedentes. Assim, não era possível realizar um *upgrade* do sistema pela simples troca do processador. Em vez disso era necessário trocar a *motherboard* (placa mãe - placa onde residem a grande maioria dos integrados do computador).

O Intel P7/Merced foi o processador da sétima geração da Intel. Este processador foi uma das maiores evoluções da arquitectura x86 desde a introdução do 80386, já que estendeu as instruções para uma arquitectura de 64 *bits*.

A Cyrix desenvolveu o 6x86. Este foi um processador ao nível do P54C da Intel, exibindo, contudo, melhores prestações que este, o que se deveu fundamentalmente ao seu superior barramento de I/O (*In/Out* - entrada / saída).

Em Junho de 2000 foi ainda lançado pela VIA Technologies o Cyrix III (nome de código Samuel). Este processador apresentou-se mais pequeno e com menores consumos quando comparado com os processadores similares da AMD e da Intel.

A AMD entrou na corrida dos processadores com o K5. Originalmente, projectado para ser equivalente ao Pentium Pro, este processador apresentou desempenhos comparáveis ao P54C da Intel. Posteriormente revisto, o K5 sofreu algumas alterações ao nível da implementação interna de algumas instruções, o que lhe permitiu obter desempenhos mais elevados (cerca 33% a mais) que o equivalente P54C.

Posteriormente a AMD lançou o K6. Originalmente desenvolvido pela NexGen, empresa adquirida em finais de 1995 pela AMD, o K6 foi um processador da sexta geração que rivalizou directamente com o Pentium II. Embora menos eficiente que o Pentium II, o K6 apresentou desempenhos superiores ao Pentium Pro e dispunha igualmente de suporte MMX.

Para combater o sucesso do K6 a Intel lançou o Pentium III. No entanto, mais tarde, foi lançado pela AMD o Atlon, que apresentou ainda melhores valores de performance.

Depois de uma interessante e disputada corrida por maiores velocidades, onde a AMD foi a primeira a ultrapassar o 1GHz, a Intel desenvolveu o Pentium IV com velocidades superiores a 2 GHz.

No presente, a Intel continua a fornecer as linhas Celeron e Pentium III, para além do Pentium IV.

A AMD tem vindo a desenvolver o Athlon. No entanto, esta companhia introduziu ainda a nova linha Duron para competir directamente com os Celeron. Com o aparecimento do Duron 1200, este processador vai apresentando funcionalidades que se aproximam do Pentium IV.

#### **4. Hardware e Software**

Um computador é composto por duas componentes fundamentais, as unidades físicas e os programas sem os quais um computador não passaria de um conjunto de integrados. À primeira componente é usual chamar-se *Hardware* e abrange tudo o que é material num computador, desde a caixa do computador, passando pelos seus periféricos (impressoras, modems, ecrãs, teclados, ratos, etc.) até aos circuitos com que é construído. A segunda componente designa-se por *Software* e descreve todo o domínio da programação.

Os domínios do *Hardware* são:

- Tecnológico: o domínio tecnológico do *Hardware* diz respeito à tecnologia da constituição dos componentes, em particular dos componentes electrónicos (transístores, resistências, etc.) que constituem o computador.
- Lógico: o domínio lógico do *Hardware* especifica metodologias e técnicas da interligação dos componentes em unidades funcionais (módulos lógicos com base nos quais os computadores são construídos).
- Arquitectónico: o domínio arquitectónico do *Hardware* especifica o modo como as unidades lógicas são interligadas.

O domínio do *Software* compreende todos os aspectos relacionados com a elaboração e utilização de programas. Como já foi amplamente discutido, o *Hardware* de um computador só entende sequências de 0s e 1s. Assim, é necessário que um programa seja constituído por um conjunto de instruções codificadas de forma conveniente em sequências de dígitos binários. A esta

linguagem chama-se linguagem máquina, a única que um computador entende. Pelo exposto, facilmente se entende que uma linguagem máquina reflecte obrigatoriamente a estrutura interna do computador, razão pela qual um programa escrito para PC não é entendido por um Macintosh da Apple.

Embora os conjuntos de 0s e 1s sejam de fácil entendimento para um computador, o mesmo não acontece para o ser humano, a entidade que tem de escrever os diversos programas. Por essa razão, os fabricantes de processador concebem linguagens baseadas em notações mnemónicas, de melhor assimilação. Estas mnemónicas correspondem directamente aos conjuntos de sequências binárias que constituem a linguagem máquina de um determinado processador. A esta linguagem chama-se *assembly language*. A tradução do conjunto de mnemónicas para as sequências de binárias é realizada por um programa denominado o *assembler*.

Quer a linguagem máquina, quer a linguagem *assembly* não constituem linguagens portáteis, isto é, um programa escrito numa determinada linguagem máquina ou *assembly* não corre numa máquina cujo processador não seja o que entende essa linguagem. Atendendo a que os programas actuais, são compostos por centenas de milhares ou mesmo vários milhões de instruções e dada a complexidade que caracteriza esses programas, não seria economicamente possível escreve-los em linguagem máquina ou mesmo *assembly*. A maioria dos programas actuais são escritos em linguagem de alto nível (C, C++, Java, Pascal, Cobol, Fortran, Basic, etc.), linguagem que permite facilmente exprimir procedimentos a seguir na resolução de problemas usando frases simples da linguagem corrente e expressões matemáticas e lógica convencionais. O programa que traduz uma determinada linguagem de alto nível para o conjunto de mnemónicas, que constituem a linguagem *assembly* de um determinado processador, chama-se compilador. Assim, uma vez escrito um programa em linguagem de alto nível, basta empregar um compilador e um *assembler* adequado para o fazer executar em qualquer processador (pelo menos em teoria).

Como já foi referido, os programas actuais são, regra geral, bastante complexos, verificando-se que são escritos por equipas de programadores muitas vezes distantes entre si (muitas vezes as equipas encontram-se a trabalhar em continentes diferentes). Assim, cada equipa escreve os seus módulos do programa, que no final têm que ser ligados para constituírem o produto final. Essa tarefa é realizada pelo *Linker*.

Os diversos níveis da programação encontram-se ilustrados na figura 16.

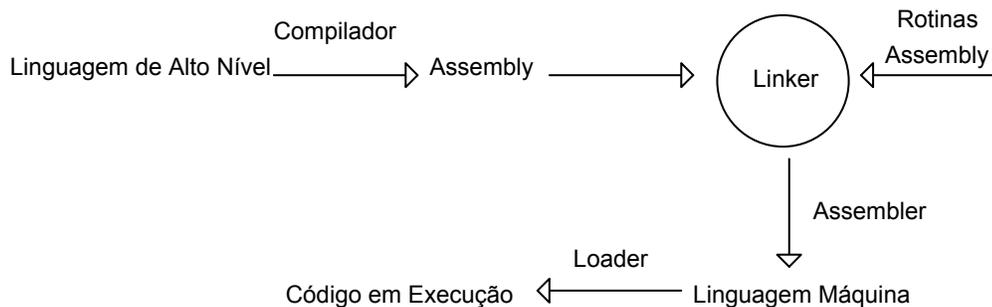


Figura 16: Níveis de programação.

Existem linguagens de alto nível que não são compiladas. Nessas linguagens (chamadas linguagens interpretadas) os compiladores são substituídos por um programa chamado intérprete. Um intérprete é um programa que executa, uma a uma, as instruções de um programa. Isto é, quando se manda executar um programa interpretado, o intérprete busca a primeira instrução do programa, realiza a sua tradução para linguagem máquina e executa-a. Se não tiver ocorrido erro, realiza o mesmo procedimento com a segunda instrução do programa, e assim sucessivamente até executar a última instrução de alto nível do código fonte (vide figura 17). Estas linguagens ainda são bastante vulgares em ambientes de desenvolvimento interativo, como acontece em muitas linguagens de simulação (ex: MatLab - programa de simulação matemática).

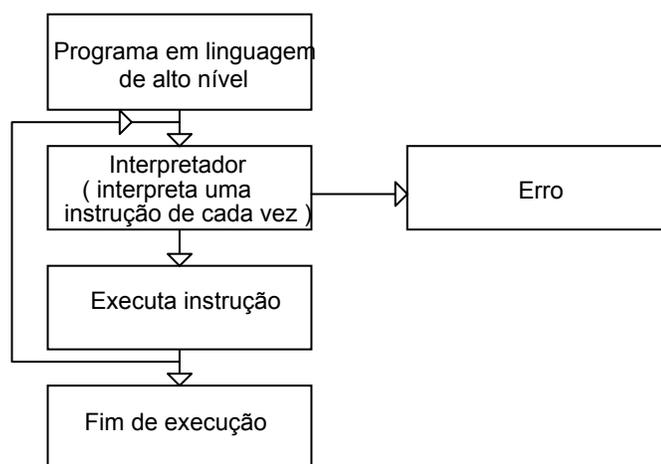


Figura 17: Linguagem interpretada.

Independentemente da linguagem com que forma escritos, é usual distinguir-se entre três categorias de *Software*:

- Software de Sistema: destina-se à gestão do sistema informático. Encontra-se geralmente integrado num todo que constitui o chamado sistema operativo.
- Software de Suporte: destina-se à escrita e aperfeiçoamento dos programas de aplicação (Compiladores, *assemblers*, *linkers*, interpretadores, etc.).
- Software de Aplicação: programas para fins específicos (Processadores de texto, Folhas de Cálculo, etc.).

## 5. Bibliografia

- “Pentium Parte II”, PC Magazine, Julho, 1997.
- A. Dias de Figueiredo, “Apontamentos de Tecnologia da Informática”, DEE, FCTUC.
- A. Van Horn, “Pentium II: King of the x86 Hill”, Byte, June, 1997.
- Gordon B. Davis, “Introduction To Computers”, McGraw-Hill, 1977.
- Nick Tredennick, “Microprocessor-based Computers”, IEEE ?, October, 1996.
- Peter Wayner, “Fast, Smart RAM”, Byte, June, 1995.
- Thomas C. Bartee, “Computer Architecture And Logic Design”, McGraw-Hill, 1991.
- Tom R. Halfhill, “The x86 Gets Faster with Age”, Byte, November, 1996.
- Tom R. Halfhill, “x86 Enters the Multimedia Era”, Byte, July, 1996.
- Tom R. Halfhill, “Intel’s P6”, Byte, Abril, 1995.

# CAPÍTULO II

## Introdução ao Ambiente Windows

*Neste capítulo será proporcionado ao leitor um primeiro contacto com o sistema operativo Windows 2000, sendo introduzidos todos os conceitos básicos necessários à sua manipulação.*

---

### 1. Introdução

Um computador é basicamente uma máquina combinacional complexa que permite, de uma forma rápida, executar instruções (conjunto de entradas) que, uma vez decodificadas (combinadas), conduzem à realização das tarefas pretendidas (saídas). Dada a complexidade destes circuitos, é necessária a existência de um programa que os inicialize e permita realizar determinadas operações básicas, como por exemplo, gestão de periféricos (discos, teclado, rato, ecrã, etc.), por forma a garantir a funcionalidade mínima do equipamento. O programa que assegura estas tarefas chama-se **SISTEMA OPERATIVO**. Os restantes programas (processadores de texto, compiladores, folhas de cálculo, etc.) recorrem, geralmente, aos serviços do Sistema Operativo para realizar as suas tarefas básicas. Por exemplo, quando um programa de processamento de texto (ex: *WORD*) necessita de ler entradas do teclado, informa o sistema operativo o qual concretiza esta tarefa e lhe devolve os dados pretendidos. Por analogia podemos pensar no Sistema Operativo como sendo o conjunto de funcionários administrativos de uma instituição (por exemplo o departamento de engenharia informática). Sem estes, todo o sistema entraria em colapso, uma vez que asseguram serviços básicos de gestão, fundamentais à sua operacionalidade. Por exemplo, um professor recorre aos diversos oficiais de secretaria para gerir o expediente, arquivos e diversos *dossiers*.

Não é que o professor não seja capaz de exercer estas tarefas, simplesmente não está vocacionado para o fazer, nem tão pouco dispõe do tempo necessário. Por outro lado, a existência de funcionários especializados em determinadas tarefas, permite que um conjunto de professores partilhem os seus serviços para o mesmo fim, o que naturalmente eleva a eficiência do sistema. Em suma, cabe ao Sistema Operativo assegurar todas as tarefas de gestão (gestão de periféricos, iniciar e finalizar programas, gestão de memória, etc.) que são necessárias no computador.

O **Sistema Operativo** que se irá utilizar neste curso será o **WINDOWS 2000** da **MicroSoft**. O Windows 2000, à semelhança dos seus antecessores (Windows NT, Windows98, Windows 3.11, Windows 3.1, Windows 3.0, etc.) deve o seu nome ao facto de todas as aplicações serem executadas dentro de janelas (*windows*).

A grande vantagem deste S. O. (Sistema Operativo) em relação a outros programas do género (*DOS, UNIX, etc.*) reside no facto deste, através de uma interface gráfica, tornar a utilização de um PC (*Personal Computer*) bastante fácil e intuitiva.

## 2. Início da Sessão de Trabalho

Para iniciar uma sessão Windows, basta ligar o computador. Após alguns testes ao equipamento (memória, discos, teclado, placa gráfica, etc.) surge-lhe no ecrã o pedido para introduzir o seu *login* e a sua *password*. Este sistema de acesso, para além de permitir o suporte à informação privada, possibilita que o utilizador encontre a sua área de trabalho tal como a configurou sempre que inicia a sessão Windows.

Existem dois tipos de utilizadores no Windows 2000: o administrador e o utilizador comum. Ao contrário deste último, o administrador tem acesso a todos os recursos do sistema, nomeadamente pode adicionar novos utilizadores ou remover utilizadores já existentes.

Ao fim de alguns segundos aparece-lhe no monitor o conteúdo do **DESKTOP** (área de trabalho) que será bastante semelhante ao que se representa na Figura 2.

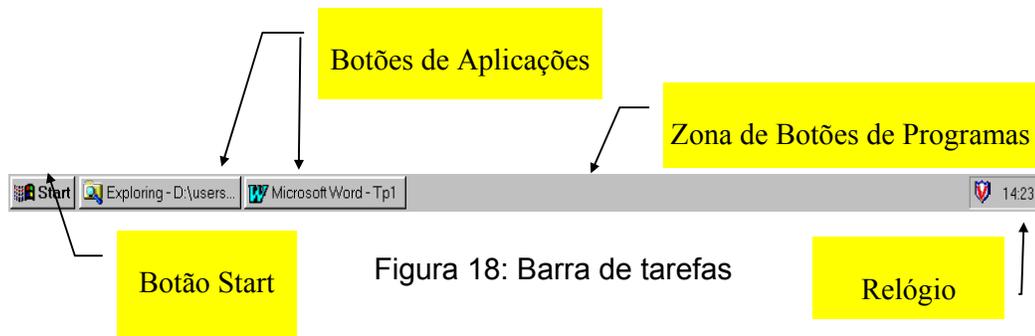
### 2.1 A Área de Trabalho

Ao ecrã da Figura 2 chama-se *DeskTop*, uma vez que, à semelhança do tampo de uma secretária, contém todos os recursos (objectos) disponíveis ao nosso trabalho.

Deste modo o *DeskTop* vai conter os nossos programas, impressoras, cesto de lixo, documentos, etc..

São basicamente três os elementos que compõem o *DeskTop*:

- ① **Barra de tarefas** (vide figura 1): esta barra contém três elementos: o botão start  que, como se verá mais à frente, dá acesso a todos os recursos do sistema (por analogia, o botão *Start* funciona como uma gaveta da secretária onde guardamos todos os objectos de que dispomos para o trabalho); a zona dos botões dos programas (tarefas) que se encontram a correr (esta zona estará obviamente vazia no caso de nenhum programa estar a correr); e o relógio .



- ② **Zona de Recurso:** esta região contém os diversos recursos que, regra geral, são necessários a uma sessão de trabalho. Estes recursos encontram-se representados sob a forma de ícones com títulos   , cujos desenhos são, em princípio, elucidativos do seu conteúdo. Estes recursos podem ser:

- Aplicações:  Cardfile  Microsoft  Notepad
- Recursos Físicos:  Shortcut to Volume 1 (C:)  Shortcut to HP DeskJet 660C
- Documentos:  Shortcut to Tpt1
- Pastas com Documentos e/ou Aplicações:  Eppa

- ③ **O fundo:** esta zona corresponde ao tampo da secretária, onde serão colocados todos os objectos necessários ao nosso trabalho, com a vantagem de, como veremos no final deste capítulo, poder ser personalizado.

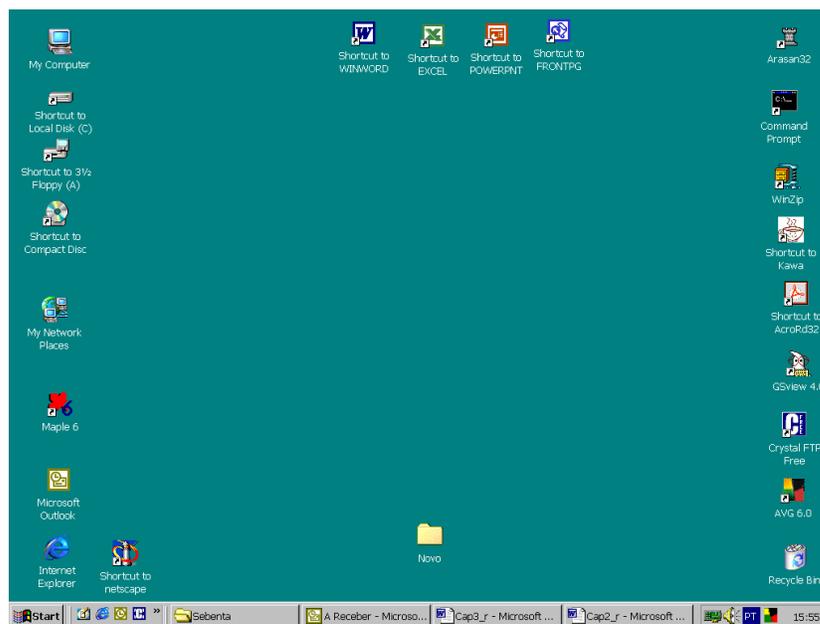


Figura 19: *DeskTop* do Windows 2000

### 3. Execução de Programas

Qualquer Sistema Operativo seria desprovido de interesse se não permitisse correr programas. Iremos então estudar as principais formas de executar aplicações em Windows 2000. Como foi referido na secção anterior, o Windows 2000 permite guardar objectos em dois locais distintos: no botão *Start* e nos ícones sobre o fundo

do *DeskTop* (o mesmo objecto pode ser guardado em ambos os locais em simultâneo).

Para executar qualquer programa a partir de um ícone presente no *DeskTop*, basta colocar o apontador  sobre esse ícone e clicar duas vezes seguidas, com toques rápidos, no botão do lado esquerdo do rato . Em qualquer *DeskTop* Windows existe o ícone relativo ao “Caixote de Lixo” (*Recycle Bin*)  .

Vamos então executar o programa de gestão dos “lixos” produzidos. Coloque o apontador sobre o ícone do *Recycle Bin*. Clique duas vezes seguidas no botão do lado esquerdo do rato. Se tudo tiver corrido bem, surge-lhe junto ao botão uma janela onde o programa pretendido está a ser executado (vide Figura 20).

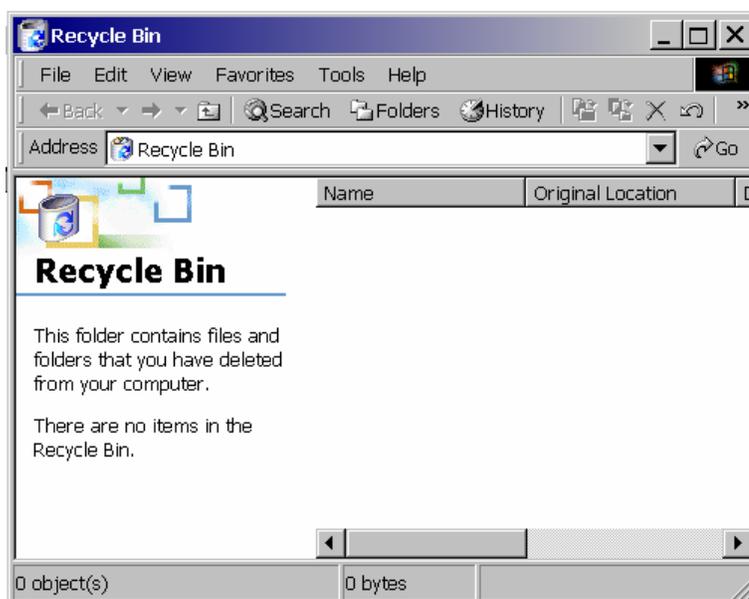


Figura 20 : Janela do *Recycle Bin*

Em alternativa, as aplicações podem ser executadas a partir do botão *Start*  da barra de tarefas. Coloque o apontador do rato sobre o referido botão  e prima o botão do lado esquerdo do rato. Se a operação tiver sido realizada correctamente, surge-lhe no ecrã uma nova janela com as seguintes indicações:

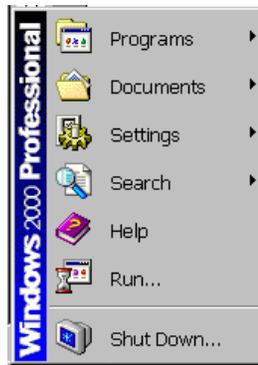


Figura 21: O MENU *START*

À medida que desloca o apontador sobre os seus itens, estes são seleccionados ficando com um fundo diferente do dos seus vizinhos. Para activar qualquer item basta que, uma vez seleccionado, pressione uma vez o botão esquerdo do seu rato. Realize esta operação sobre a opção *Programs*.



Figura 22: O Sub-Menu *Programs*

Repare que é aberta uma nova janela (janela de sub-menu). Alguns dos seus elementos contêm do lado direito uma pequena seta . Esta seta indica que o elemento corresponde a uma nova pasta (sub-menu) de itens. Estes itens também podem facilmente ser identificados pelo ícone . Cada pasta contém por sua vez novos objectos (que também podem ser pastas). Assim, todos os itens da opção  *Programs* da janela correspondem, ou a pastas ou a aplicações. Para abrir uma nova pasta ou executar um programa contido na pasta corrente, basta seleccionar, com o apontador, o item correspondente e premir o botão do lado esquerdo do rato

Mantendo a janela do Recycle Bin aberta, chame o Paint (Programs/Accessories/Paint). Sobre o seu *DeskTop* encontram-se agora abertas duas janelas. Isto significa que o computador está a correr em simultâneo duas aplicações distintas. Coloca-se então a questão de saber como se comuta entre as diversas aplicações em execução no Windows. Embora existam diversas formas de o conseguir, iremos estudar somente as duas mais significativas.

#### 4. Comutação entre Programas

Em Windows 2000, a aplicação “activa”, é aquela onde o utilizador pode introduzir dados, isto é, a janela que não contém nenhuma outra sobreposta a ela. No caso do exemplo da figura 6a) a aplicação activa é o Paint. Ora, uma forma muito simples de activar uma janela consiste em, uma vez deslocado o apontador sobre uma porção visível da sua área, premir o botão do lado esquerdo do rato . Vamos então activar a janela Recycle Bin. Arraste o apontador até uma secção visível da sua área e prima o botão esquerdo do rato. Repare que a janela do Recycle Bin se sobrepõe à da aplicação Paint.

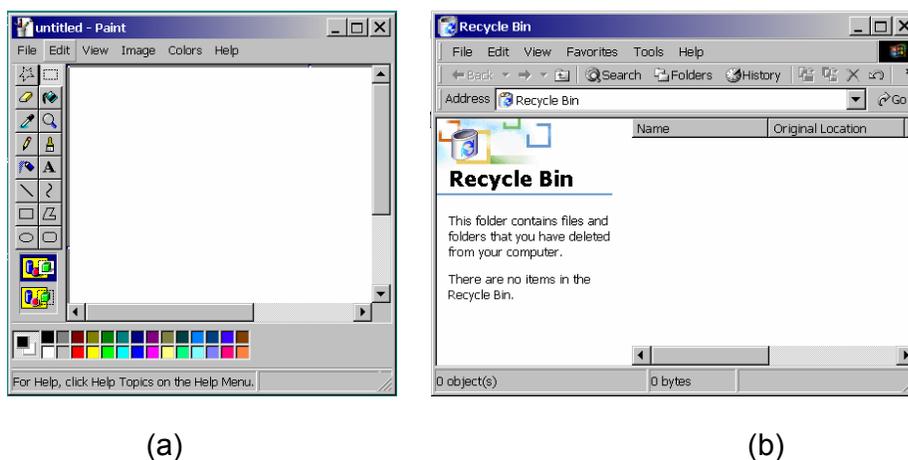


Figura 23: Exemplos de janelas activas

Vai agora repetir o exercício activando novamente a janela do Recycle Bin. Ora, para isso...☹. Neste momento já se terá certamente apercebido que esta janela foi completamente escondida pela do Recycle Bin (vide figura 6b). Esta é uma de muitas situações em que o processo anterior não pode ser aplicado. E agora, como

se poderá resolver esta situação? Felizmente o Windows dispõe de um mecanismo alternativo de comutação.

Já se terá certamente apercebido que, quando executou os programas Recycle Bin e Paint, surgiram dois botões  Recycle Bin  untitled - Paint na barra de tarefas. Estes botões têm diversas finalidades, uma das quais consiste na comutação de aplicações. Repare que o botão correspondente à aplicação Paint dá a sensação de estar premido. A ideia desta barra assemelha-se à de uma aparelhagem de som: para seleccionarmos o gira-discos como meio de reprodução, temos que premir o botão apropriado. Assim, para comutar entre aplicações basta premir o botão correspondente à aplicação pretendida. Comute, então, para a aplicação Recycle Bin.

## 5. Utilização do Rato em Windows

Como já se terá apercebido, um conceito intimamente ligado ao Windows é o do RATO. Embora seja possível trabalhar sem este periférico auxiliar, é no seu uso que reside grande parte da facilidade e intuição subjacente a este Sistema Operativo. Em Windows, somente os botões do lado esquerdo e direito têm funções específicas e bem definidas (nos ratos com mais de dois botões, os restantes têm funções que dependem da aplicação em causa).

O botão do lado direito dá, regra geral, acesso aos MENUs auxiliares. Este assunto será debatido mais à frente neste texto.

São fundamentalmente três as operações que são possíveis executar com o botão do lado esquerdo:

- ① **Premir uma vez o botão (clique simples).** Regra geral, esta operação resulta na selecção do objecto que se encontra sob o apontador. Coloque o apontador sobre um dos ícones presentes no *DeskTop* (por exemplo:  My Computer ) e prima o botão esquerdo do rato somente uma vez. Repare que o ícone foi seleccionado (  My Computer ).

- ② **Premir uma vez, mantendo o botão em baixo durante o arrastamento do apontador.** Esta operação resulta, regra geral, na selecção múltipla de todos os objectos compreendidos entre o ponto inicial (ponto onde o botão foi inicialmente premido) e o final (ponto onde se eliminou a pressão sobre o botão) do arrastamento, podendo ainda ser utilizado para arrastar objectos quando estes o permitem. Prima duas vezes seguidas as teclas ALT F4 (o seu significado será explicado mais à frente neste texto). Repare que as duas aplicações que estavam activas foram terminadas (desapareceram as janelas e os botões de aplicação na barra de tarefas). Seleccione todos os ícones do seu *DeskTop*. Agora mova o apontador para fora da região seleccionada e prima uma vez o botão esquerdo do seu rato. O que aconteceu? Sabe explicar?
- ③ **Premir duas vezes consecutivas o botão (duplo clique).** Esta opção permite quase sempre executar operações sobre o objecto sob o apontador. Por exemplo, se o objecto for o ícone de uma aplicação, esta será activada.

## 6. As Janelas

Como já terá notado, um elemento fundamental no *Windows*, como aliás o próprio nome indica, são as JANELAS. Embora existam vários tipos de janelas, iremos estudar as de aplicação e as dos diálogos.

Quando se executa uma aplicação, em princípio, esta corre numa janela que lhe pertence exclusivamente. Como já vimos, para poder interactivar com uma aplicação, é necessário que a sua janela se encontre activa no *DeskTop*. Embora janelas diferentes exibam estruturas organizativas distintas, existe um conjunto de elementos que usualmente são comuns a todos os tipos.

Uma janela típica de uma aplicação apresenta, regra geral, os seguintes elementos (vide figura 7):

- Barra de Título
- MENU
- Barra de Ferramentas
- *Scrollers*
- Área de Trabalho

- Botão de Sistema
- Botões de Controlo (redimensionamento e encerramento da aplicação)

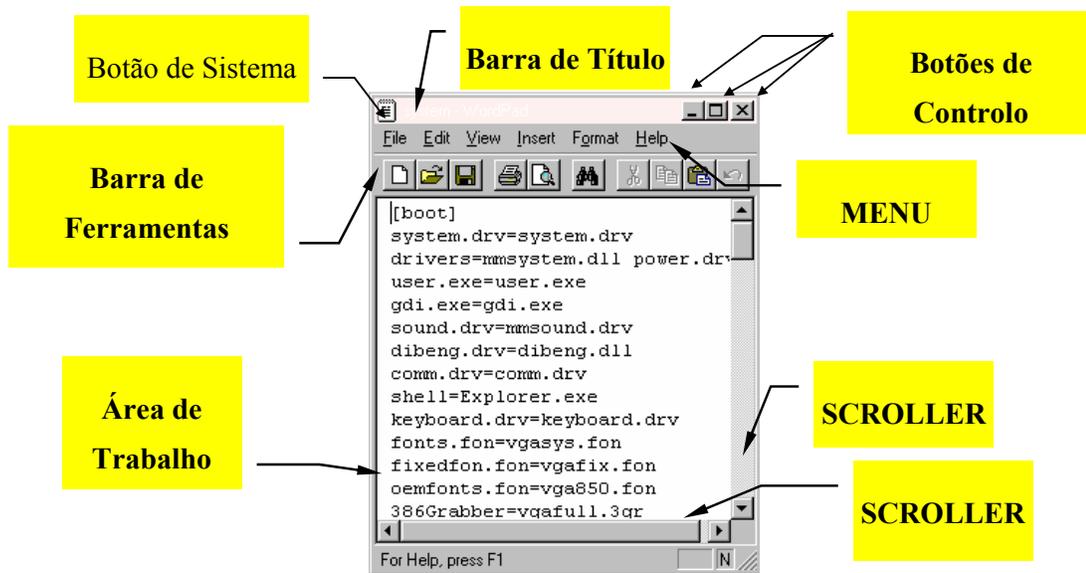


Figura 24: Elementos de uma janela típica de aplicação

As barras de título incluem três tipos de elementos: o botão de sistema, o nome da aplicação (eventualmente pode conter outro tipo de informação) e, em geral, três botões que permitem controlar a janela.

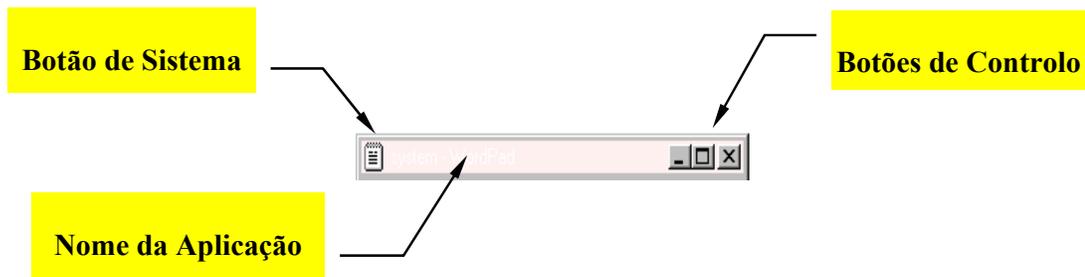


Figura 25: Barra de título de uma janela de aplicação

O botão de controlo serve para minimizar a janela em que está integrado. Corra a aplicação *Recycle Bin* e prima o botão da sua janela. Se tiver executado correctamente o procedimento, a janela da aplicação desaparece do seu *DeskTop*. Contudo, repare que a barra de tarefas continua a exibir o botão relativo a esta aplicação. Isto significa que, embora não se encontre visível a sua janela, a aplicação continua a correr. Ora, se a aplicação está a correr, é quase certo que

necessitará eventualmente de a activar novamente. Como é que poderá então restaurar a aplicação? Esta operação é realizada de forma análoga à comutação de aplicações, recorrendo-se à barra de tarefas. Prima o botão da aplicação nesta barra. Repare que surge novamente a janela no mesmo local e com as mesmas dimensões que exibia antes da sua minimização.

O botão  permite terminar a aplicação associada à janela. Prima esse botão na barra de título da janela do Recycle Bin. Repare que à semelhança do que aconteceu quando premiu , a janela desapareceu do seu *DeskTop*. Só que desta vez não lhe será possível restaurá-la, já que não surge na barra de tarefas nenhum de botão relativo a esta aplicação. Este facto significa que o programa terminou.

O segundo botão da barra de título pode tomar diferentes aspectos ( e ) consoante a janela esteja ou não maximizada (ocupe todo o ecrã). Quando executa uma aplicação, a sua janela pode, logo à partida, ocupar toda a área do ecrã do computador. Se esta não ocupar toda a área visível, então o segundo botão que surge na barra do título será , podendo maximizar a janela premindo-o. Corra novamente o programa Recycle Bin e prima o botão de maximização. Repare que este botão é alterado para  (vide figura 9a). Este novo botão permite restaurar a janela ao seu tamanho e localização anteriores. Realize, então, esta operação. O que aconteceu (vide figura 9b)? Pois é, a janela foi restaurada e o botão  foi novamente alterado para  (note que agora volta a fazer sentido poder maximizar-se a janela).

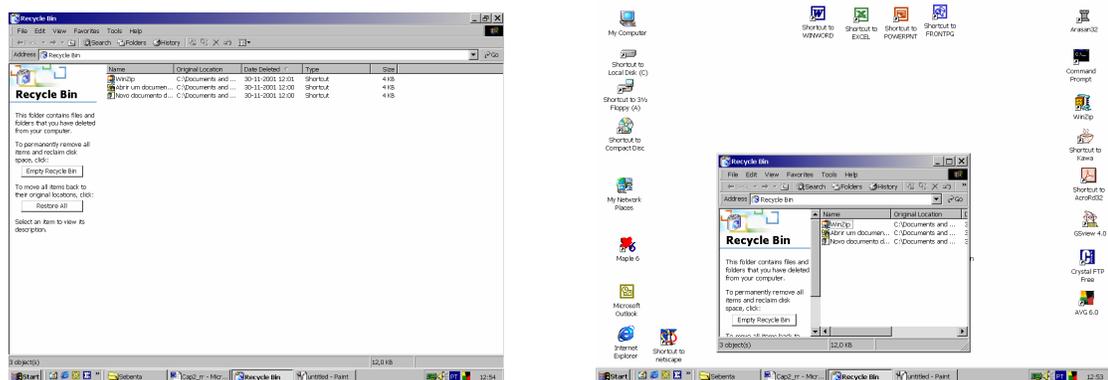


Figura 26: Maximização e restauração de uma janela

O botão de sistema encontra-se sempre no canto superior esquerdo da janela, sendo a figura uma réplica do ícone da aplicação. Quando este botão é premido uma vez, abre-se um menu com alguns comandos básicos de sistema (vide figura 10).

Os comandos *RESTORE*, *MAXIMIZE* e *MINIMIZE* do menu oferecem alternativa aos botões de redimensionamento já descritos. As opções *MOVE* e *SIZE* permitem, por sua vez, mover e redimensionar, respectivamente, a janela. Estas operações são, contudo, mais fáceis de realizar quando, como será explicado mais à frente neste texto, executadas com o rato.

Finalmente, o último comando presente no menu de sistema é o comando *CLOSE* que permite terminar, à semelhança do botão , a aplicação.

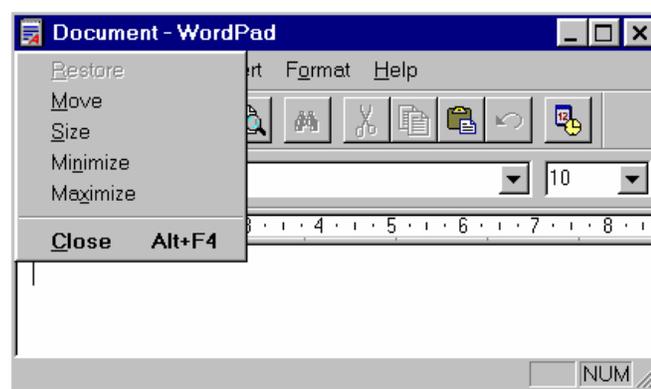


Figura 27: O menu de sistema

### 6.1 Redimensionamento de Janelas

Já vimos que é possível minimizar, maximizar e restaurar (tamanhos de) janelas. Contudo, imagine que pretende alterar as dimensões propostas pelo Windows para uma dada janela. Será possível realizar esta tarefa? Desde que esta não esteja nem maximizada nem minimizada, isto é, não ocupe a área toda do *DeskTop*, basta posicionar o apontador sobre um dos seus bordos até que este passe do formato  para um dos seguintes formatos: , ,  ou . Na figura 11 exemplifica-se esta operação para o bordo lateral direito da janela.

Encontrando-se o apontador na situação descrita, deve-se pressionar o botão esquerdo do rato e arrastar o apontador até à dimensão da janela ser a pretendida. Só então é que se deve largar o botão do rato previamente pressionado. Note-se que as dimensões da janela só podem ser alteradas nas direcções expressas pelos

apontadores. Isto é, se, por exemplo, o apontador for o ↔, então só é possível alterar a largura da janela.

**Exercício 1:** Execute a aplicação Recycle Bin e realize as seguintes operações sobre a sua janela: maximizar, minimizar, restaurar, redimensionar em todas as direcções e fechar a aplicação.

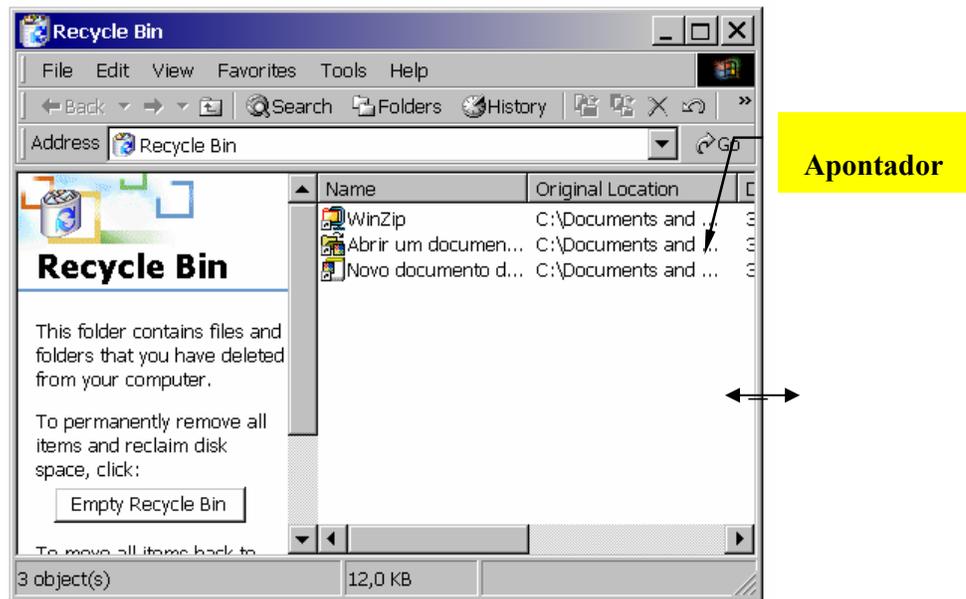


Figura 28: Alteração das dimensões de uma janela de aplicação

## 6.2 Deslocamento de Janelas

Pode também acontecer que se pretenda mudar a janela de posição no *DeskTop*. Para o efeito será obviamente necessário que a janela não se encontre nem maximizada nem minimizada. Para realizar esta tarefa, deve-se colocar o apontador sobre a barra de título da janela e, pressionando o botão esquerdo do rato, arrastá-la para a posição pretendida (vide figura 12).

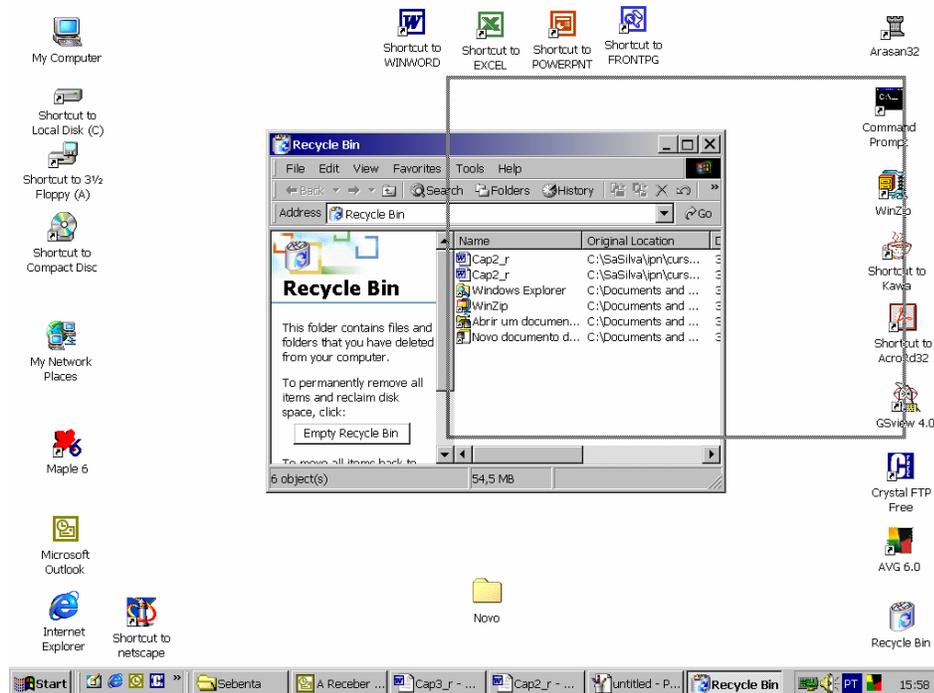


Figura 29: Mudar uma janela de posição no *DeskTop*

**Exercício 2:** Dimensione a janela do Recycle Bin para cerca de um quarto da área do seu monitor e coloque-a no canto superior direito do mesmo.

**Exercício 3:** Execute as aplicações *MyComputer*, *Network Neighborhood* e *Explorer*. Dimensione e mova as janelas das quatro aplicações que se encontram a correr no seu computador de modo a que cada uma ocupe aproximadamente  $\frac{1}{4}$  da área do ecrã, evitando que estas se sobreponham entre si.

### 6.3 Os Menus

O MENU é um elemento fundamental para a interacção com uma aplicação, dado que é nele que se encontram todos os comandos que são possíveis de realizar sobre a aplicação em causa. A selecção de um comando pode ser efectuada de duas formas alternativas: a primeira, por utilização do apontador e, a segunda, por combinação de teclas.

A manipulação de MENUS com o apontador já foi introduzida aquando da explicação do botão *START* (pois é, o botão *START* dá acesso a um MENU). Quanto à manipulação de MENUS a partir do teclado, esta é realizada da seguinte forma: uma opção do MENU selecciona-se premindo a tecla ALT em simultâneo com a letra sublinhada do item pretendido. Por exemplo, para activar a opção FILE do MENU premia ALT F.

Quando realizar a operação sugerida, surge-lhe uma janela de comandos (SUB-MENU de *FILE*, vide figura 13) em que a palavra chave *FILE* está seleccionada. O mesmo acontecerá se escolher uma outra opção do MENU. Qualquer um dos comandos pode ser seleccionado usando as setas verticais. Alguns destes comandos podem ser SUB-MENUS - exibem do lado direito uma seta ►. Para mandar executar um destes comandos deve:

- Seleccioná-lo (usando as teclas verticais) e premir *ENTER*.
- Premir o respectivo caracter sublinhado (exemplo: “C” ou “c” para o comando Close).
- Premir a(s) tecla(s) alternativa(s), quando existe(m) (exemplo: “CTRL A” para o comando *Select All* no Menu *EDIT*)

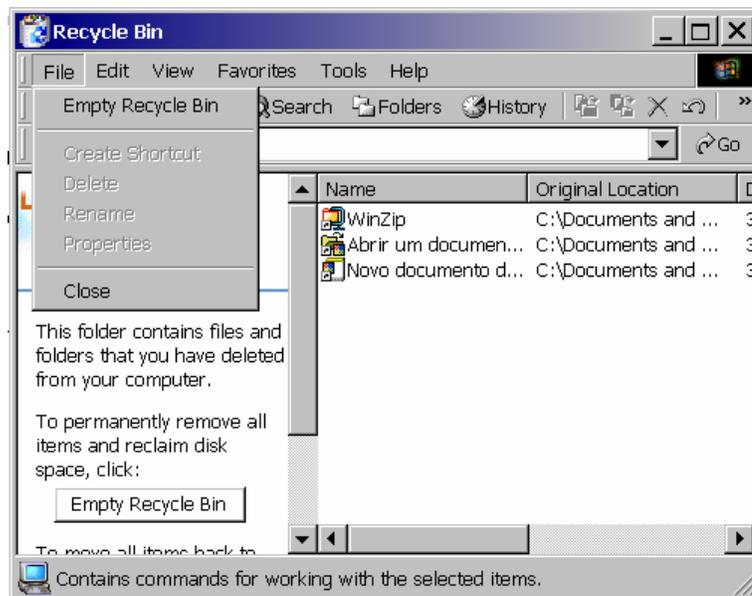


Figura 30: Sub-Menu *FILE*

Execute o comando *Select All* das diversas formas que aqui enunciamos.

## 6.4 Os Comandos

Existem fundamentalmente três tipos de comandos nos MENUS:

- **Os comandos de Execução Imediata:** identificado por um nome e eventualmente por um conjunto de teclas alternativas (exemplo: ). Estes comandos, quando activados, executam as respectivas operações sem necessitar de introdução de dados. Por exemplo, o comando *CLOSE* não necessita, para a sua execução, de nenhuma introdução de dados, uma vez que simplesmente vai fechar a aplicação.
- **Os comandos activos:** repare que o comando *STATUS BAR* do menu *VIEW* da figura 14 exibe o símbolo . Este símbolo serve para indicar ao utilizador que o comando está ligado, isto é, produz um efeito permanente enquanto a aplicação estiver a correr.
- **Os Sub-Menus:** Estes itens são facilmente identificáveis por uma seta do lado direito do nome do comando. Quando executados, abrem um novo SUB-MENU de comandos (vide figura 14).

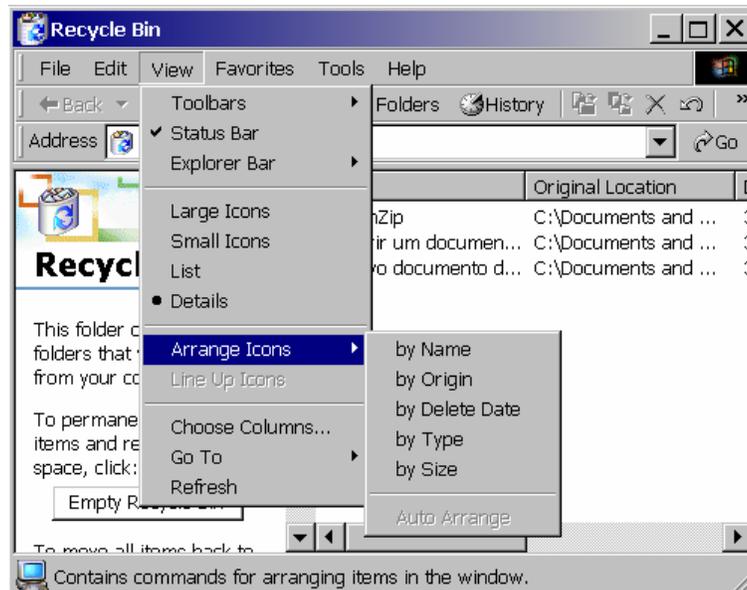


Figura 31: Comandos de Sub-Menu.

- **Os Comandos de Execução não Imediata:** Estes comandos são facilmente identificáveis pela existência de três pontos (•••) do lado direito do seu nome. Quando executados, abrem novas janelas (janelas de diálogo), que são específicas para a introdução de dados (vide figura 15).

**Exercício 4:** Termine todas as aplicações que estão a correr no seu computador (excepto o Windows 2000). Execute a aplicação *WordPad* que se encontra em *START-ACESSORIES*. Abra o menu *FILE*. Quais dos seus comandos são de execução imediata, não imediata e sub-menus? Mandar executar o comando *Open* da forma que achar mais conveniente e edite o ficheiro *c:\windows\system.ini* (escreva o nome do ficheiro no campo *FILE NAME* da janela que se abre e preme o botão  usando o apontador e o botão esquerdo do rato).

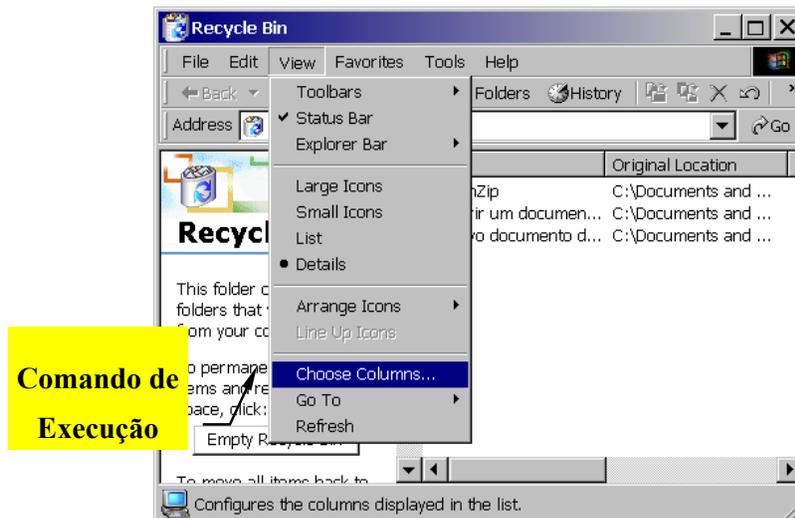


Figura 32: Comandos de execução não imediata

## 6.5 Os MENUS Auxiliares

Existe em Windows uma outra categoria de menus que ainda não foram explicados: os MENUS AUXILIARES. Os menus auxiliares (quando existem) são acedidos colocando o apontador sobre o objecto pretendido e premindo o botão direito do rato



. Estes menus dão, regra geral, acesso a um conjunto de comandos relacionados com o objecto em causa.

Feche todas as aplicações que estão a correr no seu *DeskTop* e posicione o apontador sobre o ícone do programa *Recycle Bin*. Prima o botão do lado direito do seu rato.

Repare que surge uma janela por baixo do apontador com um conjunto de comandos que pode realizar sobre este ícone (vide figura 16). Por exemplo, pode executar a aplicação, seleccionando o comando *OPEN*. Experimente!



Figura 33: O Menu Auxiliar

## 6.6 A Área de Trabalho

A área de trabalho de uma aplicação encontra-se sempre na respectiva janela e corresponde, geralmente, à zona onde são visualizados os dados desse programa. Por exemplo, num processador de texto a área de trabalho é a zona da janela onde o texto é editado.

Acontece frequentemente que, devido ao excesso de dados ou devido às dimensões reduzidas da janela, não é possível visualizar em simultâneo todos os dados. Nestas circunstâncias, surgem na janela, geralmente nas partes inferior e lateral direita, duas barras (*Scrollers*) que nos permitem deslocar a área de trabalho visível na janela (vide figura 17).

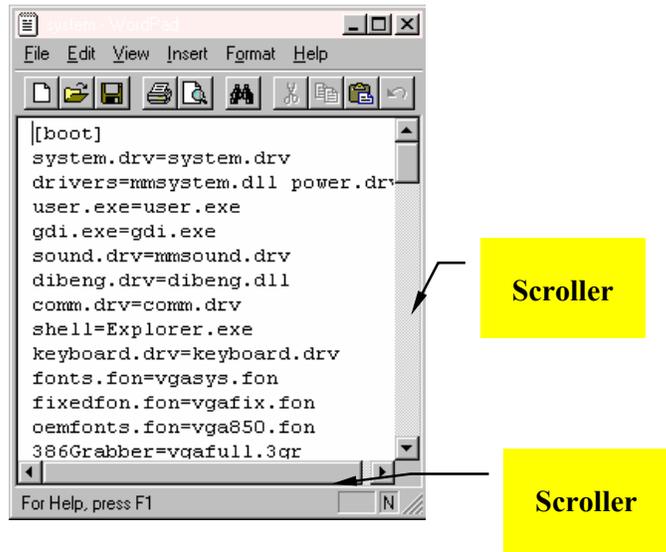


Figura 34: Os SCROLLERS de uma janela.

Execute a aplicação *WordPad* que se encontra em *START-ACESSORIES* e edite o ficheiro *c:\windows\system.ini*. Redimensione a janela (se necessário) até surgirem as referidas duas barras (vide figura 17). Repare que nas extremidades das barras se encontram duas setas (▲ e ▼ nas barras verticais; ◀ e ▶ nas barras horizontais). Estas setas servem para deslocar, nos respectivos sentidos, a janela sobre a área de trabalho. A ideia subjacente é a indicada na figura 18.



Figura 35: Deslocamento da área de trabalho sobre o documento, usando os SCROLLERS

Pressione as referidas setas e verifique o seu efeito.

Outra forma de obter o mesmo propósito será executando movimentos de arrastamento das regiões rectangulares que se encontram nas barras. Verifique o efeito desta operação na sua janela. Repare que os deslocamentos horizontais e verticais já são bastante mais acentuados.

## 6.7 Os Diálogos

São muitas as circunstâncias em que o utilizador tem que estabelecer um diálogo (troca de informação) com a aplicação em que se encontra a trabalhar. Por exemplo, imagine que pretende abrir um documento num editor de texto. Para além de correr a aplicação de edição, ter-lhe-á, ainda, que indicar o nome do documento que deseja manipular.

No Sistema Operativo Windows 2000 a troca de informação entre o utilizador e uma aplicação estabelece-se por intermédio de uma janela de diálogo, podendo este realizar-se pela introdução explícita de informação (ex: o nome de um documento) ou pela selecção de um conjunto de opções permitidas.

Praticamente todos os diálogos exibem três botões que permitem confirmar ou anular as opções escolhidas:

-  - Permite confirmar todas as opções escolhidas no diálogo, sendo este fechado.
-  - Permite anular as opções escolhidas no diálogo, sendo este fechado.
-  - Permite confirmar todas as opções escolhidas no diálogo, continuando este aberto.

### 6.7.1 As Caixas de Texto

Uma caixa de texto permite introduzir directamente informação, por exemplo, o nome de um ficheiro, no formato de texto que será fornecido ao programa. Estes elementos de comunicação identificam-se facilmente pela sua forma rectangular () que pode, à partida, conter já algum texto. Para alterar o conteúdo de uma caixa de texto, coloque o apontador  sobre a caixa pretendida e prima uma vez no botão esquerdo do rato . Surge, então, um cursor vertical | na

caixa, permitindo apagar o texto que nela consta e inserir o pretendido (vide figura 19).



Figura 36: Um Diálogo

**Exercício 5:** Execute o programa *WordPad* que se encontra em *PROGRAMS-ACESSORIES* e execute o comando *OPEN* do menu *FILE*. Preencha o diálogo com a informação necessária à abertura do ficheiro de nome *d:\users\focus.txt* (vide figura 20).

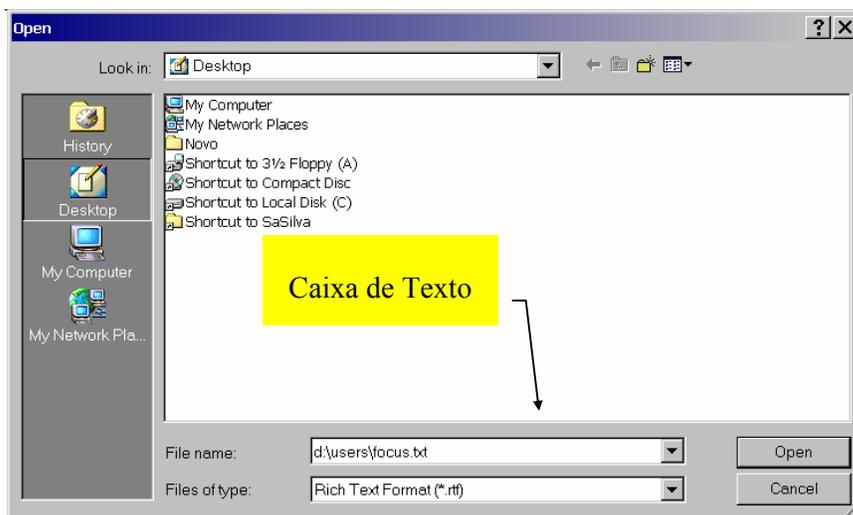


Figura 37: O diálogo para abrir um ficheiro

### 6.7.2 As Caixas de Listagem

Quando o domínio de uma determinada configuração é limitado, por exemplo, na selecção do tipo de papel para uma impressora, não faz sentido o programa permitir ao utilizador especificar a sua opção pela introdução de texto, até porque este não conhece, regra geral, todas as opções ao seu dispor. Nestes casos, as várias alternativas disponíveis são listadas em caixas de listagem que podem ter dois aspectos distintos (*List Boxes* e *Combo Boxes*).

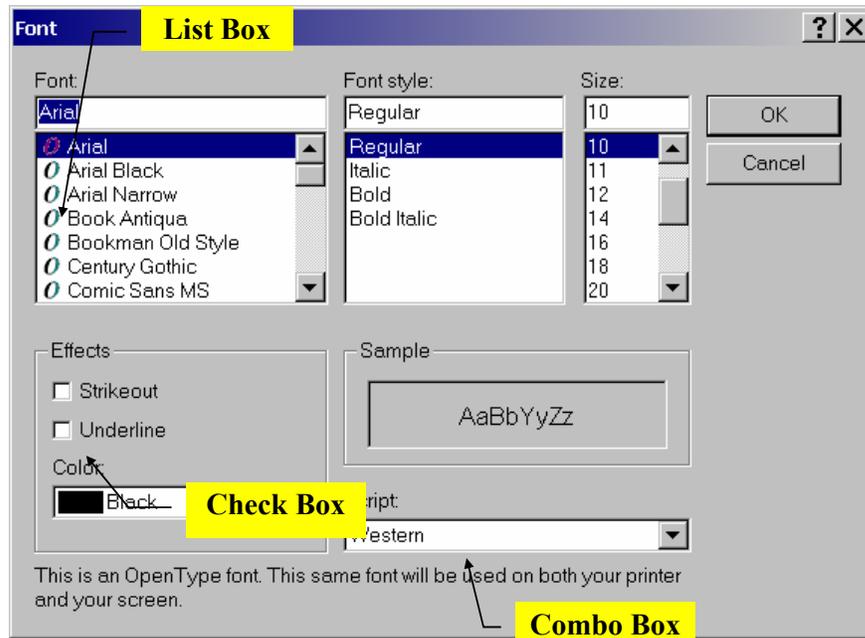


Figura 38: Diálogos com caixas de listagem

Nas caixas de listagem do tipo *List Box*, o utilizador escolhe a opção pretendida:

1. usando o *Scroller* vertical (se este existir), visualiza a opção pretendida. Por exemplo, para visualizar o tipo de carácter (*FONT*) *ARIAL* no diálogo da figura 21 é necessário premir o botão  do *Scroller* associado até que este tipo de letra seja visível;
2. colocar o apontador sobre a opção pretendida e premir o botão esquerdo do rato.

O segundo tipo de caixas de listagem (*Combo Box*) identifica-se facilmente pela existência de um botão  do lado direito do campo (vide figura 21). Para poder alterar a opção pretendida (a caixa visualiza sempre a opção corrente) deve premir este botão, operação essa que conduz à abertura de uma nova janela sobre o diálogo com as opções disponíveis (vide figura 22). À medida que desloca o apontador nesta janela, a opção sob este é seleccionada exibindo um fundo distinto das demais. A confirmação da sua escolha estabelece-se simplesmente premindo o botão esquerdo do rato, ficando o campo preenchido com a opção seleccionada.

**Exercício 6:** Execute a aplicação *WORD* e corra o comando *FONT* do menu *FORMAT*. Preencha o diálogo que lhe surge com a informação que se exhibe na figura 23.

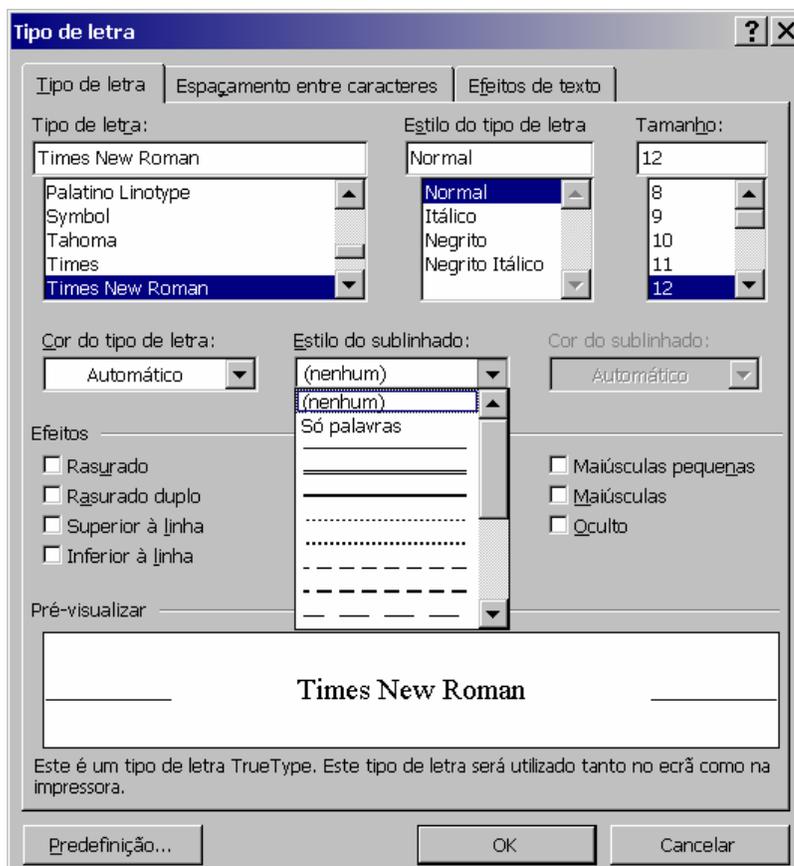


Figura 39: Controlo de uma *Combo Box*

### 6.7.3 Os Botões de Selecção (*Radio Buttons*)

Um outro elemento que ocorre frequentemente nas caixas de diálogo são os botões de selecção (vide figura 24). Estes botões identificam-se facilmente pelo seu aspecto redondo característico . Para seleccionar uma dada opção com estes elementos de comunicação, basta colocar o apontador sobre o botão pretendido e premir o botão esquerdo do rato. Quando realizar este procedimento, reparará que, em cada momento, somente uma opção estará seleccionada - a selecção de uma opção conduz automaticamente à eliminação da selecção anterior - encontrando-se o botão correspondente preenchido com uma bola a preto .

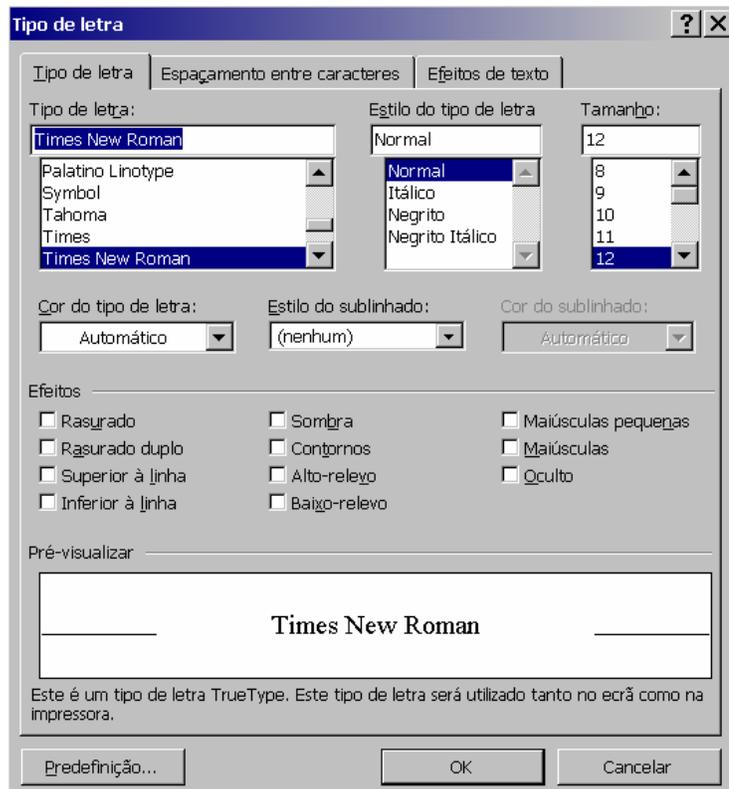


Figura 40: Diálogo exemplo

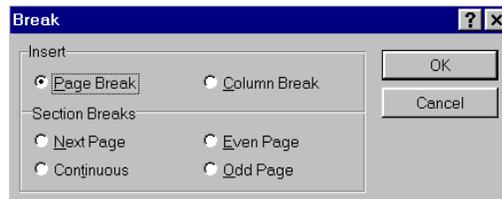


Figura 41: Os *Radio Buttons*

#### 6.7.4 Os Botões de Selecção (*Check Box*)

Um segundo tipo de botões de selecção são as *Check Box* (vide figura 21). Estes elementos identificam-se facilmente pelo seu aspecto de pequenas caixas quadradas que estão vazias  se a opção não estiver seleccionada e estarão preenchidas  se a opção estiver seleccionada. Para comutar entre o estado de selecção e não selecção, ou vice-versa, da opção, o utilizador terá que, após a

colocação do apontador sobre o elemento pretendido, premir o botão esquerdo do rato.

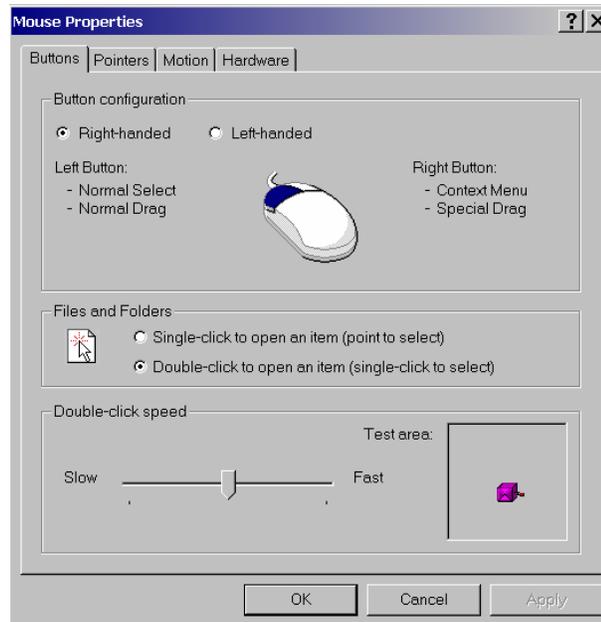


Figura 42: Diálogo com *SLIDER*

#### 6.7.5 Outros Elementos de Comunicação

Embora menos frequentes podem ainda surgir nas caixas de diálogo dois outros elementos: os *SLIDERS* (vide figura 25) e os *SPINNERS* (vide figura 26).

Quando uma caixa de diálogo apresenta uma opção em termos de intensidade (ex: cor, som, etc), muitas vezes é utilizado um *SLIDER* . Este elemento é uma barra deslizante que representa uma escala de valores. Para aumentar ou diminuir a intensidade pretendida, basta premir o botão esquerdo do rato e arrastar o apontador no sentido desejado.

Em alguns casos as opções existentes nas caixas de diálogo apresentam do lado direito dos respectivos campos (regra geral, os campos permitem a introdução de informação numérica) duas setas verticais  semelhantes às que se encontram nos *scrollers*. Estes elementos destinam-se a incrementar (seta ascendente) ou decrementar (seta descendente) o valor da caixa associada.

**Exercício 7:** Corra a aplicação WORD e execute o comando *PRINT* do menu *FILE*. Preencha o diálogo associado à operação de impressão com os seguintes dados:

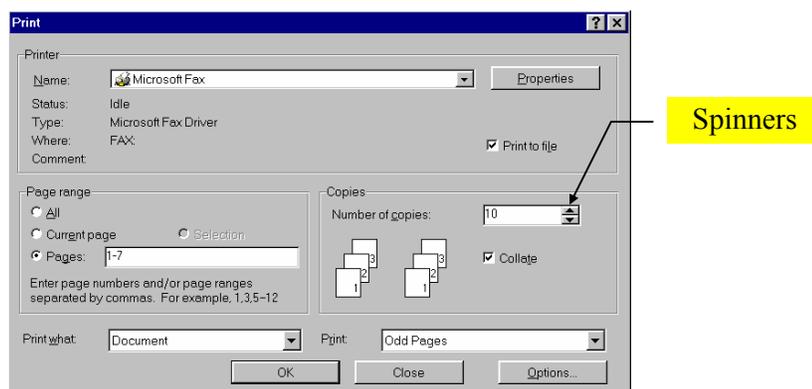


Figura 43: Diálogo exemplo

## 7. Personalização do Ambiente de Trabalho

O *Windows 2000* quando é instalado apresenta um conjunto de opções estéticas que podem não ser do agrado do utilizador. Tratando-se de um ambiente bastante versátil, este S. O. permite que o utilizador personalize o seu ambiente de trabalho. Embora existam bastantes aspectos em relação aos quais esta tarefa pode ser realizada, iremos analisar neste capítulo somente os seguintes:

- Alteração do conjunto de cores do ambiente
- Alteração do fundo do *DeskTop*
- Instalação e alteração do *Screen-Saver*

Para poder alterar as características acima enunciadas, deve abrir o menu auxiliar sobre o *DeskTop*. Para isso, desloque o apontador para uma zona não ocupada e prima o botão do lado direito do rato. Se tiver executado devidamente este procedimento, surge-lhe no ecrã o seguinte Menu:

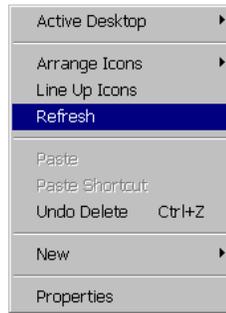


Figura 44: Menu auxiliar do *DeskTop*

Selecione o comando *Properties* e execute-o. Aparece, então, no seu monitor o diálogo da figura 28.

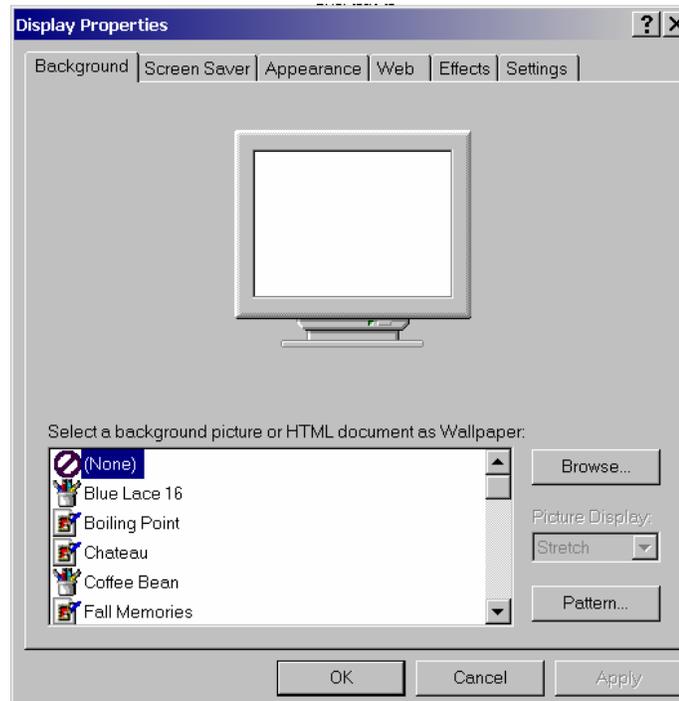


Figura 45: Diálogo que permite a personalização do ambiente de trabalho

Para mudar o fundo do seu *DeskTop* deve alterar o campo *Wallpaper*. Imagine, por exemplo, que pretendia escolher o fundo *WATER COLOR*. Como esta opção não se encontra disponível na *List Box* (caixa com as diversas opções por baixo de *WALLPAPER*) correspondente ter-se-á que servir do *Scroller* que se encontra do lado direito (prima o botão  do *Scroller*), até que a referida opção surja na área visível. Uma vez visível deve seleccioná-la deslocando o apontador sobre a mesma e premindo o botão direito do rato. Note que no centro do “ecrã exemplo”, incluído no

diálogo, aparece a imagem pretendida. Ora, acontece que esta imagem (*WATER COLOR*) é demasiado pequena para ocupar todo o *DeskTop*. Para repetir a imagem poderá seleccionar a opção *PICTURE DISPLAY TILE*. O resultado desta opção encontra-se na figura 29.

Para que a sua escolha seja permanente deve, no final deste procedimento, premir o botão  , e para invalidar a opção deverá escolher o botão  .

Se premir sobre a opção *SCREEN SAVER* o diálogo modifica-se para:

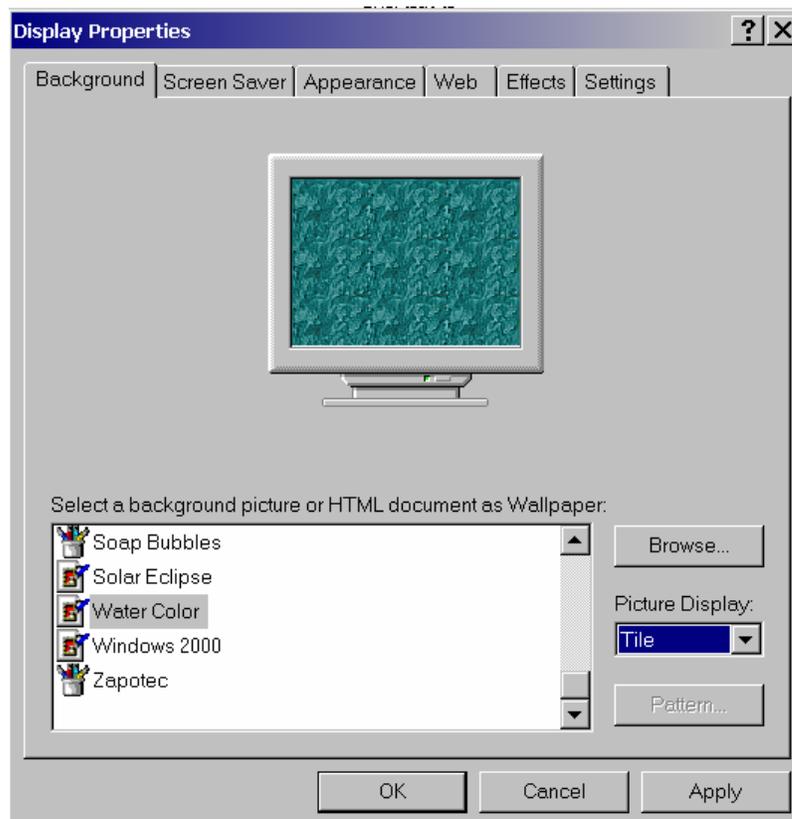


Figura 46: Resultado da alteração do fundo do *DeskTop*

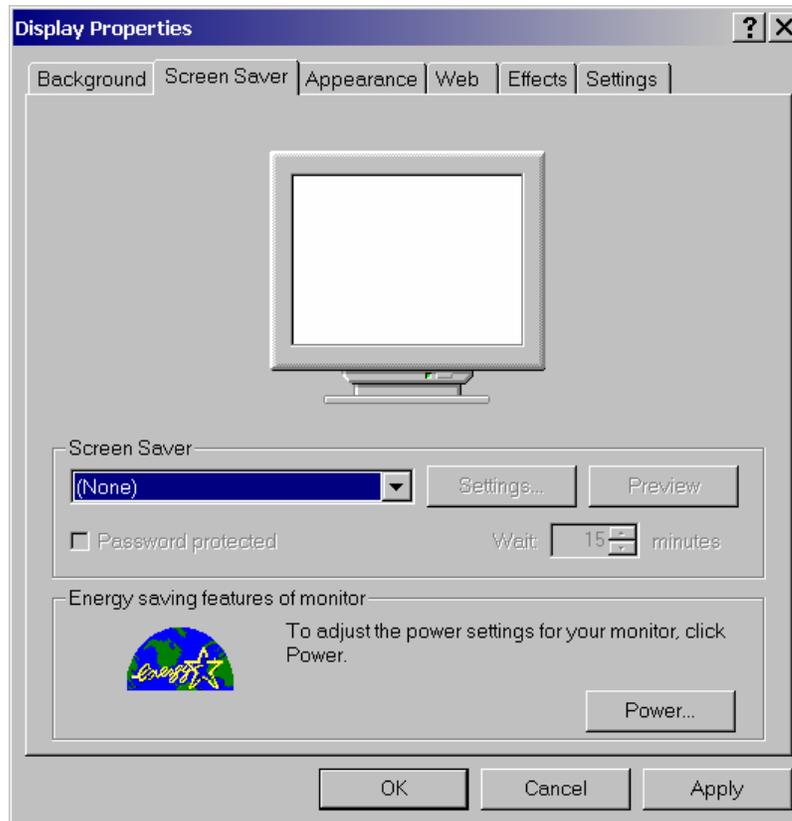


Figura 47: Alteração do *Screen Saver*

Um *SCREEN SAVER* é um programa que se auto-activa ao fim de algum tempo (especificável no campo  ), gerando, regra geral, um conjunto de animações com cores escuras que protegem o seu monitor. Assim, para escolher um determinado *SCREEN SAVER* deverá accionar o botão  do campo  . Poderá observar no monitor, desenhado no diálogo, o efeito que este proporciona. Para ter uma ideia exacta do seu efeito poderá premir o botão *preview* . Poderá ainda configurar diversos parâmetros do seu *SCREEN SAVER* accionando o botão . Por exemplo, para o *SCREEN SAVER 3D FLYING OBJECTS*, poderá configurar os seguintes parâmetros:

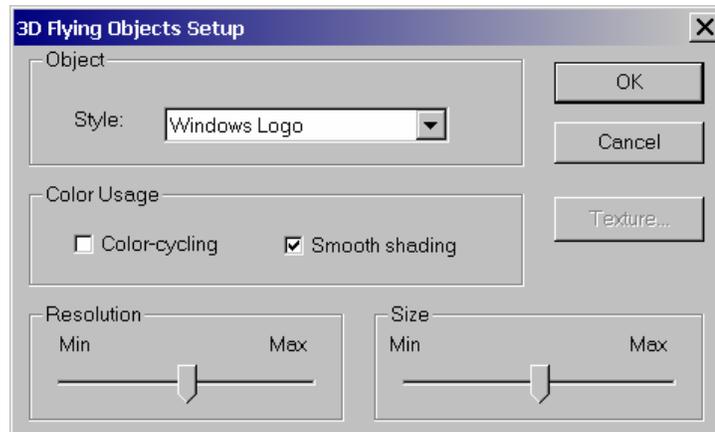


Figura 48: Alteração dos parâmetros de um *Screen Saver*

**Exercício 8:** Seleccione o *SCREEN SAVER 3D FLYING OBJECTS* e configure-o de forma a que o seu tamanho seja mínimo.

O *SCREEN SAVER* poderá ser também um elemento de segurança para o utilizador de um computador. Imagine que se encontra a trabalhar num projecto confidencial e que abandona temporariamente o seu computador. Por lapso, qualquer pessoa poderá aceder ao seu trabalho. Para evitar estas situações, o *SCREEN SAVER* permite a inclusão de uma senha (*PASSWORD*). Para isso deverá clicar sobre a *CHECK-BOX*  *Password protected*. Execute esta operação.

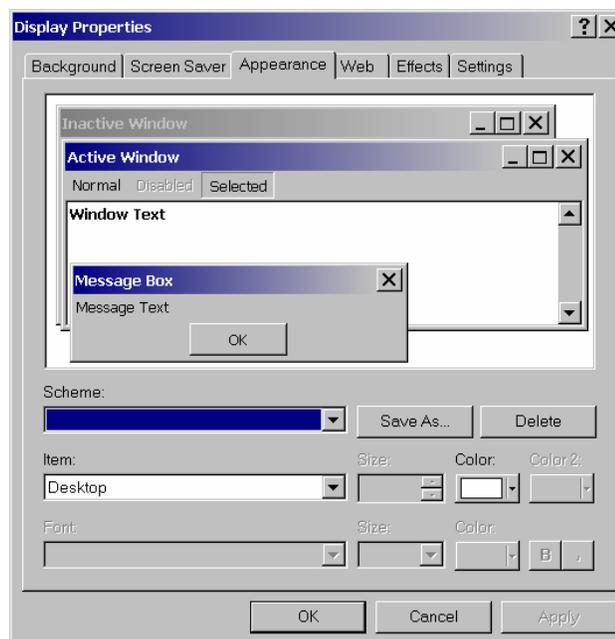


Figura 49: Alteração dos esquemas de cores

**Exercício 9:** Escolha o *SCREEN SAVER FLY AWAY* e configure-o de forma a que o *SCREEN SAVER* deverá activar-se ao fim de 1 minuto e deverá pedir uma *PASSWORD*.

O Windows 2000 dispõe de um conjunto de combinações de cores que poderá testar. Para o efeito basta que prima o botão  associado a *SCHEMES* (vide Figura 32) e seleccione um determinado esquema de cores. Seleccione o esquema *WHEAT*. Como poderá observar, as cores do “ecrã exemplo”, incluído no diálogo, mudaram para as do novo conjunto. Se, contudo, nenhum conjunto de cores lhe agradar, pode ainda alterar individualmente as cores dos diversos componentes que compõem o ambiente *WINDOWS*. Para o efeito deverá seleccionar o elemento cuja cor pretende alterar, o que poderá concretizar posicionando o apontador sobre esse elemento no “ecrã exemplo” do diálogo e premir com o botão esquerdo do rato. Por exemplo, imagine que pretendia alterar a cor das barras de título das janelas activas. Seleccione esta barra no “ecrã exemplo”. Já se terá certamente apercebido que o campo *ITEM*  surge agora com o texto *ACTIVE TITLE BAR*. Para alterar a sua cor deve premir o botão  do campo *COLOR* e seleccionar a cor pretendida. A sua escolha reflecte-se imediatamente sobre o conteúdo do “ecrã exemplo” do diálogo, podendo deste modo ser avaliado o seu efeito. Se nenhuma das cores disponíveis lhe agradar deverá premir o botão  na janela de selecção de cores. Neste caso, aparece um novo diálogo contendo o espectro de cores possíveis. Para obter uma determinada cor poderá seleccionar (movimento habitual do apontador) uma das constantes nas *BASIC COLORS* (vide Figura 33) - quadrados do lado esquerdo, ou então, clicar com o rato sobre a zona do lado direito (espectro de cores) até encontrar a que lhe parecer melhor. Repare que, à medida que realiza o movimento descrito, o campo *COLOR* exhibe a cor seleccionada. Uma vez obtida a cor pretendida, deverá confirmar a sua escolha premindo em sequência os botões  e .

**Exercício 10:** Mude as cores dos seguintes elementos: barra de título das janelas activas e desactivadas, zona de trabalho das janelas, menu e botões. Para o efeito, recorra a cores que não estejam definidas à partida.

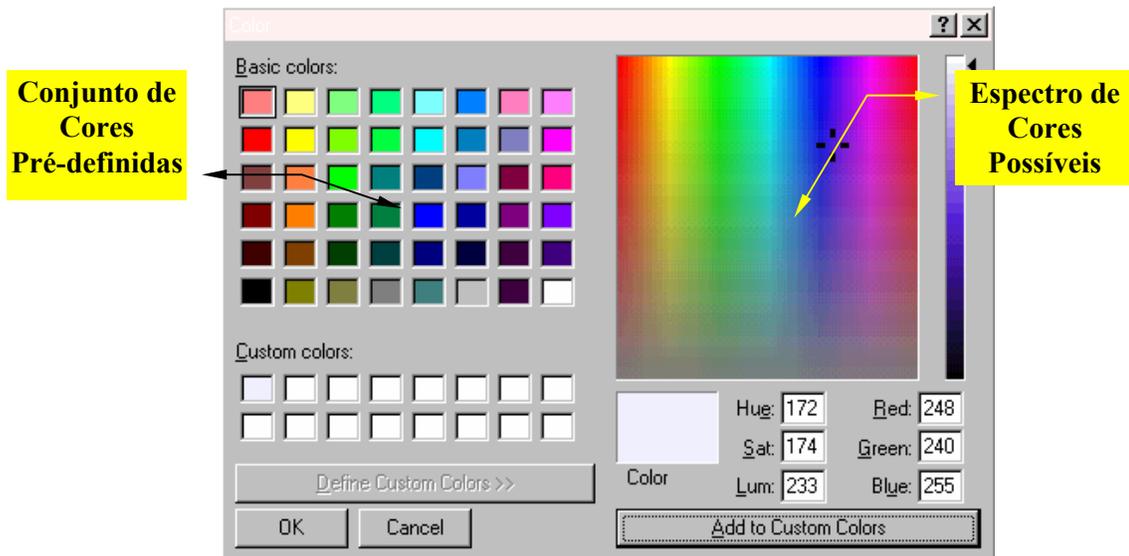


Figura 50: Diálogo para a escolha de cores.

## 8. Fechar uma Sessão Windows

Após um dia de trabalho, chega o momento de desligar o computador. Será, então, o momento de fechar todas as suas aplicações, premindo o botão  ou executando o comando *CLOSE* do *MENU FILE*. Ora, mas onde está esta opção para o *DeskTop* do Windows 2000? Será que os técnicos da *MicroSoft* se esqueceram de a incluir no Sistema Operativo?

Neste momento estará provavelmente a ponderar a hipótese de simplesmente desligar o interruptor de alimentação do seu computador. Nunca o faça nestas circunstâncias; os resultados podem ser bastante desagradáveis, uma vez que o *DeskTop* mantém um conjunto de ficheiros abertos que desta forma poderão ser destruídos. Para além do que foi atrás descrito, verifica-se que o Windows ao encerrar uma sessão verifica se existem programas a correr, “alertando” para o facto se os respectivos dados ainda não tiverem sido salvaguardados.

Para sair de uma forma “limpa” do Windows 2000, prima o botão  da barra de tarefas e seleccione a opção *Shut Down* do respectivo *MENU*. Se o fizer, surge-lhe no ecrã o seguinte diálogo:



Figura 51: Diálogo de fim de sessão.

Para desligar o computador deve premir o botão OK e para anular esta acção o botão Cancel.

## 9. Pesquisar na Ajuda

Foi objectivo deste capítulo introduzir alguns conceitos básicos que permitam ao leitor operar no ambiente Windows 2000. Para mais informações deve consultar o programa de ajuda integrado neste ambiente. Para tal, seleccione a opção *HELP* do MENU START . Se o fizer, surge-lhe no ecrã o seguinte diálogo:



Figura 52: Diálogo para pesquisa de ajuda

Existem fundamentalmente duas formas de pesquisar ajuda:

- Por palavra chave (usar a opção Search ou Index)
- Por assunto (usar a opção Contents)

Para pesquisar a ajuda por palavra chave, deve introduzir a palavra e seleccionar uma das possíveis respostas, carregando em DISPLAY (Figura 36).



Figura 53 - Pesquisa

Na opção *Contents* Contents do diálogo de ajuda, a pesquisa é realizada de forma análoga à de um livro. Existem diversos livros (por exemplo:  *Introducing Windows*) cada um versando sobre um determinado assunto. Cada livro, uma vez aberto -  premir duas vezes com o botão esquerdo do rato sobre este, dá acesso a novos livros  e documentos . Os documentos serão editados se forem “executados” (operação semelhante à execução de um programa a partir do seu ícone). Na Figura 37 exemplifica-se este procedimento para o tópico *Hardware overview*.



Figura 54: Janela com a descrição de ajuda sobre o item seleccionado.

# CAPÍTULO III

## Configuração do Ambiente de Trabalho *Windows* 2000

*Neste capítulo serão descritos ao leitor os mecanismos necessários à configuração do ambiente de trabalho no sistema Windows 2000.*

---

### 1. Introdução

São inúmeras as ocasiões em que as configurações que o *Windows* sugere à partida não se adequam nem ao ritmo, nem ao tipo de trabalho de um determinado utilizador. Imagine, por exemplo, de que dispõe de um ecrã grande. Para tirar partido do seu investimento, o ideal será naturalmente ter a oportunidade de visualizar, em simultâneo, mais informação, isto é, aumentar a “resolução” do ecrã.

Sendo este sistema operativo bastante versátil, é permitido ao utilizador reconfigurar, de uma forma fácil e visual, um conjunto de opções de funcionamento que o tornam bastante mais agradável de utilizar. Neste capítulo iremos explorar algumas das opções de configuração mais significativas deste sistema operativo.

### 2. O Control Panel

Como já foi referido, embora o *Windows* possa ser utilizado na sua versão estandardizada, são inúmeras as opções que podem ser personalizadas. Estas opções encontram-se agrupadas no chamado “CONTROL PANEL” que se encontra no menu *START / SETTINGS*. Para correr o *Control Panel* abra o submenu *SETTINGS*  do menu *Start*  (vide figura 55) e execute o comando *CONTROL PANEL*  .

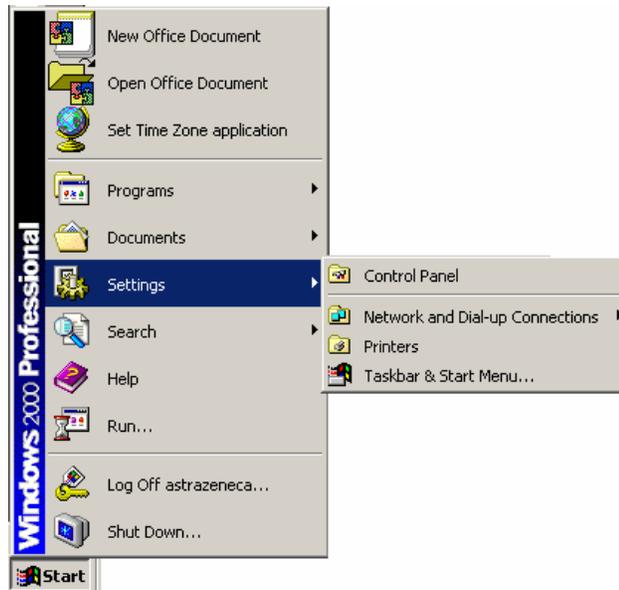


Figura 55: O submenu Settings.

Se tiver realizado devidamente o procedimento descrito, surge-lhe no ecrã a seguinte janela com as ferramentas de configuração à sua disposição:

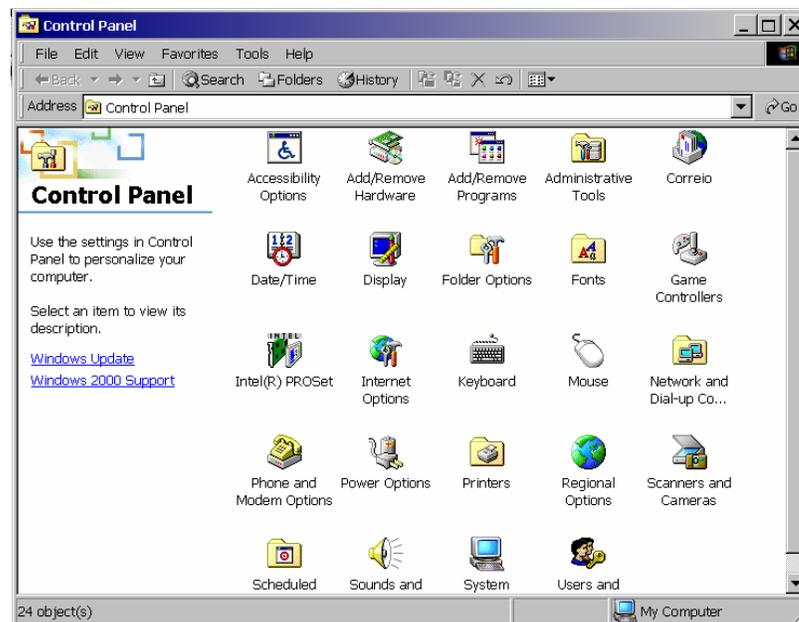


Figura 56: O Control Panel.

Uma outra forma de aceder às ferramentas do *Control Panel* é a partir do *Explorer*. Para tal deverá seleccionar o recurso *Control Panel* na área de recursos do *Explorer* conforme é sugerido na figura 57.

Conforme é possível observar nas figuras 56 e 57, o *Control Panel* é composto por uma série de utilitários, destinando-se, cada um, a um aspecto específico da configuração do sistema. O número de utilitários varia de sistema para sistema em função do *Software* instalado:



**Accessibility Options:** Disponibiliza um conjunto de opções de configuração para utilizadores com determinados tipos de deficiências.



**Add New Hardware:** Esta ferramenta permite ao utilizador acrescentar e configurar novo hardware, por exemplo, uma placa de som, no seu sistema.



**Add / Remove Programs:** Esta ferramenta deve ser sempre utilizada para instalar ou remover programas do sistema, já que permite realizar de uma forma automática a gestão de todos os ficheiros (próprios da aplicação e comuns a outras aplicações) dos diversos programas instalados no sistema.



**Date / Time:** Esta ferramenta permite configurar a hora e a data do sistema.



**Display:** Esta ferramenta define as cores e a apresentação do ecrã do sistema.



**Fonts:** Esta ferramenta serve para gerir as fontes (tipos de caracteres) existentes no sistema, quer para o ecrã, quer para as diversas impressoras que a este se encontram eventualmente ligadas.



**Internet:** Permite configurar o acesso via modem à internet.



**Keyboard:** Esta ferramenta permite configurar a velocidade do teclado e a sua disposição.



**Mail:** Acrescenta perfis para utilização com aplicações de comunicação.



**Modems:** Esta ferramenta permite ao utilizador do sistema gerir e configurar o modem, se existir, para comunicações (Internet, Fax, etc.)



**Mouse:** Permite adaptar a manipulação do rato às necessidades de cada utilizador.



**Network:** Esta ferramenta permite realizar a instalação e configuração dos diversos módulos (protocolos) de comunicação necessários para aceder aos diversos tipos de redes existentes (*Windows*, internet, etc.).



**Passwords:** Permite definir perfis de utilizadores consoante a senha introduzida no arranque do sistema.



**Printers:** Esta ferramenta deve ser sempre utilizada para adicionar novas impressoras ao sistema, podendo ainda servir para a sua configuração.



**Regional Settings:** Esta ferramenta permite ao utilizador definir diversos parâmetros próprios de cada país, como, por exemplo, unidade monetária, separador de números decimais, etc.



**Sounds:** Este utilitário permite ao utilizador configurar os sons do sistema.,



**System:** Esta ferramenta permite realizar a configuração de baixo nível do sistema, não devendo ser manipulada por utilizadores sem conhecimentos avançados na área da informática. A sua utilização indevida poderá bloquear o sistema.



Figura 57: Acesso ao *Control Panel* a partir do *Explorer*.

## 2.1 Execução de uma Ferramenta do Control Panel

Para executar uma qualquer ferramenta do *Control Panel*, posicione o apontador do rato sobre o ícone da ferramenta pretendida e realize o clique duplo do botão esquerdo do rato habitual para executar uma aplicação.

**Exercício 11:** Abra o *Control Panel* a partir do MENU *START* e execute a ferramenta *Mouse*.

**Exercício 12:** Repita o exercício anterior acedendo desta vez ao *Control Panel* a partir do *Explorer*.

## 3. Alteração da Data e da Hora do Sistema

Muitos programas usam para gerir os seus ficheiros a data e a hora da sua última actualização. Assim, reveste-se de extrema importância que o sistema esteja sempre a funcionar com a data e hora correctas. Para manipular estas variáveis o operador terá que executar a ferramenta *DATE/TIME*  do Control Panel. Execute-a! Se tiver realizado correctamente o procedimento, surge-lhe o seguinte diálogo no ecrã:

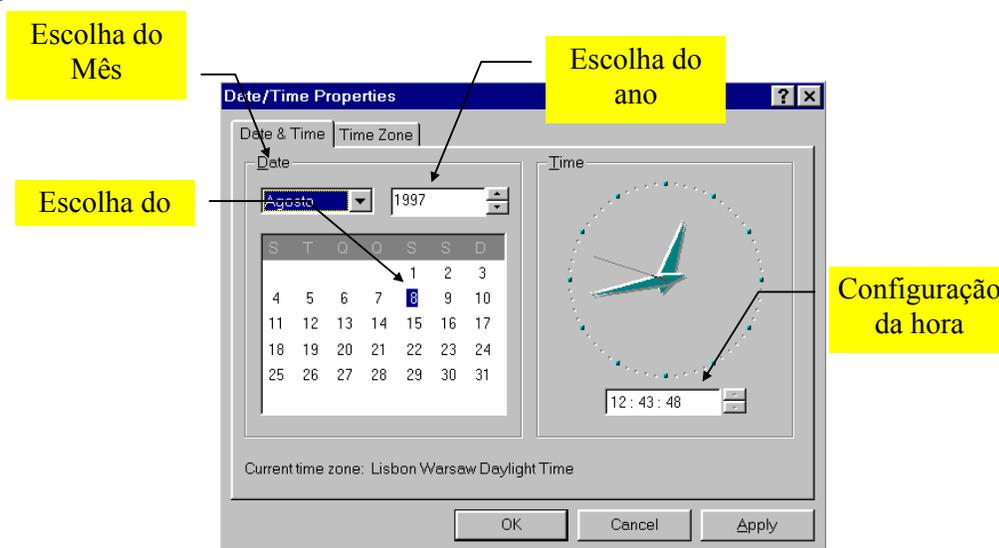


Figura 58: Diálogo para aceder ao relógio do sistema.

Na parte superior do diálogo existem dois campos que lhe permitem configurar as componentes ano e mês da data do sistema. Para mudar o mês do sistema, abra a *Combo-Box* (vide figura 59) e seleccione o mês correcto. Para o ano dispõe de um *Spinner* junto ao campo correspondente que poderá accionar para o modificar. Para alterar a componente dia da data deverá posicionar o apontador do rato sobre o dia correcto na janela de calendário que é exibida no diálogo e clicar uma vez no botão esquerdo do rato de modo a selecciona-lo.

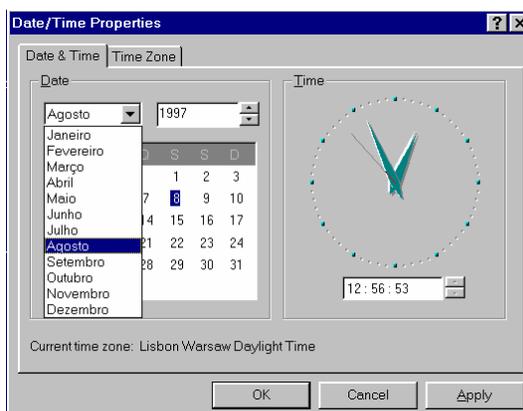


Figura 59: Alteração do mês da data do sistema.

Para alterar a hora do sistema posicione o cursor (apontador) sobre a componente (hora, minutos ou segundos) do campo da hora, apague o seu conteúdo usando a tecla *DEL* do teclado e introduza o valor pretendido. Para alternar entre as diversas componentes da hora poderá usar as teclas *TAB*, para seleccionar da esquerda para a direita e *SHIFT TAB* para seleccionar no sentido inverso.

**Exercício 13:** Configure a data e hora do sistema: Hora - 12:09:35; Data - 24 Dezembro 1998.

#### 4. O Teclado

O utilitário *Keyboard* permite ao utilizador ajustar a velocidade com que o teclado repete os mesmos caracteres, bem como definir a velocidade com que pisca o cursor, a língua do teclado e o respectivo modelo de *hardware*. Para poder alterar os

parâmetros atrás referidos terá que executar a partir do *Control Panel* a ferramenta *Keyboard* . Se o fizer surge-lhe a seguinte caixa de diálogo no ecrã:

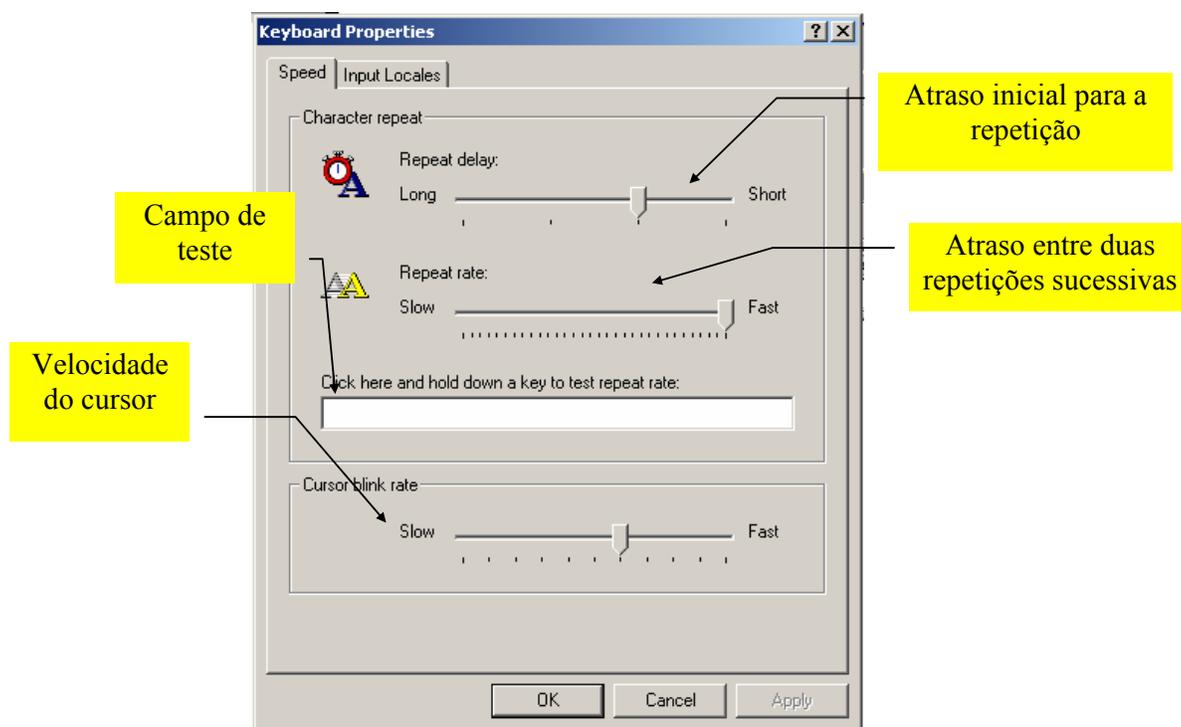


Figura 60: Ajuste de parâmetros do teclado.

Para variar um *Slider* tem que colocar o apontador do rato sobre este e premir o botão esquerdo do rato, realizando de seguida o movimento de arrastamento pretendido - permitem-lhe configurar o atraso inicial de repetição, isto é, o período de tempo que é necessário manter uma tecla premida antes que esta seja repetida pela primeira vez, e o intervalo entre repetições sucessivas. O último *Slider* controla a velocidade a que o cursor piscará nas diversas aplicações *Windows*. Dispõe, ainda, de um campo onde poderá testar o efeito das opções escolhidas.

Como sempre, para que as suas opções se tornem permanentes terá que premir o botão  e para as anular terá que premir o botão .

**Exercício 14:** Posicione o *slider* do diálogo nas posições indicadas na figura 60. Experimente o efeito no campo de teste. Coloque agora os *sliders* no extremo oposto e experimente novamente o efeito.

#### 4.1 Definição do Modelo e Língua do Teclado

Os diferentes idiomas requerem caracteres diferentes. Por exemplo, o português requer a existência de acentos e do ç ao passo que o alemão necessita de algumas vogais com trema. Suponha que trabalha frequentemente em línguas diferentes, terá certamente a necessidade de alternar rapidamente entre as diferentes línguas do teclado. Sendo o *Windows* um sistema operativo multilingue, este permite que o operador defina o tipo de língua do teclado, a qual condiciona a disposição dos caracteres no mesmo, e alterne rapidamente entre diversas línguas.

Para poder configurar esta característica do seu teclado active a ferramenta *Keyboard* do *Control Panel* e seleccione o separador *Input Locales* (vide figura 61).

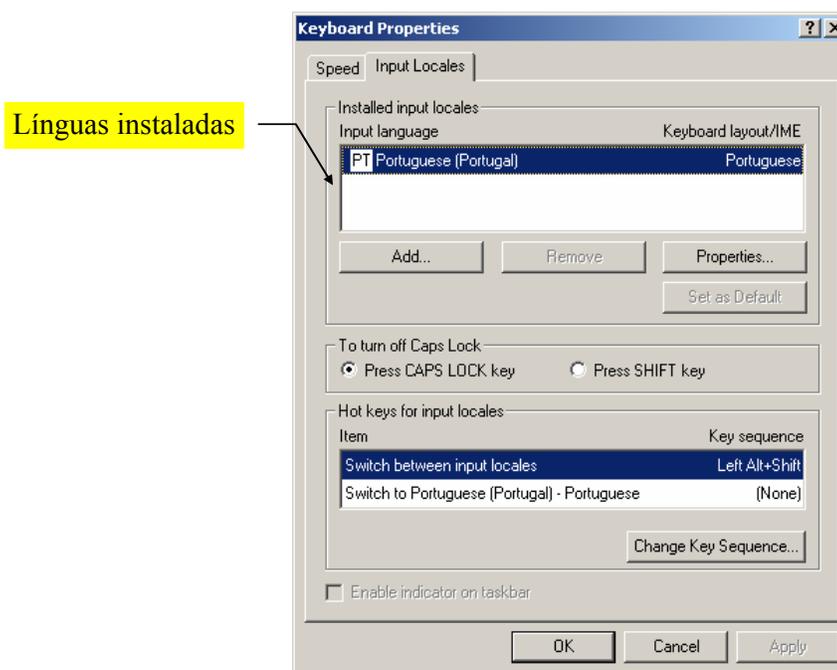


Figura 61: Alteração da língua de um teclado.

Escolha de uma nova língua para o teclado

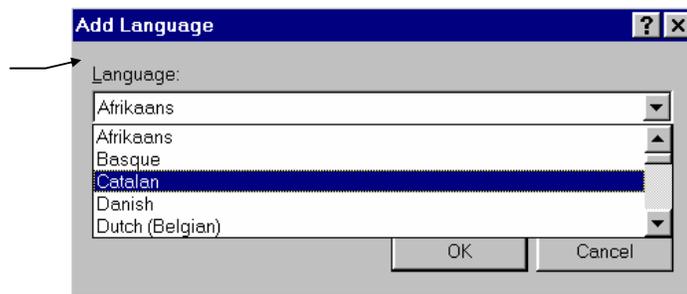


Figura 62: Diálogo para a adição de uma língua.

Para adicionar uma língua, que ainda não se encontre definida no diálogo, preme o botão  do diálogo. Se o fizer surge-lhe no ecrã um segundo diálogo onde poderá realizar a escolha da língua pretendida pelo movimento habitual de uma *Combo-box* (vide figura 62).

Uma vez escolhidas todas as línguas com que tenciona trabalhar seleccione aquela que pretende que seja a língua por defeito no arranque do sistema - habitualmente a portuguesa - e preme o botão .

Para poder alternar rapidamente entre as diversas línguas escolhidas pode optar entre duas combinações possíveis de teclas que poderá indicar ao sistema premindo o *Radio Button* correspondente: LEFT ALT+SHIFT   e CTRL+SHIFT  . Assim, por exemplo se tiver escolhido a opção LEFT ALT+SHIFT, para alternar entre as diversas línguas terá que premir as teclas ALT e SHIFT que se encontram do lado esquerdo do seu teclado em simultâneo.

Coloca-se ainda a questão de saber em cada momento em que língua se encontra configurado o teclado. O *Windows* permite que esta informação seja permanentemente visualizada na barra de tarefas. Para tal deverá activar a *Check-box*   que se encontra no diálogo da figura 61.

**Exercício 15:** Adicione as línguas Inglês (*british*) e Alemão (*standart*) e configure o sistema de modo a que a língua do teclado seja visualizada na barra de tarefas e que a comutação entre línguas se realize pela combinação de teclas CTRL+SHIFT.

## 5. Configuração do Rato

O utilitário *Mouse* permite configurar e personalizar o rato em diversos aspectos: desde a função dos botões, as formas de apontadores, até os movimentos. Para activar o diálogo de controlo do rato execute a partir do *Control Panel* a aplicação *Mouse* . Se o fizer é visualizada a seguinte caixa de diálogo:

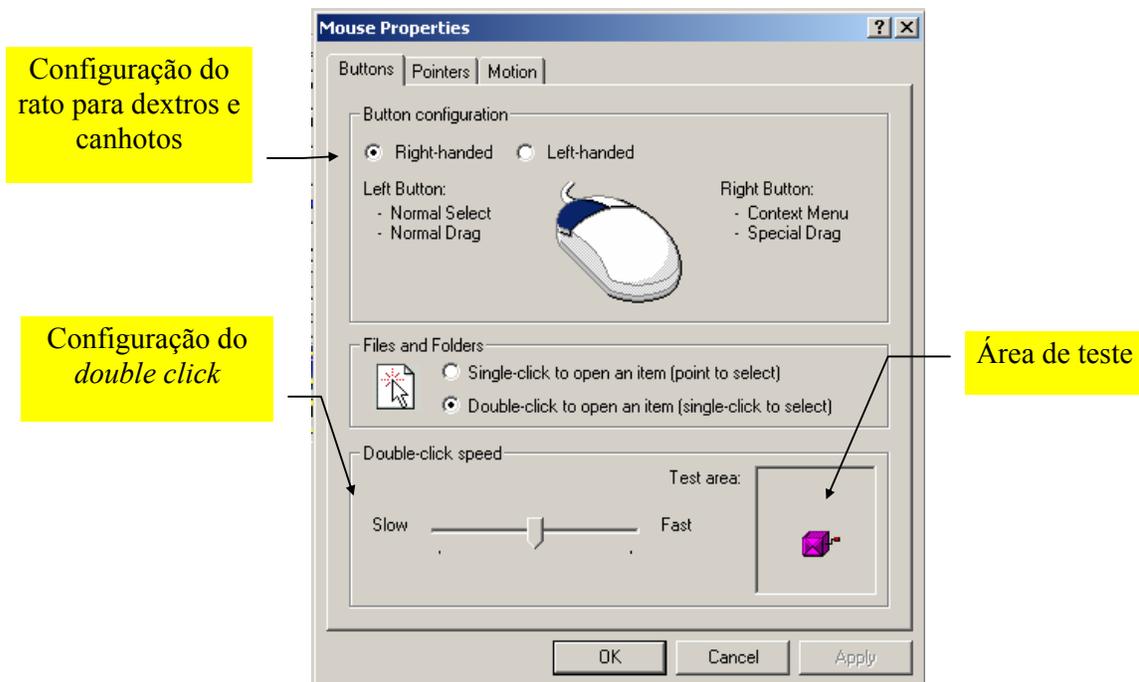


Figura 63: Diálogo para o controlo do rato.

### 5.1 Definir a Função dos Botões do Rato

Por defeito o rato está configurado para utilizadores dextros. Contudo, os utilizadores que usam a mão esquerda poderão adaptar as funções dos botões do rato da forma que mais lhes convém. Para isso basta seleccionar no diálogo da figura 63 o *Radio Button* premindo de seguida o botão .

O intervalo entre cliques sucessivos para realizar a operação de *double click* (clique duplo) do rato também é configurável. Para tal o utilizador dispõe de um *Slider*, com o qual poderá ajustar o intervalo que mais se lhe adequa. Para testar o intervalo programado dispõe de uma área de teste. Se realizar correctamente o *double click* sobre esta área surge-lhe a seguinte imagem:



**Exercício 16:** Coloque o *slider* na posição mais à esquerda e experimente o efeito na área de teste. Coloque agora o *slider* no extremo oposto e verifique o respectivo efeito.

## **5.2 Definição da Forma dos Apontadores**

O apontador do rato adopta diversas formas consoante a aplicação onde se encontra e o tipo de tarefa que se está a executar. O tipo de apontador que surge em cada momento pode, contudo, ser configurado. Para tal, o *Windows 2000* dispõe de uma biblioteca de apontadores de onde poderá escolher aqueles que mais gosta ou lhe convêm. Se escolher o separador **Pointers** surge-lhe no ecrã o seguinte diálogo:

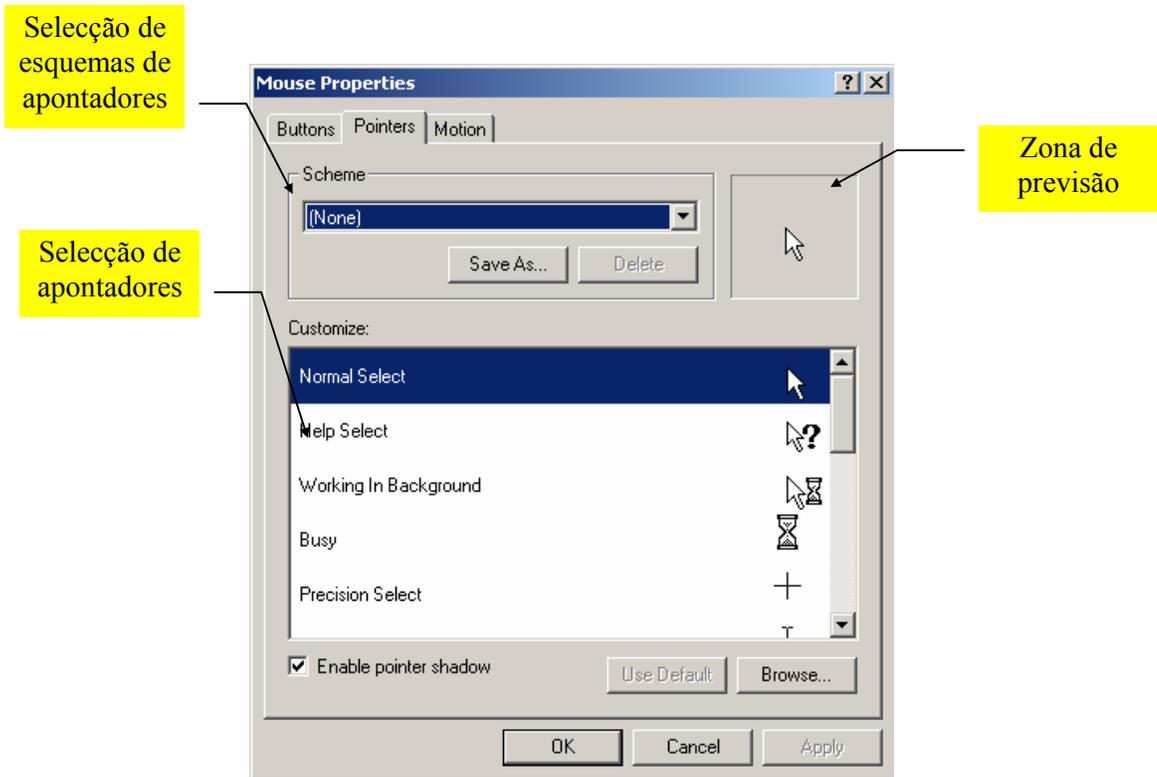


Figura 64: Selecção de apontadores.

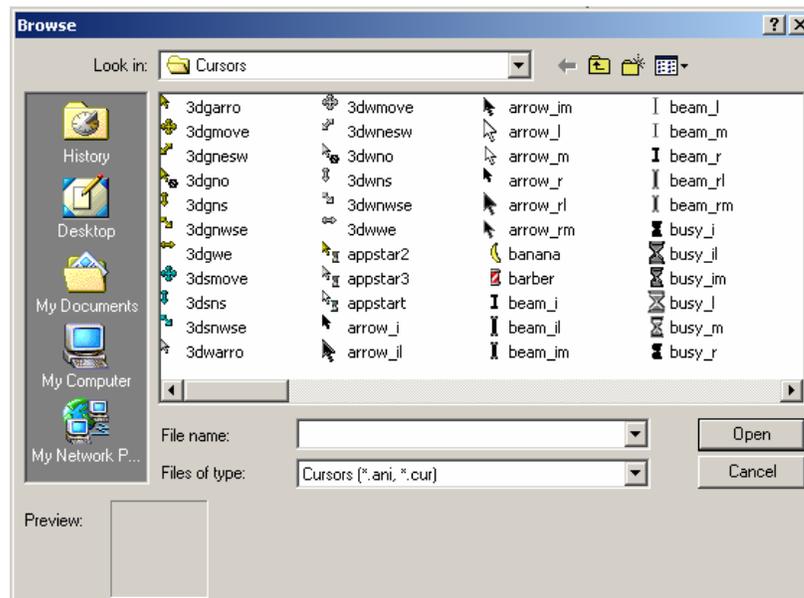


Figura 65: Diálogo auxiliar para a escolha de apontadores.

Para optar por um determinado esquema de apontadores basta abrir a *Combo-Box* existente para o efeito no diálogo e seleccionar o conjunto pretendido. Contudo, se pretender configurar os apontadores para cada tipo de situação, deverá escolher a situação pretendida e premir o botão  , surgindo-lhe, nesse caso, um diálogo auxiliar de onde poderá escolher o tipo de apontador que mais lhe convém (vide figura 65). Para validar a sua escolha deverá premir o botão  do diálogo auxiliar e seguidamente o botão  para que a sua escolha surta o efeito pretendido.

### 5.3 Configuração do Movimento do Rato

Para facilitar a visualização do apontador no ecrã, o *Windows 2000* permite ao operador escolher a velocidade com que este elemento se movimenta bem como o tipo de visualização a usar. Este último item é de extrema importância quando o sistema é formado por um computador portátil com ecrã de baixa qualidade.

Quando o operador escolhe o separador  surge-lhe o seguinte diálogo:

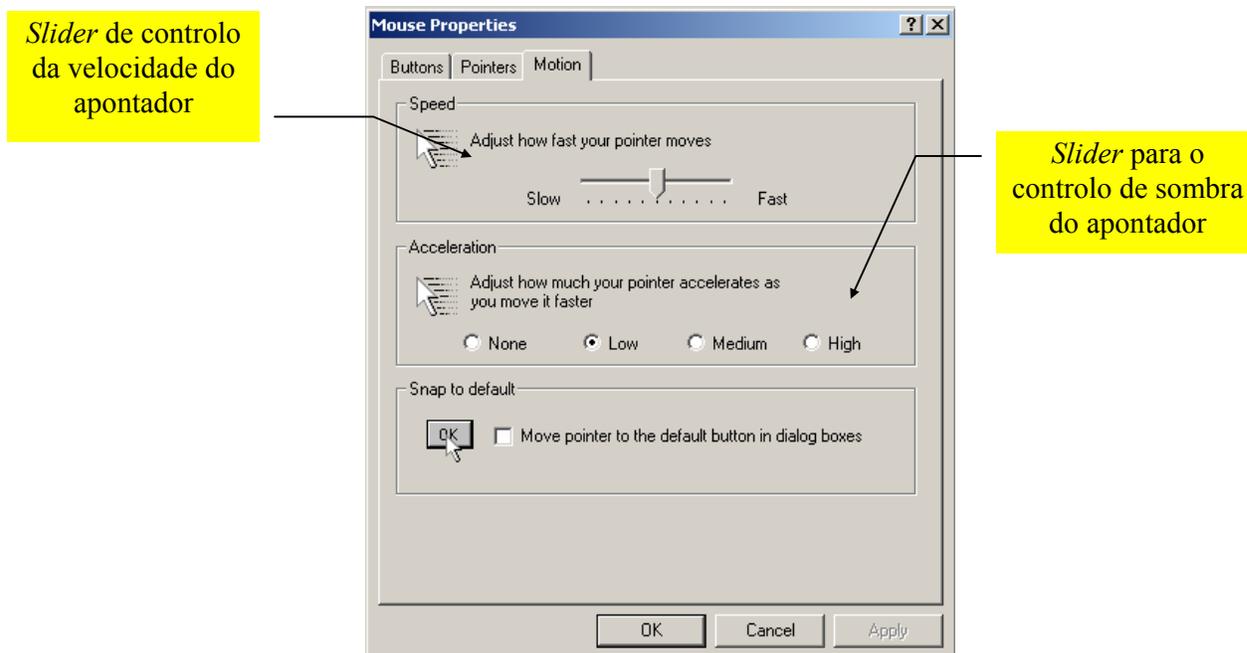


Figura 66: Diálogo para o controlo do movimento do apontador.

A velocidade do apontador é controlados pelo *Sliders* que o diálogo exhibe.

**Exercício 17:** Coloque o *slider Pointer Speed* na posição mais à esquerda e preme o botão *Apply*. Desloque o apontador desde o canto inferior esquerdo até ao canto superior direito. Repita o exercício com o *slider* na posição mais à direita. Qual foi a diferença de curso do rato?

## 6. Configurações Regionais

Como já foi referido, o sistema *Windows 2000* é um sistema multilingue. Atendendo a que existem diversos parâmetros que diferem de país para país, o *Windows 2000* permite ao utilizador a sua configuração, para o efeito disponibilizando o utilitário *Regional Options*. Este utilitário permite, entre outros, configurar parâmetros tais como os formatos de números, valores monetários, datas e horas. Para o executar corra a aplicação  a partir do *Control Panel*. Se realizar o procedimento sugerido, o *Windows* exhibe o seguinte diálogo:

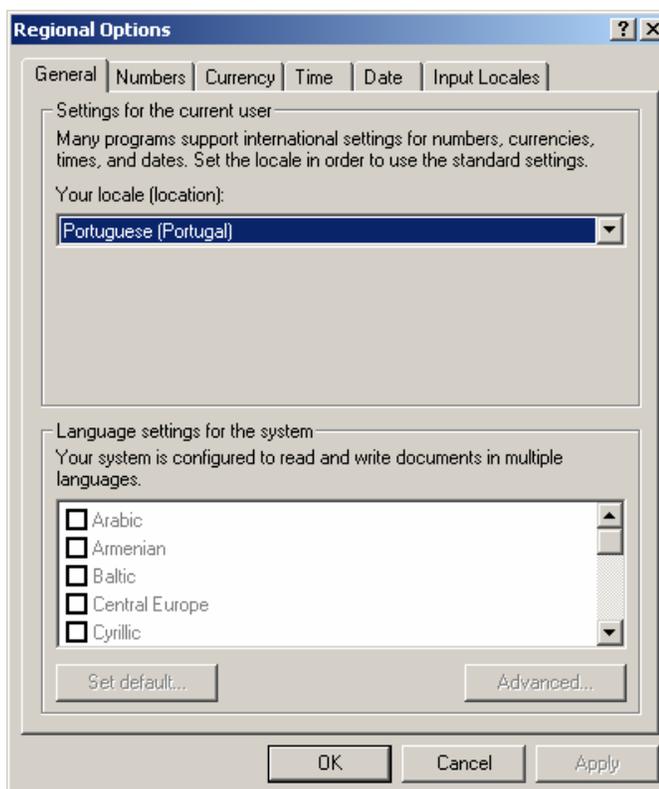


Figura 67: Diálogo para as configurações regionais.

## 6.1 Definição do País

A definição do país condiciona as diversas configurações dos restantes parâmetros que é possível definir nesta aplicação, devendo, por essa razão, ser sempre realizado. Para o efeito o *Windows* disponibiliza uma *Combo-Box* com as diversas alternativas possíveis (vide figura 14). Assim, deverá abrir a referida *Combo-Box*, escolher, da forma habitual, o país pretendido e seguidamente carregar no botão  para que a sua escolha tenha efeito.

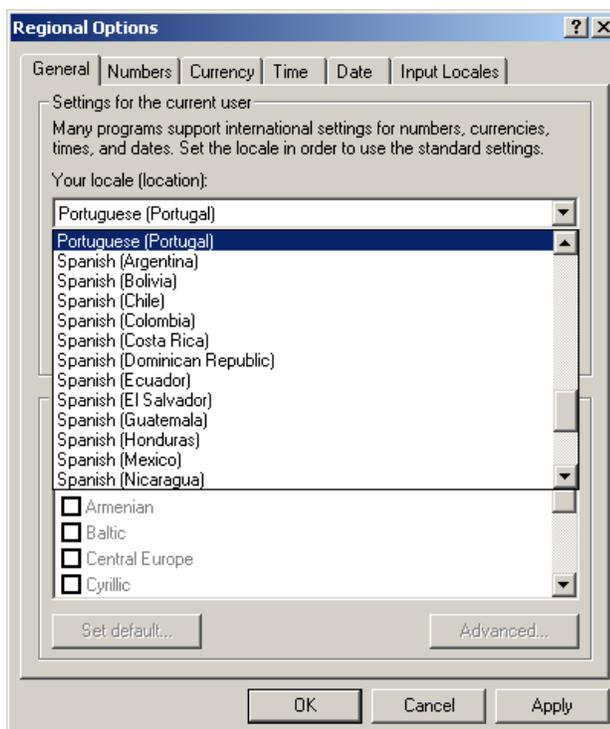


Figura 68: Definição do país.

## 6.2 Configuração do formato dos números

A configuração do formato dos números tem implicações extremamente importantes em aplicações de cálculo como é o caso do *Excel*. Assim, quando uma determinada

aplicação deste tipo não apresentar resultados correctos (assinalando erros), deverá sempre verificar o formato dos números que o sistema está a usar.

Para manipular o formato dos números do sistema active o separador `Number`. Se o fizer surge-lhe o seguinte diálogo:

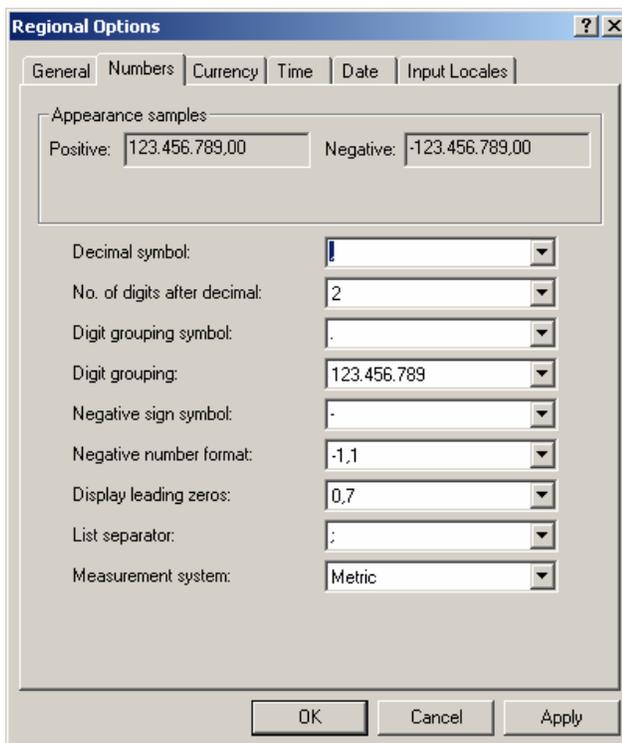


Figura 69: Configuração do formato dos números do sistema.

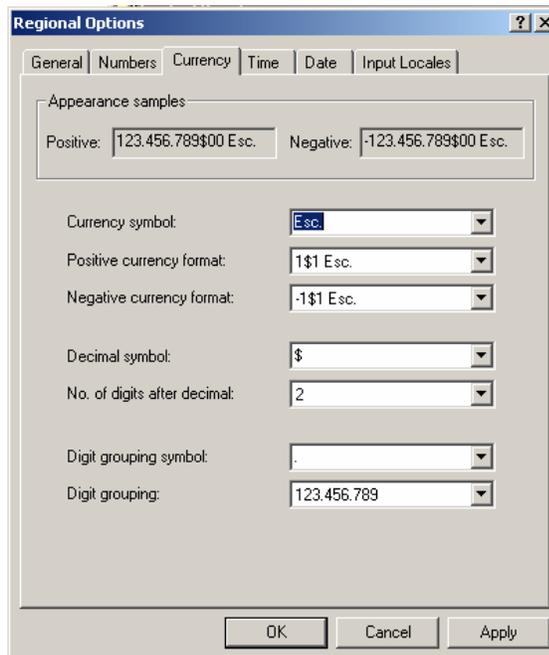
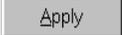


Figura 70: Diálogo para a configuração dos valores monetários.

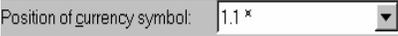
Como pode observar são vários os parâmetros que pode configurar, usando as respectivas *Combo-Boxes*, neste tópico:

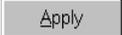
- Decimal symbol: [.] - símbolo decimal,
- No. of digits after decimal: [2] - número de dígitos após o separador decimal,
- Digit grouping symbol: [.] - símbolo de agrupamento de dígitos,
- No. of digits in group: [3] - número de dígitos por grupo,
- Negative sign symbol: [-] - símbolo para valores negativos,
- Negative number format: [-1.1] - formato dos número negativos,
- Display leading zeroes: [0.7] - visualização obrigatória de zero,
- Measurement system: [Metric] - sistema de medição,
- List separator: [;] - separador de listas de valores.

Após a alteração de qualquer um destes parâmetros, deverá premir o botão  para que as suas escolhas produzam o efeito desejado de imediato, e seguidamente o botão  para que assumam um carácter permanente.

### 6.3 Configuração dos valores Monetários

Para alterar os parâmetros que controlam a forma como são apresentados e introduzidos os valores monetários, deverá seleccionar o separador . Se o fizer, surge-lhe no ecrã o diálogo da figura 70. Como é possível observar, os parâmetros que poderá configurar são:

 Currency symbol: Esc	- símbolo monetário,
 Position of currency symbol: 1.1 *	- posicionamento do símbolo monetário,
 Negative number format: -1.1 *	- formato dos valores negativos,
 Decimal symbol: \$	- símbolo decimal,
 No. of digits after decimal: 2	- número de dígitos após o símbolo decimal,
 Digit grouping symbol: .	- símbolo para agrupamento de dígitos,
 Number of digits in group: 3	- número de dígitos por grupo.

Após a alteração de qualquer um destes parâmetros, deverá premir o botão  para que as suas escolhas produzam o efeito desejado de imediato e seguidamente o botão  para que assumam um carácter permanente.

### 6.4 Configuração da Data

Para configurar o formato da data do computador deve seleccionar o separador , surgindo, nesse caso, o diálogo da figura 17.

Neste diálogo poderá configurar os seguintes parâmetros:

 Short date style: dd-MM-yyyy	- formato simples de apresentação da data,
--	--

Date separator: -

- símbolo separador de dia, mês e ano,

Long date style: dddd, d' de 'MMMM' de 'yyyy

- formato por extenso da data do sistema.

Nos campos

Short date sample: 09-08-1997

e

Long date sample: sábado, 9 de agosto de 1997

podem ser observados os efeitos das opções

escolhidas.

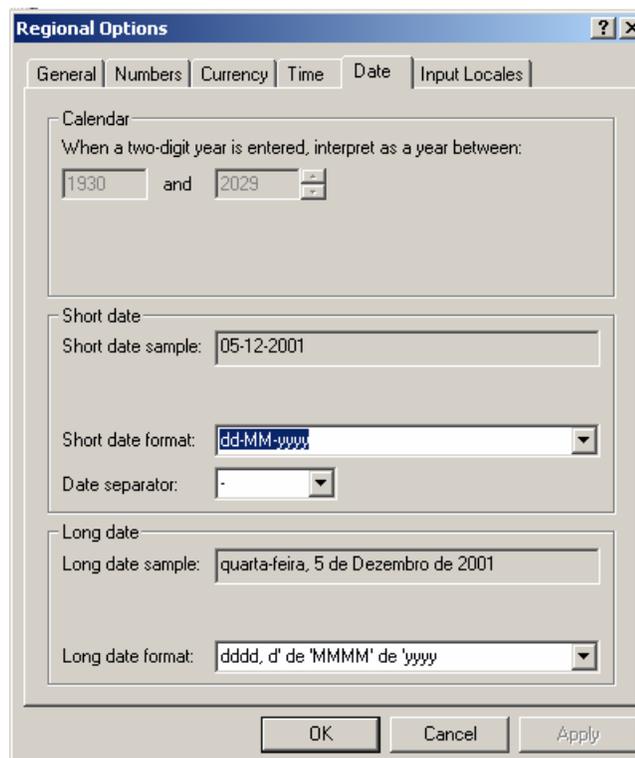


Figura 71: Diálogo para a alteração do formato da data.

## 6.5 Configuração do Formato da Hora

Para configurar o formato da hora do computador deve seleccionar o separador **Time**, surgindo, nesse caso, o diálogo da figura 18.

Neste diálogo poderá configurar os seguintes parâmetros:

Time format: H:mm:ss

- formato da hora do sistema,

Time separator: [v] - separador de horas, minutos e segundos,  
AM symbol: [v] - símbolo de AM,  
PM symbol: [v] - símbolos de PM.

No campo Time sample: [0:16:07] poderá observar o efeito das opções realizadas.

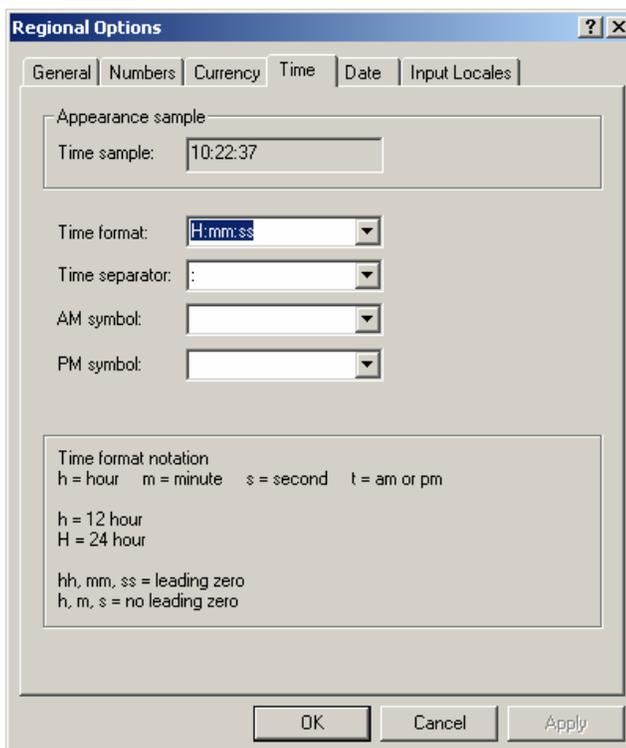


Figura 72: Diálogo para a configuração do formato da hora.

## 7. Configuração de Associação de Sons

O *Windows 2000* permite que um conjunto de sons sejam associados aos diversos eventos do sistema. Assim, poderá optar por um determinado som - se o seu sistema dispuser de equipamento multimédia poderá gravar e adquirir os sons que mais lhe convêm, por exemplo, para a saída do *Windows*.

Para executar a ferramenta de configuração de sons do sistema corra a aplicação



a partir do *Control Panel*, surgindo-lhe, se o fizer, o diálogo da figura 19.

Para escolher um determinado som para um evento, seleccione primeiro o evento pretendido. Seguidamente deverá abrir a *Combo-Box* (vide figura 20 - selecção do som *The MicroSoft Sound* para o evento *Exit Windows*) e escolher o som que deverá ser associado ao evento anteriormente escolhido, podendo ouvir o som seleccionado se premir o botão  do diálogo. Se nenhum dos sons disponíveis for do seu agrado para o evento em causa, poderá indicar ao sistema o nome de um ficheiro alternativo de sons (os tais que poderá ter gravado com o seu equipamento multimédia). Para tal preme o botão  que conduz à abertura de um diálogo auxiliar, com o qual poderá identificar o ficheiro pretendido e a respectiva localização (vide figura 21).

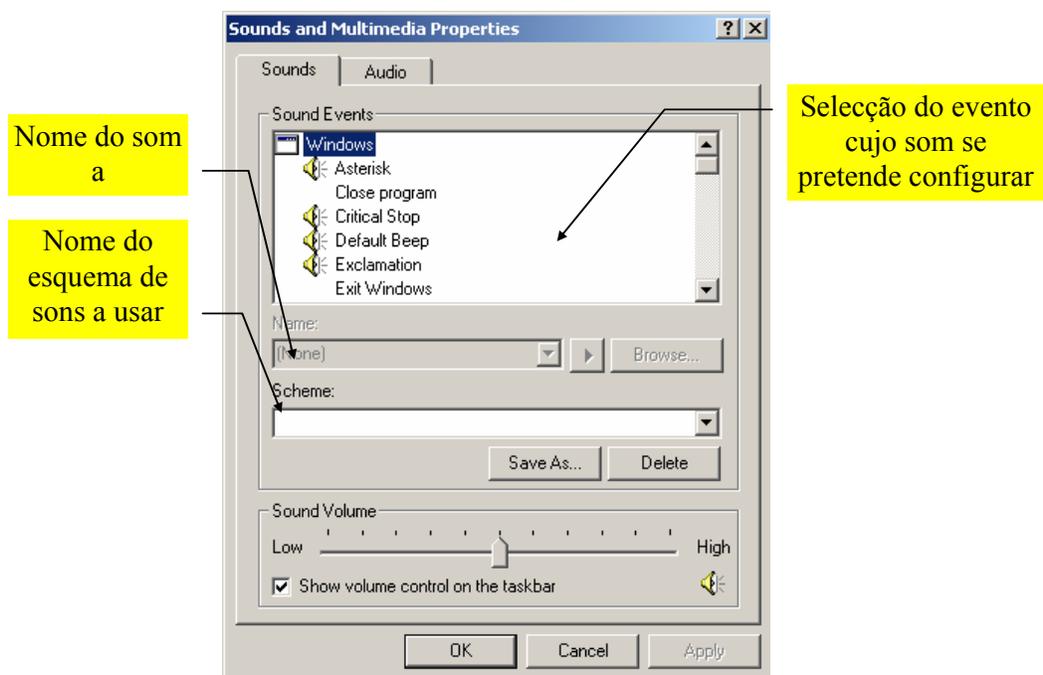


Figura 73: Diálogo para a configuração dos sons.

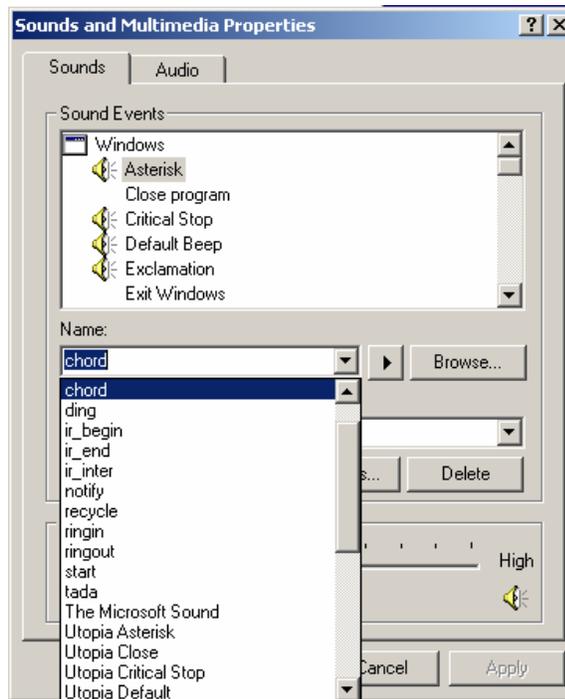


Figura 74: Exemplo de configuração de sons.

Uma vez configuradas as diversas associações de sons poderá gravar o esquema definido. Para tal preme o botão **Save As...** do diálogo de configuração, surgindo no ecrã a caixa de diálogo auxiliar da figura 22, que lhe permite indicar o nome que pretende atribuir ao esquema definido.

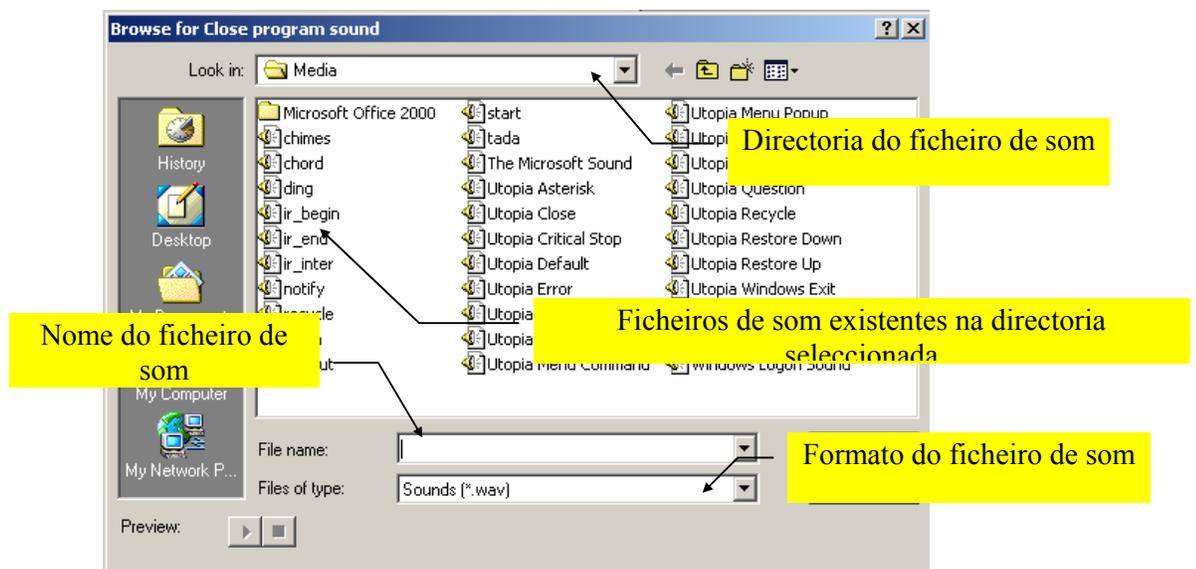


Figura 75: Diálogo para a escolha de um ficheiro de som.

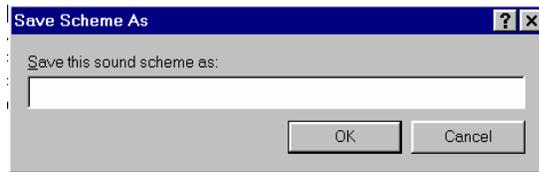


Figura 76: Diálogo para a gravação de um esquema de sons.

Após a alteração de qualquer um dos parâmetros referidos, deverá premir o botão  para que as suas escolhas produzam o efeito desejado e seguidamente o botão  para que sejam assumidas permanentes pelo sistema.

## 8. Gestão de Impressoras

A gestão das impressoras é uma tarefa fundamental, já que, na maioria dos programas, muitos dos resultados são condicionados pelo tipo de impressora seleccionada. Por exemplo, se escrever um texto num processador de texto como o *Word*, o aspecto final do documento sofre sempre ligeiras alterações quando se muda de impressora.

No *Windows 2000*, o utilitário *Printers*, acessível a partir do *Control Panel*  ou a partir do menu *Start*, submenu *Settings*  (vide figura 23), controla a gestão e configuração das impressoras.

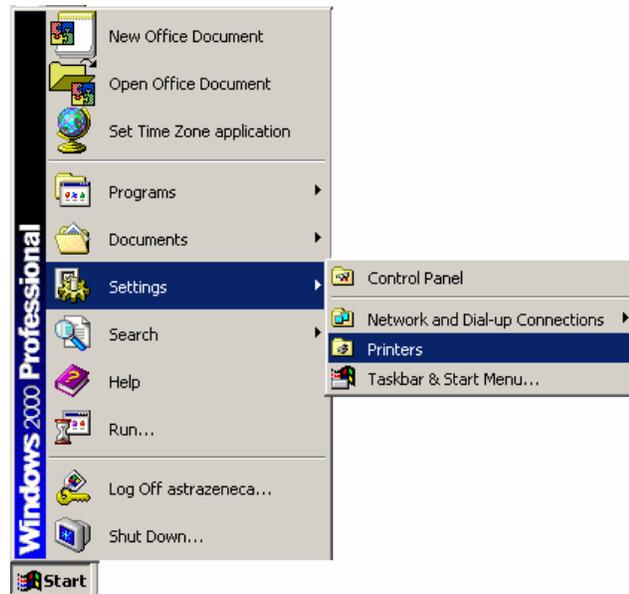


Figura 77: Acesso ao utilitário *Printers*.

Ao executar a ferramenta *Printers*, o *Windows* abre uma janela (vide figura 24) que dispõe sempre de pelo menos um ícone - o ícone *Add Printer*. Podem ainda encontrar-se nessa janela um ícone por cada impressora instalada no sistema.

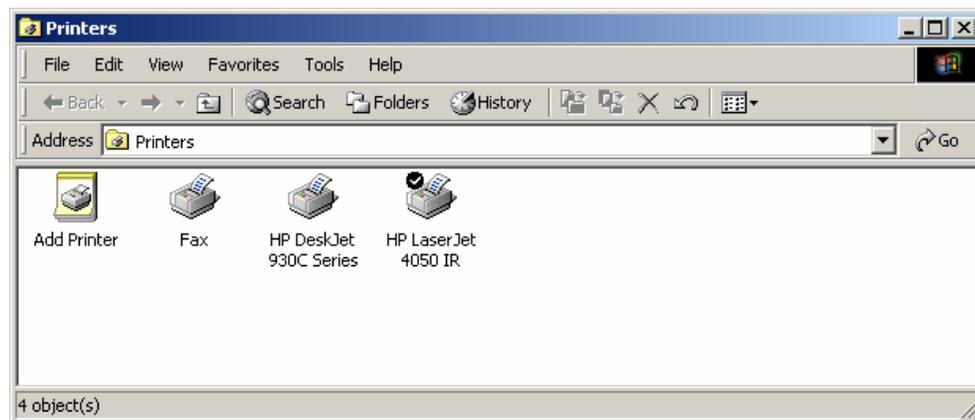


Figura 78: A janela *Printers*.

### 8.1 Instalação de uma Nova Impressora

Para instalar uma impressora no sistema *Windows* 2000 deve clicar duas vezes (movimento semelhante ao da execução de um programa) no ícone *Add Printer* .

O movimento descrito conduz à execução do *Wizard* de instalação de impressoras (vide figura 25).

Para iniciar a instalação deve premir o botão  da janela, tomando o *Wizard* o aspecto da figura 26. Nesta fase é-lhe colocada a questão se a impressora é local, isto é, se se encontra ligada directamente ao seu computador ou se se trata de uma impressora instalada em rede. Após ter respondido à pergunta colocada - a sua resposta deverá ser introduzida premindo o *Radio Button* correspondente, deve accionar novamente o botão . Como reparará certamente, o botão  encontra-se nesta fase activo. Assim, poderá em qualquer momento recuar um ou mais passos no processo de instalação para realizar eventuais correcções nas suas opções.



Figura 79: O *Wizard* de instalação de impressoras.

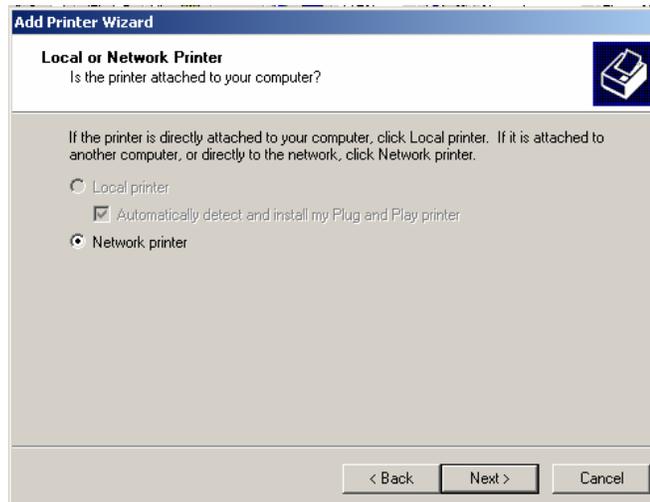
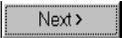


Figura 80: Escolha da localização da impressora.

Assumindo que escolheu a opção de impressora local e que premiu o botão , no próximo passo terá que indicar ao sistema o porto onde irá ligar o equipamento. Regra geral o porto escolhido deverá ser a *LTP1*. Contudo, se este porto já se encontrar ocupado poderá optar (se a impressora o permitir) por um porto de comunicação diferente. Esta informação deverá ser introduzida usando a *List-Box* (vide figura 28) que é visualizada para o efeito.

De seguida, o *Wizard* exhibe agora uma janela com duas *List-Boxes*: a do lado esquerdo inúmera um conjunto de fabricantes e a do lado direito apresenta um conjunto de modelos de impressoras de cada fabricante. Assim, deverá seleccionar primeiro o fabricante da impressora que pretende instalar, de modo a visualizar todos os modelos conhecidos pelo sistema desse fabricante. Se a impressora pretendida não constar da lista, deverá premir o botão  e inserir a primeira disquete que acompanha a sua impressora.

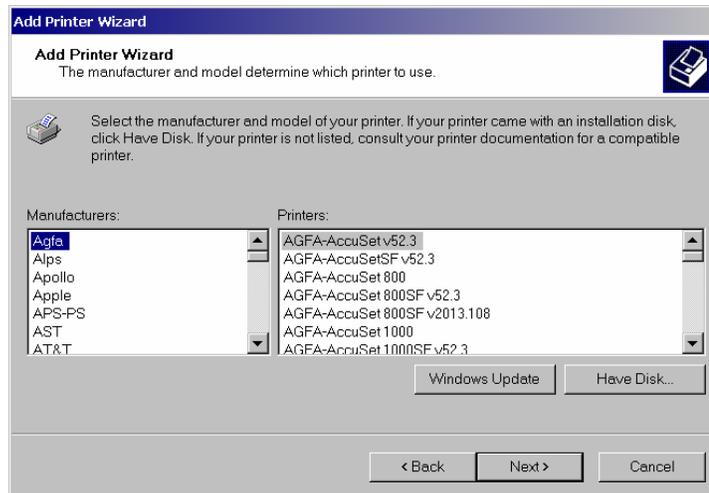


Figura 81: Identificação da impressora a instalar.

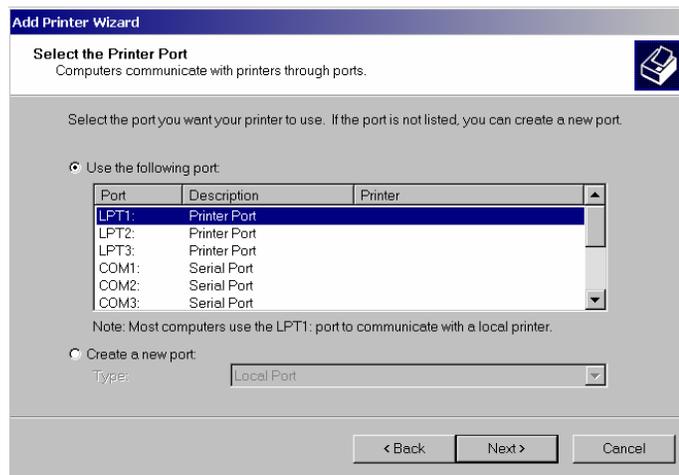


Figura 82: Escolha do porto para a ligação do equipamento.

Uma vez descrita a impressora deve, mais uma vez, premir o botão , passando, deste modo, ao próximo passo da instalação.

O próximo passo do processo de instalação consiste em indicar ao *Windows* o nome da nova impressora, e se esta deverá ser a impressora por defeito do sistema, isto é, aquela que todos os programas deverão usar se não forem reconfigurados. Esta informação é fornecida ao programa de instalação, preenchendo a caixa de texto e premindo o *Radio Button* adequado (vide figura 83).

Uma vez premido o botão  o *Wizard* pergunta ainda se o utilizador pretende partilhar a impressora e se quer imprimir uma página de teste no final da instalação

(vide figura 84). Esta facilidade permite, nomeadamente, testar se todas as opções realizadas ao longo do processo de instalação foram as correctas.

Para finalizar o processo de instalação deverá premir nesta fase o botão . O *Windows* inicia então a transferência dos ficheiros necessários para o sistema operativo, devendo o utilizador fornecer as disquetes correctas à medida que estas vão sendo solicitadas pelo programa.

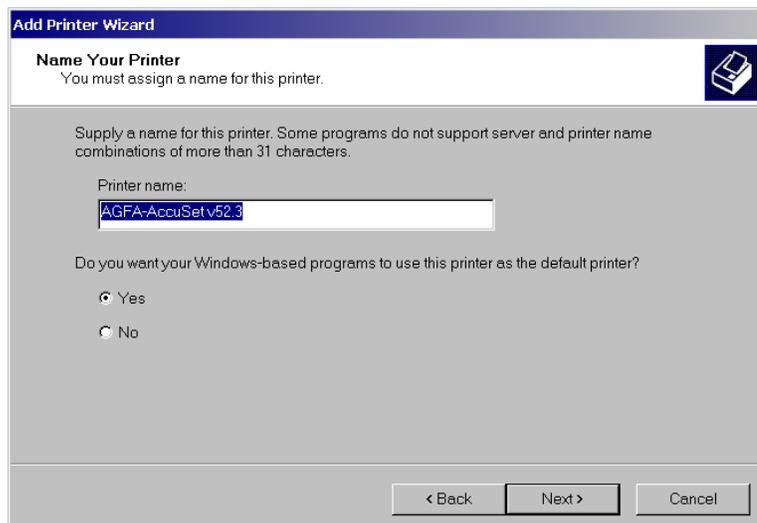


Figura 83: Escolha do nome da impressora por defeito.

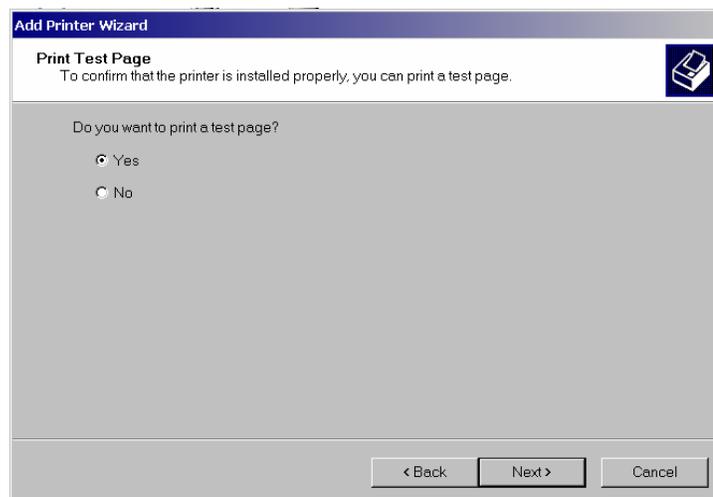


Figura 84: Escolha da página de teste.

**Exercício 18:** Realize a instalação da impressora *Apple LaserWriter Plus*.

## 8.2 Instalação de Impressoras em Rede

Quando no diálogo anterior, o utilizador opta por uma impressora localizada em rede, o *Windows 2000* visualiza o diálogo da figura 31, para que a localização exacta da impressora seja especificada. Essa localização, composta pelo nome atribuído à impressora e pelo nome do computador a que esta se encontra fisicamente ligada, pode ser introduzida directamente no campo , ou o operador poderá optar por uma escolha interactiva, sendo esta última a forma mais simples para identificar a impressora pretendida.

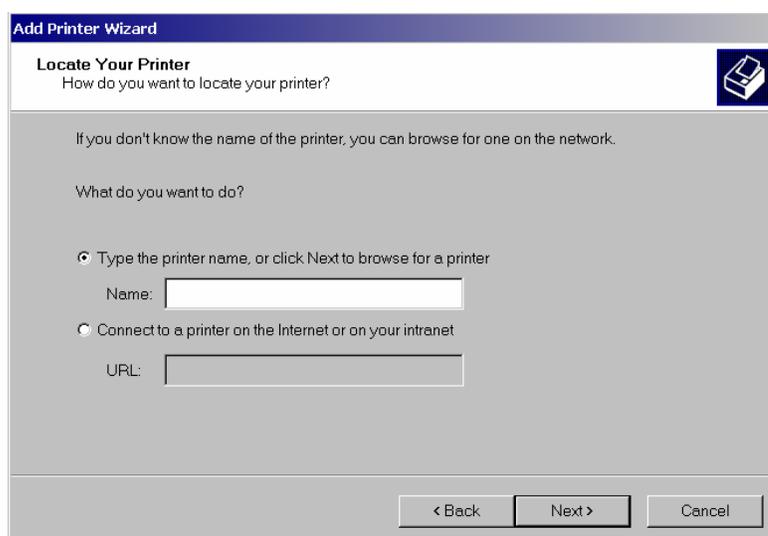


Figura 85– Localização da impressora

## 9. Instalação de *Software* Novo

Para adicionar um novo programa no sistema:

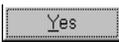
1. Prema o botão  do utilitário de instalação.
2. Prema o botão  do *Wizard* que surge no ecrã.
3. Siga todos os passos sugeridos pelo *Wizard*.

## 9.1 Remoção/Alteração de um Programa Instalado



Para adicionar o software desejado seleccione a opção **Add New Programs**. Posteriormente deverá indicar a localização dos ficheiros de instalação.

Para remover ou alterar as componentes de um programa instalado no sistema:

1. Seleccione a aplicação na *List-Box* da janela do utilitário de instalação de *Change/Remove Programs* (vide figura 32).
2. Prema o botão .
3. Confirme a sua opção premindo o botão  no diálogo que surge no ecrã.
4. Siga as instruções do *Wizard* (em princípio será iniciada nesta fase a remoção dos ficheiros).

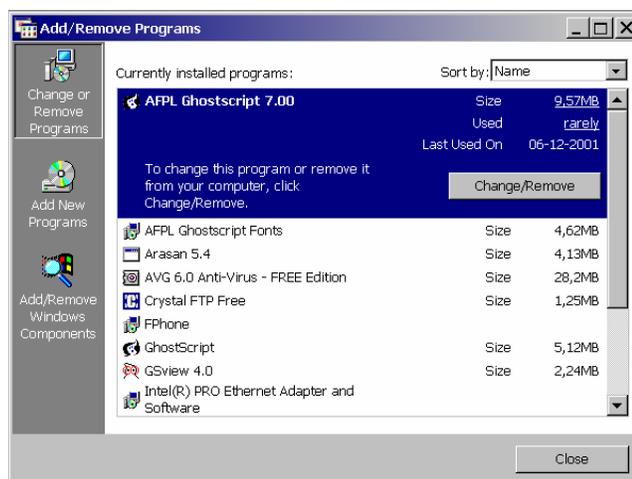


Figura 32: Janela da aplicação *Add/Remove Software*.

**Exercício 19:** Instale o *Software Netscape* que se encontra na drive **c:**, directoria **temp** do seu computador.