

Destacável mensal sobre microcomputadores
n.º 20 Outubro 1985
Coordenação de Fernando Antunes

se7e

Os "mistérios" dos ficheiros

Voltamos hoje ao tema sobre ficheiros, abordando mais alguns conceitos fundamentais.

Organização sequencial e aleatória

Voltamos à nossa loja. À medida que foram feitas as transacções, foram ficando registadas num meio magnético, fisicamente umas a seguir às outras; a 8.ª a seguir à 7.ª, que se seguiu à 6.ª e assim sucessivamente; ou seja, estão gravadas sequencialmente pela ordem em que foram entrando. Dizemos então que o ficheiro de movimentos tem uma organização **SEQUENCIAL**.

Mas o ficheiro-mestre de stocks não está assim. Quando foi criado, o computador determinou a posição de cada registo com base no código do artigo, que actua como uma chave de acesso na pesquisa do registo pretendido. Este é um exemplo de um ficheiro com organização **INDEXADA**. Existe ainda um terceiro modo de organização, digamos uma solução intermédia entre a sequencial e a indexada: Dá-se-lhe o nome de organização **RELATIVA**, pois que os registos se encontram dispostos numa forma ordenada (pele índice) e numa posição relativa entre si, ou seja, o registo que tem por índice o número 5 é o 5.º registo do ficheiro. A organização relativa e indexada é também comum chamar-se organização **ALEATÓRIA**.

Acesso sequencial e directo

Tratemos agora do modo de acesso. Enquanto que um ficheiro sequencial só pode ser acedido de um modo sequencial (é lido o registo imediatamente a seguir ao que se leu), um ficheiro indexado ou relativo pode ser acedido quer sequencial quer directamente. No modo de acesso directo, é colocado na variável-chave, o valor do índice do registo pretendido e o sistema operativo acede **DIRECTAMENTE** ao registo. A principal vantagem dos ficheiros sequenciais, é sem dúvida a

ocupação mínima do suporte magnético, já que ocupam unicamente o espaço necessário à sua dimensão, enquanto que um ficheiro aleatório em consequência da sua função de atribuição, deixa espaços vazios entre os registos, ocupando portanto um comprimento maior.

Tempos de acesso

A consulta de um registo num ficheiro sequencial pode chegar a ser muito lenta, já que dependerá de o registo pretendido estar no

princípio ou no fim do suporte; o acesso directo far-se-á em alguns décimos de segundo. O que não quer dizer que seja sempre preferível utilizar este acesso, pois que se tivermos de ler todos os registos do ficheiro (listagem), ou um grande número deles, pode resultar mais rentável o acesso sequencial, já que o tempo de acesso ao seguinte é muito menor que o tempo médio de acesso directo.

A existência de um ficheiro começa quando é criado o primeiro registo. Durante a sua vida, poderá suportar diversas operações. **ABERTURA** — Introduzir novos registos.

ALTERAÇÃO — Alterar o conteúdo dos campos.

ANULAÇÃO — Eliminar registos.

CONSULTA — Apresentação no ecrã ou na impressora do conteúdo do registo.

Veremos posteriormente estas opções mais detalhadamente quando construirmos o programa base.

Depois dessa introdução necessariamente breve e generalizada sobre ficheiros, vamos abordar o tema que os nossos leitores ansiosamente aguardam:

Ficheiros no Spectrum

Em face do seu sistema operativo, quer usemos 'disketes' ou não só poderemos usar dois modos de organização: a sequencial e a relativa. E também dois modos de acesso: O sequencial e o directo. Se no nosso programa usarmos uma colecção «DATA», esse conjunto de informações constitui um ficheiro de organização sequencial, já que os dados se encontram dispostos numa ordem aleatória, a seguir uns aos outros. O acesso é também sequencial, pois que teremos de ler **TODOS** os elementos da colecção até encontrarmos o que nos interessa. Alternativamente, se colocarmos os dados numa «DIM» de dimensão 'n', então poderemos considerar que dispomos de um ficheiro de organização relativa, pois como sabemos dispomos de 'n' registos cujo índice de pesquisa varia de 1 a 'n', ordenadamente.

O acesso a este ficheiro pode ser feito sequencialmente, manipulando o índice através de um ciclo «FOR-NEXT» e comparando o conteúdo de cada variável indexada com o valor que nos interessa. Ou de uma forma directa acedendo ao registo pretendido, atribuindo à variável-índice o valor conveniente. Para ilustrar o que foi dito e ajudar à compreensão do tema, iremos no próximo artigo estudar alguns exemplos que tornarão com certeza mais claros os conceitos descritos, talvez um pouco difíceis mas indispensáveis na 'Arte de Bem Programar'.

Hélder Guerra



