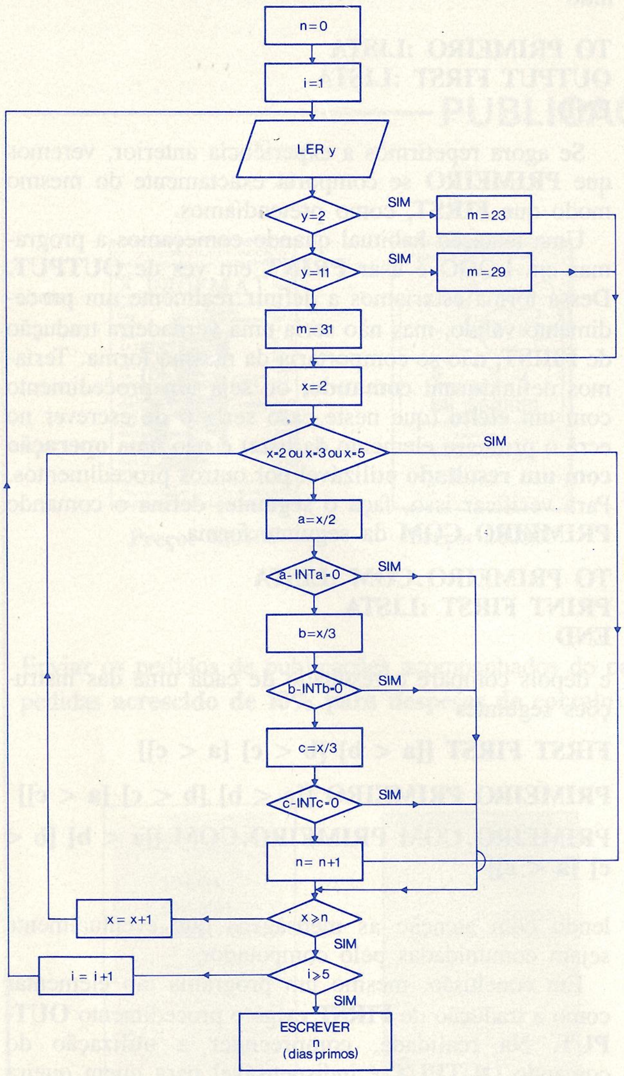
Como o computador resolve o problema n.º 19 do mês de Junho

Não era interessante, sob o ponto de vista da Matemática, fazer uma contagem de «dias primos». Era um trabalho mecânico e, provavelmente, quem não é máquina não desempenharia esta tarefa com êxito: sujeitava-se a cansaço, distracções e consequentes erros de contagem.

Entregou-se então este trabalho a quem nunca se distrai e se subordina sempre a sequências correctas e repetitivas de instruções: o computador. O homem é talhado para outro tipo de coisas — o trabalho criativo.



NOTA: y — MESES PRIMOS
m — número máximo de dias que interessam em cada mês
n — número de dias primos

Talvez o programa apresentado não seja o melhor, mas é aquele que me libertou daquilo que não gosto de fazer.

Ao tentar elaborar o programa verifiquei que todos os números, de 2 a 31, ou são primos ou divisíveis pelos primos 2, 3 ou 5. Isso é traduzido pelo segundo ciclo que aparece (o que está no interior de outro) cujo contador de «dias primos» é n = n + 1.

O primeiro ciclo serve para definir a quantidade de dias que interessam a cada «mês primo».

```
5   LET n = 0
10  FOR i = 1 TO 5
20  READ y
30  IF y = 2 THEN LET m = 23 : GOTO 60
40  IF y = 11 THEN LET m = 29 : GO TO 60
50  LET m = 31
60  FOR x = 2 TO m
70  IF x = 2 OR x = 3 OR x = 5 THEN GO TO 140
80  LET a = x/2
90  IF a - INT a = 0 THEN GO TO 150
100 LET b = x/3
110 IF b - INT b = 0 THEN GO TO 150
120 LET c = x/3
130 IF c - INTc = 0 THEN GO TO 150
140 LET n = n + 1
150 NEXT x
160 NEXT i
170 PRINT "dias primos ="; n
180 DATA 2, 3, 5, 7, 11
```

Arsénio Coelho

Nota da redacção:

O programa que aqui se apresenta não corresponde inteiramente à versão do colega Arsénio Coelho. De facto, eliminámos um ciclo FOR...NEXT para a variável i. Pensamos, no entanto, não ter defraudado a estrutura global. Esta versão corre no Spectrum, o que não acontecia com a original.

Em busca da perfeição...

No último número de Educação e Matemática, na popular secção DIA-A-DIA COM A MATEMÁTICA — quase tão popular como o LOGO.MAT... — saiu o seguinte problema:

Eu sou o primeiro número perfeito porque a soma dos meus divisores, sem contar comigo, é igual a mim. Quem sou eu?

Concluir que é o 6 = 3 + 2 + 1 não foi difícil. Mas fiquei intrigado, pois percebi que tinha tido sorte em o primeiro número perfeito ser tão pequeno... E as perguntas começaram a surgir: haverá mais? muitos ou poucos? alguma lei? terão que ser pares? Resolvi investigar.

Comecei a fazer tentativas, para ver se descobria alguma lei. Mas, entre os dois extremos — de um lado os primos, «imperfeitos» por terem só o 1 como divisor útil, para o efeito, do outro lado os números «imperfeitos» por terem muitos divisores (por exemplo 24, 48 ou 72) — tudo parecia poder acontecer, e quanto a números perfeitos... apenas encontrei, «perfeitamente» por acaso, o 28!

Resolvi recorrer ao LOGO. E como há quem diga que na aprendizagem de uma linguagem de programação é quase tão importante o esforço para ler e compreender programas já feitos como para fazê-los (e eu concordo...) aí estão os quatro procedimentos necessários:

```
TO DIVISORES :NUM
OUTPUT DIVISORES1 :NUM 1
END

TO DIVISORES1 :NUM :POSS.DIV
IF :NUM = :POSS.DIV [OUTPUT []]
IF 0 = REMAINDER :NUM :POSS.DIV [OUTPUT
FPUT :POSS.DIV DIVISORES1 :NUM :POSS.DIV
+ 1]
OUTPUT DIVISORES1 :NUM :POSS.DIV + 1
END

TO SOMA :LISTA
IF EMPTYP :LISTA [OUTPUT 0]
OUTPUT (FIRST :LISTA) + SOMA BF :LISTA
END

TO PERFEITOS :LIMITE :NUM
IF :NUM = :LIMITE [OUTPUT [ ] ]
IF :NUM = :SOMA DIVISORES :NUM [OUTPUT
FPUT :NUM PERFEITOS :LIMITE :NUM + 1]
OUTPUT PERFEITOS :LIMITE :NUM + 1
END
```

Notas:

1. DIVISORES é um procedimento que nos fornece, para cada número :NUM, a lista dos seus divisores, sem o incluir. Mas quem faz todo o trabalho é o subprocedimento DIVISORES1. Note-se que DIVISORES1 tem dois inputs, :NUM e :POSS.DIV (possível divisor), e o que faz é, recursivamente, ir experimentando cada :POSS.DIV e tomando decisões: se é mesmo um divisor (resto — REMAINDER — igual a zero), junta-o (FPUT) à lista e experimenta o seguinte (ou seja, executa o mesmo procedimento DIVISORES1 mas agora com os inputs :NUM e :POSS.DIV + 1), se não é passa logo ao seguinte. Note-se ainda que DIVISORES chama o subprocedimento DIVISORES1 com os inputs :NUM e 1, o que quer dizer que DIVISORES1 vai experimentar todos os possíveis divisores desde 1 até ao próprio :NUM.

2. SOMA é um procedimento geral que nos dá a soma dos elementos de uma :LISTA. É também uma recursão que, naturalmente, vai somando o primeiro elemento da :LISTA (FIRST :LISTA) com a soma dos restantes (SOMA BF :LISTA), e assim sucessivamente até a lista estar vazia (EMPTYP :LISTA), caso em que não soma nada, está claro (OUTPUT 0)...

3. PERFEITOS é um procedimento que se vai servir dos anteriores para nos dar o que pretendemos. Ele vai sucessivamente experimentar :NUM, :NUM + 1, :NUM + 2, ..., até ao :LIMITE. De cada vez que :NUM é perfeito (:NUM = SOMA DIVISORES :NUM) junta-o à lista dos perfeitos e passa ao seguinte, :NUM + 1. Quando não é, passa directamente ao seguinte. No fim, dá-nos a lista dos perfeitos entre :NUM e :LIMITE.

Nota à margem: se tem dificuldades em compreender o funcionamento recursivo destes procedimentos, não se admire: acontece isso com todos nós que somos normais. Mas não será a altura de dedicar algum tempo a este tipo de procedimentos tão poderosos? E saiba já: o melhor processo para começar é admitir como «fezada» que eles funcionam, experimentar construir algum e ver que, **realmente**, assim é! Porque não tenta, por exemplo, escrever o procedimento PRODUTO que calcula o produto dos elementos de uma :LISTA, seguindo passo a passo — com as convenientes modificações — o procedimento SOMA? E o que poderia ser o DOBRO de uma lista de números?

Depois de ter introduzido no computador estes quatro procedimentos, teclei **PRINT PERFEITOS 100 1** e o computador respondeu **6 28!** Ou seja, entre 1 e 100 só há estes dois! Depois experimentei entre 10 e 200 e o computador deu-me uma lista vazia — não há nenhum! O mesmo entre 200 e 300 e entre 300 e 400 (mas entre 300 e 400 já demora mais de dez minutos a dar a resposta — cada vez mais divisores a investigar e somas a fazer...). Entre 400 e 500, o resultado foi 496: mais um número perfeito! Mas nesta altura, isto já chegava para ver que a perfeição, também nos números, é rara... E toda a prudência aconselhava a não continuar por esta via... Foi então que no livro de 1964, «Recreations in the Theory of Numbers» de Albert H. Beiler, aprendi o seguinte:

- até 1964 apenas tinham sido descobertos 23 números perfeitos; a 6, 28 e 496 seguem-se 8128, 33550336, 8589869056, 137438691328,...

- Euclides deduziu uma fórmula para gerar números perfeitos pares:

$$(2^{n-1})\ (2^n-1),$$

em que **n** é um inteiro maior do que 1 que torne o segundo factor primo. Euler demonstrou, 2000 anos depois, que esta formula dá todos os perfeitos pares.

- a dificuldade em utilizar esta fórmula reside em saber determinar os valores de **n** para os quais o segundo factor é primo, pois eles não parecem obedecer a qualquer lei. Em 1953 um computador levou 13.5 minutos a provar que $2^{1279}-1$ é primo e que portanto $2^{1278}(2^{1279}-1)$ é perfeito; trata-se de um número com 770 dígitos...

- não se conhece nenhum perfeito ímpar, mas ninguém provou que não existam.

Desconheço o que se avançou neste campo desde 1964. Alguém sabe e quer dizer?

Eduardo Veloso

NOTA DA REDAÇÃO: Ainda em 1963, Donald Gillier provou que $2^{11213}-1$ é um número primo; trata-se dum número com 3376 dígitos. Até 1984 desconhecemos que descobertas se fizeram neste campo, mas em Abril deste ano novo número primo foi descoberto: $2^{132049}-1$ com 39751 dígitos. A última descoberta de que temos notícia data de Setembro de 1986. Provou-se, então, que $2^{216091}-1$ é, igualmente, um número primo, este com 65050 dígitos.

Agora é só utilizar a formula de Euclides para determinar os perfeitos correspondentes.

Permutas com Associações Estrangeiras

Conforme dissemos no último número, a APM procura desenvolver contactos com Associações de Professores de Matemática de outros países. Interessa-nos, em particular, estabelecer acordos de permuta de publicações. Neste momento, esses acordos existem já com:

- a Sociedade Andaluza de Professores de Matemática (SAPM) — publica a Revista **Thales** (três vezes por ano) que começámos a receber desde o n.º 6 (o primeiro de 1987);

- a Associação de Professores de Matemática de França (APMEP) — publica de dois em dois meses o **Bulletin de l'APMEP** que começámos a receber desde o n.º 357;

- o Grupo de Estudos e Pesquisas em Educação Matemática (GEPEM) do Rio de Janeiro — publica o **Boletim GEBEM**, de que recebemos os números 17 (1985), 18 e 19 (1986), 1.º e 2.º semestres respectivamente).

Iremos agora desenvolver esforços no sentido de estabelecer acordos com outras Associações, designadamente de outras regiões de Espanha, da Bélgica, da Itália, da Grã-Bretanha, dos Estados Unidos e da América Latina, e ainda com grupos de professores dos países de expressão portuguesa. Nalguns casos, existem já alguns contactos. Manteremos uma informação actualizada sobre os resultados desses esforços.

Estas publicações regulares, bem como outras que entretanto nos têm sido oferecidas, estarão à disposição dos sócios da APM logo que possamos dispor de uma sede. De qualquer modo, tencionamos informar todos os núcleos distritais da nossa Associação e os membros da Direcção Nacional dos vários pontos dos país (em casos especiais poderemos mesmo fazê-lo nas páginas de **Educação e Matemática**) sobre o conteúdo das Revistas que formos recebendo para que qualquer sócio possa pedir fotocópias de artigos em que esteja particularmente interessado.