

Você sabe: o seu TK-2000 COLOR é rápido e poderosíssimo, mas é preciso um software para fazê-lo trabalhar. E ele mostrará um desempenho cada vez melhor se você souber programá-lo para fazer exatamente o que você espera que ele faça...

CONHECENDO E UTILIZANDO O TK-2000 é um livro prático, com 26 programas completos, sérios, comentados, permitindo que você, com sua criatividade, os modifique ou faça deles modelos para outros programas.

No CONHECENDO E UTILIZANDO O TK-2000, você encontrará programas para:

- resolver problemas de matemática e física
- produzir sons que podem se tornar melodias
- desenhar figuras reais e abstratas, inclusive gráficos animados
- criar desenhos de duas e três dimensões, usando os modos gráficos de baixa e alta resolução e abusando da capacidade colorida do seu micro.

Victor Mirshawka

CONHECENDO E UTILIZANDO O TK-2000

nobel

CONHECENDO E UTILIZANDO O TK-2000



Compatível com o
TK-2000-II

VICTOR MIRSHAWKA

ISBN 85-213-0258-4

nobel

Litec

LIVRARIA EDITORA TÉCNICA LTDA

Rua dos Timbiras 257-CEP:01208-São Paulo

Caixa Postal 30 869 -Tel.: 220-8983

REF **433** PREÇO **25600**

CONHECENDO E UTILIZANDO O

TK-2000

84-1693 18. -001.642 18. -001.642 17. CDB-621.8

- Índices para cartões estatísticos:
1. Microcomputadores ; Programação ; Processamento de dados 621.8 (17) 001.642 (18.)
 1. Programação ; TK-2000 color ; Computadores ; Processamento de dados 621.8 (17) 001.642 (18.)
 3. TK-2000 color ; Computadores ; Programação ; Processamento de dados 621.8 (17) 001.642 (18.)

LIVRARIA EDITORA TÉCNICA LTDA
RUA DOS TIMBIRAS, 257 - SÃO PAULO - SP
CAIXA POSTAL 30.869 - TEL. (011) 220-8983

OUTRAS OBRAS DO AUTOR

Caderno de elementos de computação (Linguagem FORTRAN)
Linguagem BASIC
BASIC sem segredos
TK-divertindo
TK-lembrando
TK-calculando
TK-2000 na matemática
TK-2000 nos gráficos e na música
TK-2000 nas ciências humanas
Dê um APPLE à sua vida
TK S800 resolvendo seus problemas
Jogos e desenhos com o TK S800
Imprimindo maravilhas com a GRAFIX

CIP-Brasil. Catalogação-na-Publicação
Câmara Brasileira do Livro, SP

Mirshawka, Vítor, 1941-
M655c Conhecendo e utilizando o TK-2000 / Victor
Mirshawka. -- São Paulo : Nobel, 1984.

Bibliografia.
ISBN 85-213-0258-4

1. Microcomputadores - Programação 2. TK-2000
color (Computador) - Programação I. Título.

84-1693

17. CDD-651.8
18. -001.642
18. -001.6425

Índices para catálogo sistemático:

1. Microcomputadores : Programação : Processamento de dados 651.8 (17.) 001.642 (18.)
2. Programação : TK-2000 color : Computadores : Processamento de dados 651.8 (17.) 001.6425 (18.)
3. TK-2000 color : Computadores : Programação : Processamento de dados 651.8 (17.) 001.6425 (18.)

DATILOGRAFIA: Toshiro Iqueda
ILUSTRAÇÕES E CAPA: Lírio F. Yuasa

VICTOR MIRSHAWKA

Professor titular de Cálculo Numérico, Estatística e Pesquisa Operacional da Faculdade de Engenharia da Fundação Armando Álvares Penteado.

Professor titular de Cálculo Numérico e Ciência da Computação, Estatística e Pesquisa Operacional da Universidade Mackenzie.

Professor associado do departamento fundamental da Escola de Engenharia Mauá.

Mestre em Estatística pela Universidade de São Paulo.

CONHECENDO E UTILIZANDO O TK-2000

3ª edição

1985



LIVRARIA NOBEL S.A.
EDITORA — DISTRIBUIDORA
LOJA 1: RUA DA CONSOLAÇÃO, 49 - CEP01301
LOJA 2: RUA MARIA ANTONIA, 108 - CEP01222
LOJA 3: RUA PEDROSO ALVARENGA, 704 - CEP04531
LOJA 4: RUA BARÃO DO TRIUNFO, 371 - CEP04602
EDITORA: RUA DA BALSÁ, 559 - CEP02910
FONES: (FABX) 857-9444 e 257-2144 - SÃO PAULO - SP

Apresentação

É PROIBIDA A REPRODUÇÃO

Nenhuma parte desta obra poderá ser reproduzida, sem a permissão por escrito dos editores, através de qualquer meio: XEROX, FOTOCÓPIA, FOTOGRAFICO, FOTOMECÂNICO. Tampouco poderá ser copiada ou transcrita, nem mesmo transmitida através de meios eletrônicos ou gravações. Os infratores serão punidos através da Lei 5.988, de 14 de Dezembro de 1973, artigos 122-130.

Entre os objetivos que tive em vista ao escrever este livro, o primeiro foi o de fazer uma obra acentuadamente didática, isto é, um livro que pudesse ser útil a todos. Entretanto, não dei atenção apenas à parte didática, mas também à parte prática.

Dessa maneira, elaborei e comentei programas que servem para a matemática, a física, o desenho, a música e para o seu divertimento em geral, caro(a) leitor(a), procurando dessa forma utilizar quase todas as instruções ou comandos da linguagem APPLESOFT BASIC, usada no seu TK-2000 COLOR. Aliás, esta é a linguagem utilizada em todos os microcomputadores APPLE e similares.

Já fiz uma série de livros para o TK-85, mais ou menos no mesmo estilo e, devido à boa recepção que tiveram entre os "computamaniacos", procurei imprimir uma diretriz semelhante ao Conhecendo e Utilizando o TK-2000.

Agora, porém, as explicações sobre os comandos utilizados nos programas são apresentadas à medida que elas aparecem nos próprios. A descrição do uso correto dos diversos comandos não é muito profunda, mas explica-se o suficiente para compreendê-los adequadamente.

Caso o(a) estimado(a) leitor(a) necessite de mais detalhes, recomendo recorrer a um outro livro meu - Dê um APPLE à sua vida - mas sinceramente espero que isto não seja necessário. O(A) caridoso(a) leitor(a) pode evidentemente, para aumentar os seus conhecimentos, adquirir o livro citado, mas não para "acompanhar" especificamente o que ora está lendo...

O segundo objetivo óbvio foi o de tornar mais simples o uso dessa "ferramenta" fantástica que

é o microcomputador TK-2000 COLOR da MICRODIGITAL. É evidente que todos sabem que o TK-2000 COLOR tem bom padrão de qualidade, tem um preço justo, usa um meio barato de armazenamento (gravador cassete comum) tem excelente "design", permite o acionamento de uma impressora, pode ser conectado a um monitor de vídeo profissional além de estar ligado a uma T.V. colorida, tem 16 K bytes de ROM e 64 K bytes de RAM, permite ao(a) usuário(a) a criação de varia dos efeitos visuais diretamente pelo APPLESOFT BASIC, sem necessitar gerar estes efeitos através da linguagem de máquina, pois possui 50 caracteres gráficos, incorpora uma saída especial para joystick, tem lindas cores e um belo som etc.

Bem, se você não sabia especificamente disto, ficou sabendo...

Acredito que a forma mais simples de aprender a usar uma ferramenta tão poderosa com o TK-2000 COLOR é sentindo o que ele pode fazer. Isto o(a) interessado(a) leitor(a) notará e entenderá após teclar todos os programas deste livro. Tornar-se-á também um "expert" no assunto quando conseguir fazer todas as "tarefinhas" indicadas no fim de cada programa.

Ao todo apresentei 26 programas-título, porém em alguns casos tem-se mais que um programa. Para os(a) leitores(as) mais abonados informo que pode ser adquirido, através de um pedido formal à Livraria Nobel S.A., um disquete com todos os programas (menos aqueles que você deve preparar...). Isto evidentemente lhe trará uma economia de tempo, porém retardará o contato mais íntimo e o domínio completo do TK-2000 COLOR da sua parte.











Pressionando continuamente as teclas você vai fazer "mil e uma cócegas" no seu TK-2000 COLOR, que sucumbirá a essas carícias transformando-se em seu submisso servo...

Acredito que este livro com ou sem disquete é um software simples e barato, e desvendará o enorme leque de possibilidades do TK-2000 COLOR.

Agradecendo desde já ao(a) usuário(a) deste livro, recomendo também os outros livros meus para o TK-2000, no mesmo estilo:

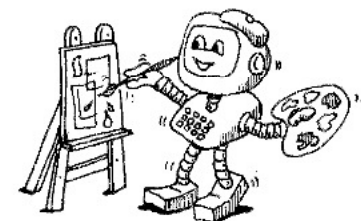
- "TK-2000 na Matemática"
- "TK-2000 na Física"
- "TK-2000 nas Ciências Humanas"
- "TK-2000 nos Gráficos e na Música"

O terceiro aspecto que me "incentivou" a produzir este Conhecendo e Utilizando o TK-2000 é que ele pode ser útil para:

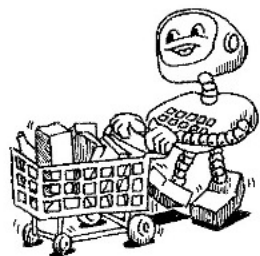
- a) pessoas sofisticadas; 
- b) casais comuns; 
- c) casais estranhos; 
- d) casais separados; 
- e) tocar música sintetizada nos casamentos; 
- f) o macho chauvinista; 
- g) uma família; 
- h) uma família um pouco estranha; 
- i) uma grande família; 
- j) eliminar discussões entre elementos muito agressivos de uma família. 

Finalmente, um quarto e último aspecto que me impeliu a trazer-lhe, respeitado(a) leitor(a) essas informações foi a certeza de que o TK-2000 COLOR poderá auxiliá-lo(a) a:

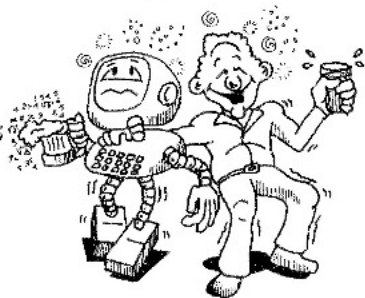
- 1) ser um grande artista;



2) fazer compras adequadamente;



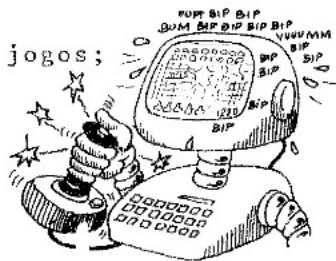
3) tornar-se um matemático
"exponencialmente" feliz;



4) distrair-se;



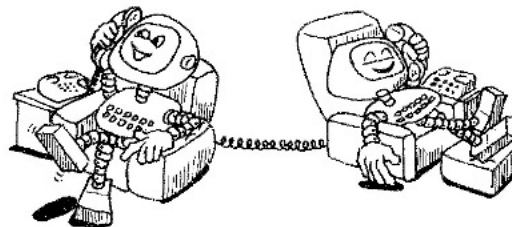
5) aprender os mais diversos jogos;



6) transformar-se em um grande músico;



7) ligar-se a quem desejar;



8) ter uma fonte inesgotável de informações.



Agradecimentos

Feliz possuidor de um TK-2000 COLOR não resista e atenda o pedido da moça...



Figura 0 - "Venha comigo que vou lhe mostrar o TK-2000 COLOR..."

Foi contínua e cada vez mais indispensável a ajuda da minha querida esposa e amiga Nilza Maria e dos meus dois filhos maiores, Victor Junior e Sergio, na elaboração deste livro. Cumpre destacar que o terceiro filho, Alexandre (Sacha, para os íntimos) se não ajudou muito, também não atrapalhou...

Sem dúvida nenhuma, sem o auxílio deles não poderia jamais apresentar, em prazo tão curto, uma série de textos, diferenciados quanto ao conteúdo específico, porém todos eles voltados para o uso do microcomputador.

Acredito muito hoje (e o TK-2000 foi fundamental nisto) na frase: "Uma família unida trabalha e se diverte unida..."

Muito obrigado a George Kovari, diretor da MICRODIGITAL, por ter permitido ser eu um dos primeiros a possuir e a usar um TK-2000 COLOR. Sem isto seria impossível escrever este livro.

"Thank you", Toshiro Iqueda, datilógrafo incomparável e amigo de muitos anos, que decifrou os originais e os transformou em mais um livro.

"Merci", Angélica Ayres e novamente Toshiro Iqueda, pelo excelente trabalho de editoração.

Finalmente, sou grato mais uma vez aos amigos da Livraria Nobel S.A., Ary K. Benclowicz e Luigi Zamboni, por acreditarem na validade deste trabalho, colocando à nossa disposição todo o setor de publicação e divulgação.

Sumário

1 - Números perfeitos, abundantes e deficientes	1
2 - Números aleatórios	21
3 - A fórmula de Heron	31
4 - O teorema de Pitágoras e as triplas pitagóricas	39
5 - Cinemática	51
6 - Calorimetria	59
7 - Resistências em séries e paralelo	71
8 - Lei de Ohm	75
9 - Desenhando com o PRINT	81
10 - Maçã estilizada	85
11 - Bandeiras nacionais	93
12 - O Pisca-pisca	101
13 - Simetrias gráficas	107
14 - Caleidoscôpio	113
15 - Traçando em alta resolução	117
16 - Gráficos radiais	123
17 - Reticulados sofisticados	129
18 - Manipulação de triângulos retângulos e paralelogramos	133
19 - Divertindo-se com a geometria	137
20 - Espirais	141
21 - Movimentando a ponta de uma seta	145

22 - Voando 155
 23 - Fogos de artifícios e figuras de Lissajou. 159
 24 - Orbitando 163
 25 - Rolando 173
 26 - Esquiando 179
 Relação de software disponível para o TK-2000
 COLOR 185
 Impressora GRAFIX 189
 Bibliografia 191

Números perfeitos, abundantes e deficientes

Santo Agostinho compreendeu o primitivo anseio dos hititas pela perfeição, quando afirmou "ter Deus criado todas as coisas em seis dias, por ser o seis um número perfeito".

$$\begin{array}{rcc}
 1 + 2 + 3 = 6 & & \\
 \downarrow & & \downarrow \\
 \text{Divisores} & & \text{Perfeito} \\
 \text{próprios} & &
 \end{array}$$

O traço característico da primitiva ciência numérica dos chineses é que nela os números eram *figurados* ou seja representados por traços, linhas, círculos e pontos (veja a Figura 1).

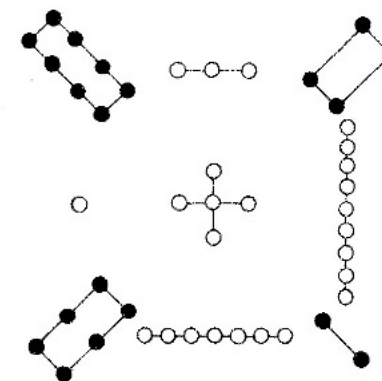


Figura 1 - Representação do primeiro quadrado mágico da história construído originalmente cerca de 1000 anos antes de Cristo. Livro chinês das Permutações .

Para nossos propósitos, isto é muito importante pois ajuda a reconstituir as origens de um ramo muito útil da Matemática que é o *estudo das séries*.

No Livro Chinês das Permutações, escrito cerca de 500 anos antes que Pitágoras tenha nascido, representavam-se os oito primeiros números por combinações de traços horizontais ora inteiros (-), ora interrompidos, (- -), os primeiros representando o princípio *macho* dos números ímpares e os últimos o princípio *feminino* dos números pares.

A separação, na primitiva ciência numeral dos chineses dos números pares e ímpares, aliás fêmeas e machos mostra a preocupação do homem primitivo com a fertilidade do seu rebanho, de seus campos e com o grupo familiar patriarcal.

O tratamento "animístico" dado aos números nos primeiros livros chineses de Matemática (veja a Figura 2) entrelaçava-se, aparentemente, com o re-

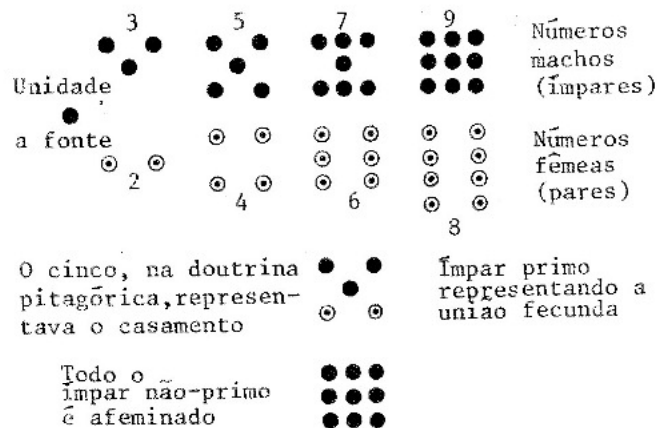


Figura 2 - O animismo e o gênero, na ciência numeral dos chineses e dos pitagóricos.

conhecimento de uma classe especial de números ímpares, os hoje chamados primos que analisaremos com mais detalhes no 2º programa.

Números primos são aqueles insuscetíveis de divisão em um número exato (inteiro) de números inteiros.

É por isso que eles são insuscetíveis de representação em grupos equivalentes (veja a Figura 2) de pontos ou círculos

3,5,7,11,13, etc. são primos
 9,15,21,25,etc. embora ímpares não são primos.

A interpretação de Pitágoras sobre os números era a seguinte: o 1(um), tido como a fonte de todos os outros números era a *razão*, o 2(dois) era a *opinião*, o 3(três) era o primeiro número *macho* e era a *potência*, o 4 (quatro) era a *justiça*, o 5(cinco) era o *matrimônio* pois era a fusão do 3 (primeiro número macho) e do 2 (primeiro número fêmea).

No 5 (cinco) residia o segredo da cor, no 6 (seis) o segredo do frio, no 7 (sete) o segredo da saúde, no 8 (oito) o segredo do amor isto é 3(potência) "ajuntado" a 5 (casamento).

O sólido de seis faces encerrava o segredo da Terra.

A pirâmide continha, o segredo do fogo.
 O sólido de doze faces continha o segredo dos ceus.

A esfera era a figura mais perfeita.
 Havia os números perfeitos, abundantes, deficientes, amigáveis etc.

Nicômaco de Alexandria perdeu muito tempo para mostrar que 496 e 8128 são números perfeitos.

O pior é que o próximo número perfeito é 33 550 336.

Nicômaco num esforço improfícuo de alcançá-lo concluiu amargamente que: "o feio e o mal são prolíficos ao passo que o belo e o bom são raros e contáveis nos dedos".

Observação Importante - O.I.

Toda vez que aparecer o zero (0) o mesmo será representado por \emptyset para não confundir com a letra O.

Isto parece brincadeira mas não é!!!!

Com esta distinção você evitará muitos erros no futuro.

No nosso primeiro programa estudaremos os números que podem ser definidos e classificados em função da soma dos seus *divisores próprios*.

Definições

1 Entende-se por *divisores próprios* de um número inteiro positivo a todos os divisores inteiros que são menores que o próprio número.

Por exemplo, os divisores próprios de 10 são 1, 2 e 5.

2 Um número é *abundante* se a soma dos seus divisores próprios é maior do que o número em si. Assim o número 12, que tem como divisores 1, 2, 3, 4 e 6 com soma:

$$1 + 2 + 3 + 4 + 6 = 16 \quad (16 > 12)$$

é um número abundante.

3 Um número é *deficiente* se a soma dos seus divisores próprios é menor que o número em si. É o caso do 10, pois:

$$1 + 2 + 5 = 8 < 10$$

4 Um número é *perfeito* quando a soma dos divisores próprios é igual ao número em si. Por exemplo, o número 6 é perfeito pois tem como divisores

$$1, 2 \text{ e } 3 \text{ e } 1 + 2 + 3 = 6$$

O nosso problema, tendo entrado com um número, é saber se ele é perfeito, abundante ou deficiente!!!!?!!!!

ANÁLISE DO PROBLEMA

Para determinar se um número N é perfeito, abundante ou deficiente precisa-se comparar a soma dos seus divisores S com o próprio número N.

- I) Se $S > N$ então o número é abundante
- II) Se $S = N$ então o número é perfeito
- III) Se $S < N$ então o número é deficiente

Algoritmo para obter a soma dos divisores próprios de um número N, que é um inteiro positivo maior que 1.

Em primeiro lugar deve-se entrar com N, a seguir deve-se inicializar a soma S em 1 e o divisor em 2.

Caso o divisor D seja maior que o quociente N/D deve-se comparar S com N imprimir a mensa-

gem correspondente e ir para (GOTO) 1 [veja o fluxograma da Figura 3].

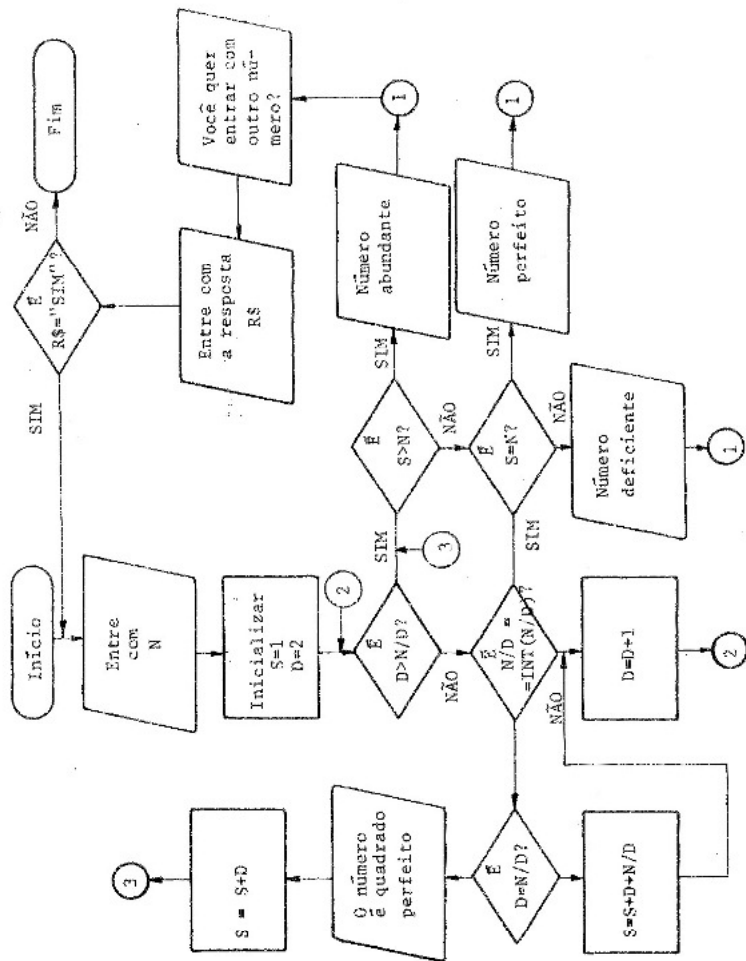


Figura 3 - Fluxograma para facilitar o entendimento do programa

Se por outro lado D não for maior que N/D deve-se continuar.

Faz-se o teste, se $N/D = \text{INT}(N/D)$ então D é um divisor e devem ser feitas outras verificações como por exemplo se $D = N/D$ então o número N é um *quadrado perfeito* quando então usa-se a instrução de atribuição $S = S + D$, imprime-se a mensagem, o valor de S e deve-se ir para ③.

Se por outro lado $D \neq N/D$ então o D e o quociente (N/D) são fatores, usa-se a instrução de atribuição $S = S + D + N/D$ e a seguir deve-se aumentar D de uma unidade e ir para ③ [veja o fluxograma da Figura 3].

Finalmente se $N/D \neq \text{INT}(N/D)$ então D não é um divisor de N , aumenta-se D de 1 e deve-se ir para ②.

Aí vai o plano final para o "ataque"

- 1 - Entre com o número N (maior do que 1).
- 2 - Ache a soma dos divisores próprios de N ; chame essa soma de S .
- 3 - a) Se $S > N$ então imprima "NÚMERO ABUNDANTE" e vá para ①
 b) Se $S = N$ então imprima "NÚMERO PERFEITO" e vá para ①
 c) Se $S < N$ então imprima "NÚMERO DEFICIENTE" e vá para ①.

Observação Importante (O.I)

Símbolos usados nos fluxogramas e o seu significado.



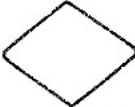


Símbolo	Significado
	O micro isto é o seu TK-2000 deve começar ou terminar um programa.
Início/Fim	
	O operador (você) deve entrar com informação dentro do TK-2000 COLOR ou pede-se ao TK-2000 que imprima uma informação que já foi processada.
	O seu TK-2000 COLOR vai decidir entre intruções alternativas.
DECISÃO	
	O TK-2000 COLOR atribue a variável no membro da esquerda ou seja à esquerda do sinal de igual (=) o valor que está à direita.
Atribuição	
	O programa vem de ou vai para outra parte do fluxograma.
Conectar	

Figura 4

E finalmente chegamos ao programa em BASIC para o seu TK-2000 COLOR.

```

10 PRINT "ABUNDANTE, PERFEITO OU DEFICIENTE"
20 INPUT "ENTRE COM UM NÚMERO INTEIRO";N
30 S = 1
40 D = 2
50 IF D > N/D THEN 90
60 IF N/D = INT (N/D) THEN 130
70 D = D+1
80 GOTO 50
90 IF S > N THEN PRINT N;"É UM NÚMERO ABUNDANTE":GOTO 120
100 IF S = N THEN PRINT N;"É UM NÚMERO PERFEITO":GOTO 120
110 PRINT N;"É UM NÚMERO DEFICIENTE"
120 PRINT : PRINT : GOTO 200
130 IF D = N/D THEN PRINT "ANTES DE MAIS NADA O NÚMERO";
    N;" É UM QUADRADO PERFEITO": S=S+D:GOTO 90
140 S = S + D + N/D : GOTO 70
200 FOR I = 1 TO 40
205 PRINT "*";
210 NEXT
220 PRINT "VOCÊ QUER ENTRAR COM OUTRO NÚMERO?(SIM/NÃO)"
230 INPUT R$
240 IF R$ = "SIM" THEN HOME : GOTO 10
250 END.

```

Comentários resumidos sobre as instruções BASIC do seu TK-2000 COLOR.

- 1) **PRINT** (veja as linhas de números 10, 90, 100, 110, 120, 130, 205, 220).

PRINT (expressões) (; ou ,)

Com essa instrução imprime-se expressões de diversas formas no modo texto.

Existem três modos de operação na tela.

Um deles é usado para a apresentação de textos (modo texto) em preto e branco e os outros dois que usaremos principalmente no livro "TK-2000 no gráfico e no som" são usados fundamentalmente para gráficos coloridos.

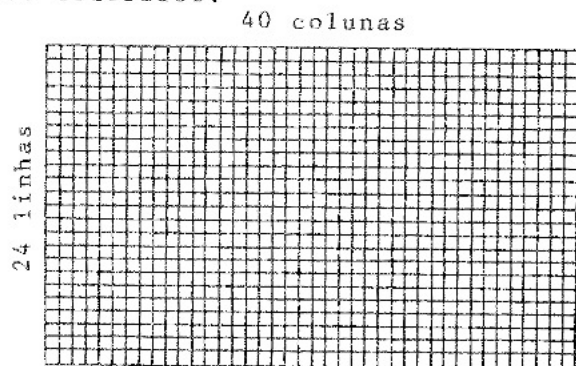


Figura 5

No modo texto, a tela pode conter até 24 linhas de 40 caracteres cada uma (Figura 5).

O sinal de interrogação (?) pode ser utilizado como abreviação da palavra reservada PRINT e quando se pedir um LIST será impresso como PRINT.

Aliás com a instrução LIST todas as linhas de comando do programa são apresentadas, ou seja, listadas na tela.

Pode-se também listar apenas um segmento ou trecho do programa utilizando-se

LIST X,Y ou LIST X-Y ou LIST-Y ou LIST X-

onde X é a primeira linha que se quer listar e Y é a última.

Sem qualquer opção, PRINT provocará a execução de um salto de linha e a volta do cursor à primeira posição da linha seguinte na tela (veja a linha 120 onde saltamos duas linhas).

Opcionalmente pode-se especificar uma lista de expressões separadas por vírgula e ponto e vírgula que será avaliada e impressa item por item (veja o PRINT na linha 130).

Se um item numa lista é seguido por um ponto e vírgula, então o próximo item é concatenado, isto é, impresso sem espaços separando-o do item anterior (veja a linha 205).

Para que um programa apresente o conteúdo fixo ou que utilize caracteres alfabéticos deve-se usar a instrução PRINT, após a qual digitar este conteúdo entre *aspas*.

2) HOME

O sinal de COMANDO (>) que você vê na primeira posição da tela (primeira coluna da primeira linha), indica que o TK-2000 COLOR está apto para receber instruções através do teclado.

O cursor (■) sempre indica a posição da tela na qual será registrado o próximo caractere digitado no teclado.

Quando se utiliza a instrução HOME (linha 240) todos os caracteres são sumprimidos e o cursor se posiciona na primeira coluna da primeira linha.

3) INPUT (veja as linhas 20 e 230)

O formato usado por esta instrução é:

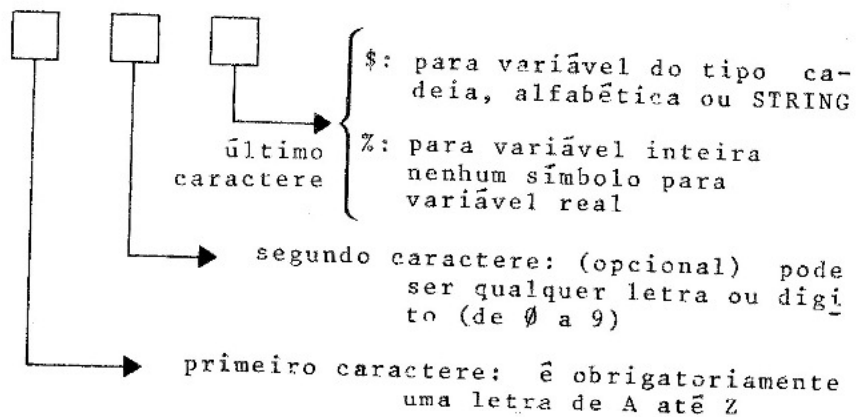
```
INPUT nome de variável (,nome de variável, ...)
```

Esse comando, permite uma associação entre nome de variável e o seu valor.

Vejamos então como se escolhe o nome de uma variável.

No BASIC do TK-2000 COLOR o nome de uma variável pode ter um, dois ou três caracteres.

Para determinar este nome deve-se observar as seguintes regras.



Ressaltamos assim que pode-se ter três tipos de variáveis de acordo com o tipo de dados que elas contêm: as variáveis tipo cadeia, que são relacionadas com qualquer conjunto de até 255 caracteres (com exceção das aspas), as variáveis inteiras que correspondem a números sem parte fracionária e as variáveis reais associadas a qualquer número inteiro, fracionário ou com parte inteira e fracionária.

No nosso primeiro programa todas as variáveis são reais com exceção de R\$ que é uma variável do tipo cadeia que pode inclusive tomar o valor "SIM".

A instrução 230 INPUT R\$ executa as seguintes operações

- apresenta um ponto de interrogação na tela;
- aciona o cursor;

c) espera pela digitação da variável alfabética.

Quando a cadeia é digitada e a tecla RETURN é calcada, faz com que a mesma seja associada a variável R\$ e só depois disso é que se permite que o programa prossiga para a próxima linha de comando.

No caso da linha 20 temos inicialmente a apresentação na tela da frase "ENTRE COM UM NÚMERO INTEIRO" e aguarda-se a introdução de um dado que é associado ao nome da variável real N.

4) MODO PROGRAMADO

É evidente que o seu TK-2000 COLOR está pronto para ser usado no modo imediato, também chamado de direto ou modo calculadora.

Nesta forma de operação o micro responde imediatamente as instruções desejadas.

Digite a instrução PRINT 385*44 e pressione a tecla RETURN.

Surgirá na linha seguinte do vídeo a resposta do produto ou seja 16940.

O modo imediato é porém muito limitado em termos da capacidade real de um micro.

É no modo programado que o micro TK-2000 poderá ter todo o seu potencial explorado.

Para se executar um programa é imprescindível que cada linha do mesmo seja relacionada a um número inteiro positivo.

Não existem números fixos a serem utilizados para esta numeração mas, comumente as linhas são numeradas em ordem crescente de dez em dez, no decorrer do desenvolvimento do programa.

Toda instrução, precedida com um número de linha, uma vez terminada a sua digitação deve ser seguida de um RETURN.

A tecla RETURN indica ao TK-2000 COLOR que a linha de comando foi terminada.

Você irá notar que o cursor vai para a próxima linha depois de ter sido pressionada a tecla RETURN porém a instrução não foi executada.

Para que uma instrução ou um conjunto de instruções (programa) seja executado é necessário usar a instrução `RUN`.

`RUN` é a instrução que "dá partida" na execução de um programa ou como se costuma dizer na gíria dos "computacas", faz com que o programa "rode".

As vezes, para analisar um programa, é útil que este seja executado a partir de uma linha de programa.

Para tanto, podemos usar o comando:

`RUN nulín`

onde *nulín* representa uma linha genérica do programa a partir da qual se quer a execução do mesmo.

Não esqueça portanto no fim do programa de digitar a palavra RUN em modo imediato e a seguir pressionar a tecla RETURN.

5) `LET` (linhas 30, 40, 70, 130 e 140)

Através do comando LET relacionamos ou melhor atribuímos um valor numérico ou alfabético, con forma o caso, a uma variável.

No seu TK-2000 COLOR o comando LET é opcional, de maneira que podemos omiti-lo para executar

a associação entre um nome de variável e o seu valor.

Foi aliás o que fizemos nas linhas 30, 40, 70, 130 e 140.

6) `IF-THEN nulín`

A instrução IF-THEN é uma instrução de desvio condicionado.

Sempre que em um programa a sequência numérica de execução das linhas não é obedecida diz-se que ocorreu um *desvio*.

Caso a afirmação ou a expressão matemática que inclui um operador relacional após o IF for verdadeira desviamos o programa para uma linha de número *nulín*.

Se por outro lado a condição após o IF não for verdadeira o programa prossegue normalmente na próxima linha.

Dessa maneira na linha 50 se D for maior que N/D vai-se para a instrução de número de linha 90 e em caso contrário vai-se para a linha 60.

Na linha 60 se N/D for igual a parte inteira de (N/D) então vai-se para a linha 130 e em caso contrário atinge-se a linha 70.

Temos ainda IF-THEN nas linhas 90, 100 e 240.

7) `GOTO nulín`

O comando GOTO permite desviar a execução do programa para qualquer número de linha do mesmo de forma incondicional.

Assim uma vez atingida a linha 80 o GOTO que aí aparece faz retroceder o programa para a linha 50.

É semelhante a interpretação dos GOTO que aparecem nas linhas 90, 100, 120, 130, 140 e 240.

8) O USO DE DOIS PONTOS (:) NUMA LINHA DE PROGRAMA

Através do uso de dois pontos (:) podemos incluir mais do que um comando numa única linha de programa.

Isto tem como vantagem por exemplo a minimização do esforço de digitação.

Tem também alguns inconvenientes.

As linhas 90, 100, 120, 130, 140 e 240 são portanto linhas com instruções múltiplas.

9) O PAR FOR-NEXT

A instrução FOR-NEXT é na realidade dividida em duas ordens que são: a ordem FOR e a ordem NEXT.

Ela tem por finalidade básica repetir um determinado número de vezes, as instruções que se encontram entre as ordens FOR e NEXT.

No nosso caso por exemplo entre o FOR (linha 200) e o NEXT (linha 210) só temos uma instrução que é a 205 PRINT "*";

O FOR é sempre seguido de uma variável à qual se atribui um valor inicial (no nosso caso I = 1) após o que aparece a palavra chave TO e finalmente após a mesma vem o valor máximo que esta variável pode assumir (no nosso caso I = 40).

O número de vezes que será repetido o "PRINT "*";" é igual a

$$\langle n \rangle = (\langle \text{valor final} \rangle - \langle \text{valor inicial} \rangle + 1)$$

No nosso caso particular

$$\langle n \rangle = 40 - 1 + 1 = 40 \text{ vezes}$$

Como no PRINT se pede a impressão de um asterisco (*) e como a próxima impressão é concatenada ou justaposta obteremos uma linha de 40 asteriscos.

Você não esqueceu que em uma linha no modo texto cabem exatamente 40 caracteres, esqueceu?

Na realidade esta é uma explicação bem resumida do poderoso e importante par FOR-NEXT.

Falaremos muito dele daqui para frente...

10) Indicamos com $\text{\textcircled{V}}$ o espaço branco muito importante por exemplo na linha 110 para que o valor da variável N não saia "encostado" na mensagem "É UM NÚMERO DEFICIENTE".

A mesma interpretação é válida para as linhas 90, 110 e 130.

11) END

A instrução END (linha 250) faz com que a execução de um programa seja interrompida ou finalizada.

Usaremos sempre o END para finalizar os programas deste livro, porém como "dica" para economia de espaço e tempo no futuro não use o END pois a sua omissão não trará nenhum malefício e fará apenas com que o programa pare no último comando executado.

Agora que está tudo explicadinho que tal você verificar que 496, 8128 e 33550336 são todos perfeitos. Quem vai ficar agradecido é o Nicômaco!!!!!!!

Um pouco mais de trabalho para
você caro(a) leitor(a)

Para cada um dos problemas abaixo apresente:

- um plano para resolver o mesmo;
- o desenho do fluxograma correspondente;
- a codificação em BASIC para o TK-2000

COLOR.

I) Entre com um número inteiro positivo e obtenha todos os divisores do mesmo.

II) Dois números chamam-se "amigáveis" ou "cordiais" se a soma dos divisores próprios de cada um é igual ao outro número.

Foi o famoso matemático suíço Leonard Euler (Figura 6) que nasceu em 1707 e morreu em 1783



Figura 6

que deu a primeira grande lista dos números amigáveis.

Nicolò Paganini, com dezesseis anos de idade em 1866 achou um par de números amigáveis que Euler tinha esquecido na sua lista.

O par de números é 1184 e 1210.

Os divisores de 1210 são

1, 2, 5, 10, 11, 22, 55, 110, 121, 242, 605
cuja soma é 1184.

Por sua vez os divisores próprios de 1184 são

1, 2, 4, 8, 16, 32, 37, 74, 118, 296, 592
cuja soma é 1210.

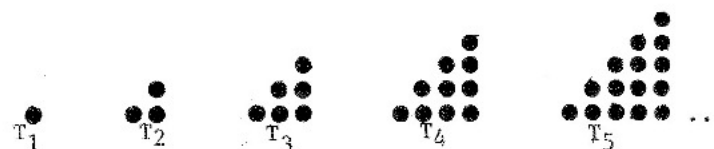
Explicado o que são números amigáveis obtenha com o auxílio do seu prestativo e "amigável" TK-2000 COLOR o número amigável de 220.

Será que o seu TK-2000 dá como resposta o número 284?

III) Obtenha a soma dos 20 primeiros termos das sequências formadas pelos seguintes números:

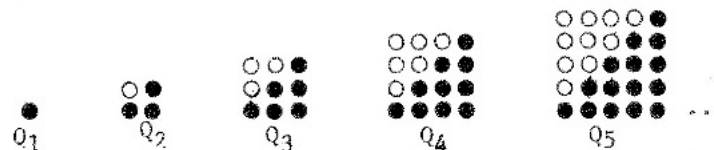
a) Os números triangulares simples

(1... 3... 6... 10... 15... 21... 28... 36, etc.)



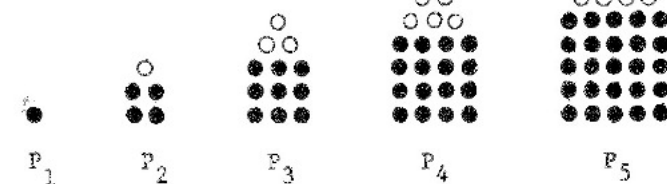
b) Os números quadrados

(1... 4... 9... 16... 25... 36... 49... 64... 81... 100)

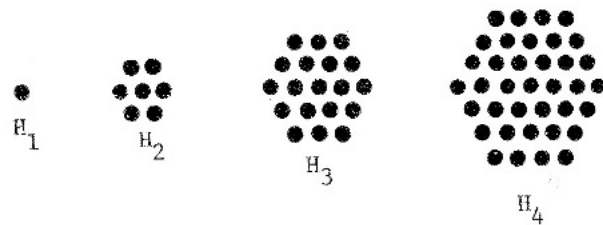


c) Os números pentagonais

(1 - 5 - 12 + 22 - 35 etc)



- d) Os números hexagonais
(1 - 7 - 19 - 37 etc.)



- e) Os números estrelados
(1 - 8 - 21 - 40 etc.)

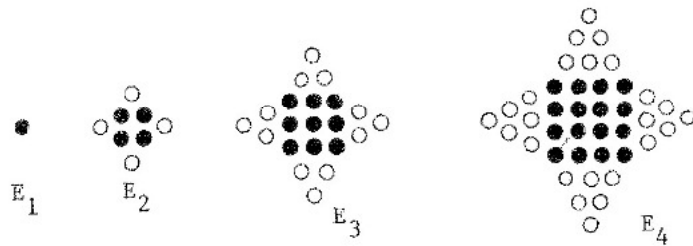


Figura 7

O.I. Como ajuda olhe o programa Nº 6 e a Tabela 10.

Números aleatórios

Diga a cor que vier primeiro a sua mente.
Escolha um número entre 1 e 10.

Ao se fazer este tipo de pergunta os psicólogos verificaram que metade das pessoas aproximadamente escolhem a cor vermelha.

O azul também é muito escolhido, porém isto não ocorre com a mesma frequência com o roxo ou o "rosa-choque" ou com o cinza, etc.

Da mesma forma parece que existe um "viés" no ser humano, levando-o a propensão de escolher na faixa indicada com mais frequência números entre 3 e 7.

Claro que o caçula Sacha escolheria sempre o 7, ou alguém que tem tendência para ser ponta direita fixo.

Caso você seja um(a) futuro(a) grande matemático(a) é capaz de escolher o e (base dos logaritmos neperianos e veja o Neper na Figura 8), o π (faremos muito sobre ele adiante) o $\sqrt{3}$ ou quem sabe 3,1873459.



Neper
(1550-1617)

Figura 8

Caso você queira sentir esse desvio ou vies dos seres humanos para um dado número faça um experimento com os colegas da sua classe.

Verifique até que ponto eles escolhem os números aleatoriamente ou seja ao acaso.

Porém o que é um número aleatório?

Resposta: Um único número ou dígito é aleatório se a sua escolha é tal que existe uma *probabilidade igual* que ele seja qualquer dígito entre os dígitos

0,1,2,3,4,5,6,7,8 e 9

Um número com mais de um dígito é aleatório se cada número com o mesmo número de dígitos tem a mesma probabilidade de ser escolhido.

Experimentos mostram que qualquer ser humano escolhido para gerar uma extensa lista de números aleatórios "contradiz" a definição ora enunciada por causa de particulares preferências por números pares ou ímpares, menores ou maiores e principalmente pela repetição de certos dígitos.

Felizmente o seu TK-2000 COLOR entende o BASIC e através dele você pode fazer com que o seu micro entenda a função RND com a qual ele torna-se um gerador "perfeito" de números aleatórios entre 0 e 1 (não incluído).

Com o seguinte programa obteremos uma tabela de trinta e seis números aleatórios.

```

10 PRINT "NÚMEROS ALEATÓRIOS"
20 FOR I = 1 TO 12
30 FOR J = 1 TO 3
40 R = RND(1)
50 PRINT R; " ";
60 NEXT J
70 PRINT
80 NEXT I
90 END

```

saem 12 linhas de números aleatórios

saem três números por linha

Caso você queira apenas números entre 0 e 99 o que deveria ser feito?

Resposta: Para criar um número entre 0 e 99 deve usar a instrução

```
40 R = INT(100 * RND(1))
```

Caso você queira incluir o 100 e eliminar o 0 basta escrever

```
40 R = INT(100 * RND(1)) + 1
```

Comentário:

Na linha 40 do programa "NÚMEROS PRIMOS" aparece uma função aleatória ou randômica RND().

Existem três tipos de funções randômicas no TK-2000 COLOR:

$$\begin{cases} \text{RND}(1) \\ \text{RND}(\emptyset) \\ \text{RND}(-X) \end{cases}$$

a) RND(1) fornece um número aleatório entre

$$\emptyset \text{ e } \emptyset.999999999$$

b) RND(\emptyset) repete o número fornecido pela instrução RND anterior.

c) Cada RND(-X) seleciona um número diferente e de valor bastante pequeno.

Agora vai um probleminha de probabilidade.
Qual é a probabilidade de escolher aleatoriamente um número entre 1 e 1000 que seja primo?

No nosso livro "Probabilidades e Estatística para Engenharia" vol.1 definimos o que é probabilidade, eventos, variável aleatória discreta e contínua, etc.

No momento, de uma forma compacta (e um pouco imprecisa) diremos que "probabilidade é a medida de uma incerteza".

Assim a probabilidade de sair a face 1 no lançamento de um dado é 1/6 pois existe apenas uma possibilidade de ocorrer esse evento e o número total de eventos igualmente prováveis é seis.

Dessa maneira a probabilidade que um evento A ocorra é dada pelo quociente do número de formas que esse evento possa ocorrer $[n(A)]$ sobre o número total de eventos igualmente prováveis $[n(E)]$.

Costuma-se chamar a E de espaço amostral.

Portanto a probabilidade de um evento é expressa por:

$$P(A) = \frac{n(A)}{n(E)} \quad (1)$$

e obviamente é um número entre \emptyset e 1 (extremos incluídos).

Análise do problema

Embora possamos contar o número de "primos" menores ou iguais a 1000, podemos nos aproximar do resultado teórico pelo método das tentativas.

Devemos para isso fixar o número de tentativas (T), gerar T números aleatórios entre 1 e 1000, contar os números primos (P) e achar o quociente $(\frac{\Sigma \text{ primos}}{T})$ para obter a probabilidade aproximada.

Definição de número primo

Um número primo é um inteiro positivo maior que 1 e que tem exatamente dois fatores inteiros 1 e ele próprio.

Assim o 7 é um número primo, viu Sacha!!!!

Os fatores inteiros de 7 são 1 e ele próprio.

Já o número 9 não é primo pois os seus fatores são 1, 3 e 9.

Um número que não é primo é chamado de número composto.

Assim para testar se um número é primo precisamos estar seguros que não existem outros fatores além do 1 e do próprio número.

Para se chegar ao número primo serão feitas verificações sucessivas dos divisores (D) do número proposto e a análise do resto.

Como se pode saber se o número de divisores que se experimentou é suficiente?

Resposta: Quando o quadrado de um divisor é maior que o número e não se tem resto zero, o número é primo e não há necessidade de se experimentar divisores maiores.

Análise do fluxograma

a) Usamos a variável C para contar a quantidade de números aleatórios gerados e P para contar os números primos entre os mesmos.

b) Inicialmente verifica-se se o número é par.

Caso o número seja par, precisa se testar se é 2 pois este é o único número par que é primo.

c) Para testar se o número é primo faremos o seguinte teste:

$$D^2 > R \text{ ou o seu equivalente } D > R/D$$

O.I. - Aí vai o fluxograma (Figura 9) porém queremos destacar desde já que no fluxograma não estão indicadas todas as instruções que apresentamos no programa principalmente quando elas se referem ao embelezamento de uma saída (saltar linhas, imprimir tracejados, tabulações, etc.) pois as mesmas não mudam em nada a lógica da solução do problema matemático.

Aliás esse tipo de postura adotaremos em quase todos os fluxogramas que aparecerem pela fren

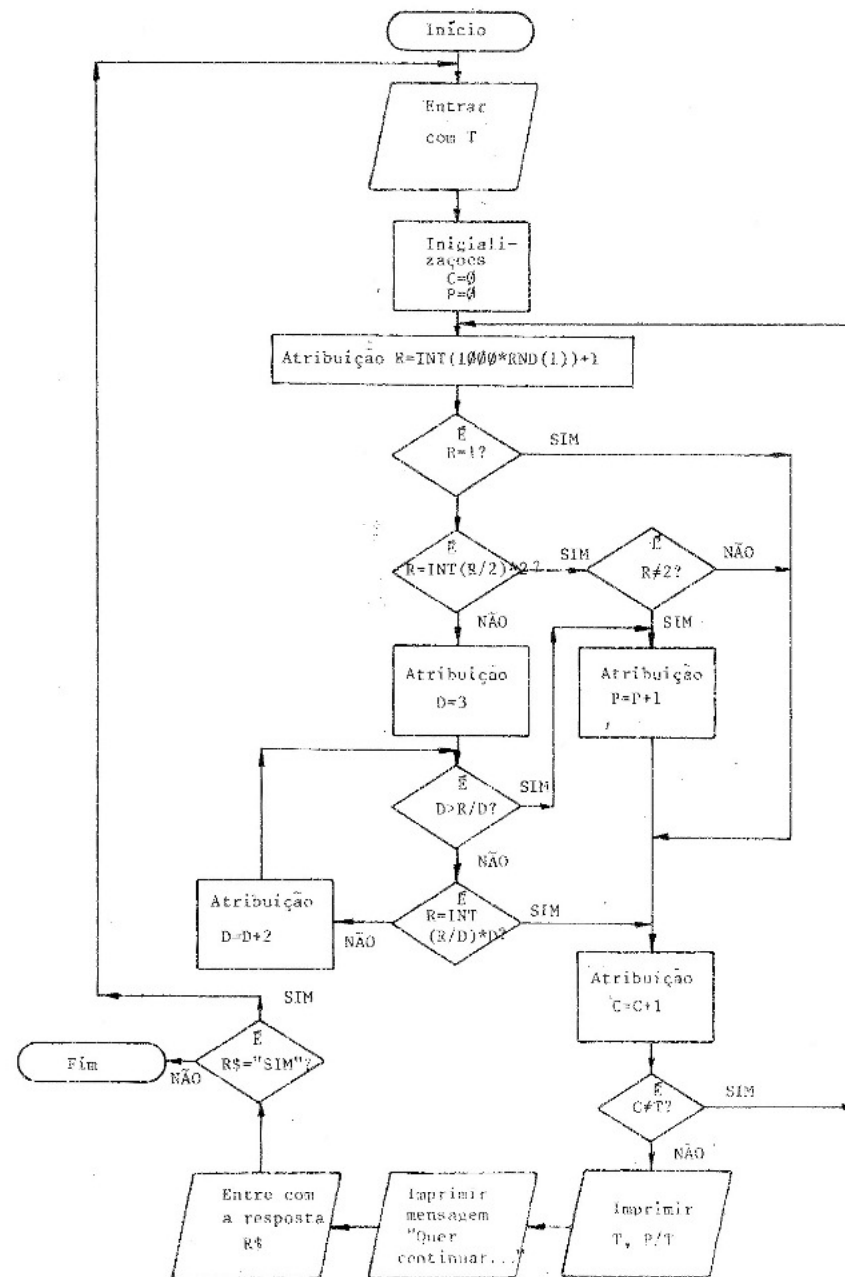


Figura 9

te inclusive indo para a condição de não complicar aquilo que tem como finalidade ilustrar e simplificar...

```

10 PRINT "EXPERIMENTO PROBABILÍSTICO"
15 PRINT "CÁLCULO DA PROBABILIDADE DE SE OBTER NÚMEROS PRI-
    MOS ENTRE 1 E 1000"
20 PRINT: PRINT
25 PRINT "NO DE TENTATIVAS";TAB(20);"PROBABILIDADE"
28 FOR I = 1 TO 40
30 PRINT "-";
35 NEXT I
38 PRINT "ENTRE COM O NÚMERO DE TENTATIVAS";
40 INPUT T
50 PRINT T
55 C = 0: P = 0
60 R = INT(1000 * RND(1)) + 1
70 IF R = 1 THEN 150
80 IF R = INT(R/2) * 2 THEN 130
90 D = 3
100 IF D > R/D THEN 140
110 IF R = INT(R/2)*D THEN 150
120 D = D + 2
130 IF R <> 2 THEN 150
140 P = P + 1
150 C = C + 1
160 IF C <> T THEN 60
170 PRINT TAB(7); T; TAB(25); P/T
180 PRINT "VOCÊ QUER OUTRO EXPERIMENTO? (SIM/NÃO)"
190 INPUT R$
200 IF R$ = "SIM" THEN 28
210 END

```

Comentários:

- 1) As linhas 10 até 35 não aparecem representadas no programa.

Leia o último O.I. para saber a razão dessa omissão.

- 2) Instrução TAB

A instrução TAB é usada para deslocar ou melhor iniciar a impressão de dados a partir de uma coluna qualquer.

O TAB está associado a instrução PRINT.

No caso da linha 25 a mensagem "PROBABILIDADES" começa a ser impressa numa dada linha a partir da coluna 20.

Já no caso da linha 170 o valor da variável T aparece impresso a partir da coluna 7 e o da expressão P/T a partir da coluna 25.

O argumento da função TAB deve estar sempre entre parênteses e o seu valor deve estar entre 0 e 255.

Com a função TAB pode-se mover o cursor para a posição indicada pelo valor do argumento a partir da margem esquerda da janela de texto desde que este valor seja maior que o valor da posição atual do cursor relativamente à margem esquerda.

- 3) Nas linhas 130 e 160 aparecem o símbolo <> que no BASIC do seu TK-2000 significa diferente de (\neq).

Na Tabela 1 temos todos os tipos de operações comparativas, assim como os respectivos símbolos usados pelo TK-2000 COLOR.

Operação	Símbolo
menor que	<
maior que	>
igual	=
diferente de	< >
maior ou igual a	>=
menor ou igual a	<=

Tabela 1

As operações comparativas geralmente aparecem dentro do IF... THEN.

Veremos entretanto que além das operações comparativas são muito importantes dentro do IF-THEN as operações lógicas.

Trabalho para você obnubilado(a) leitor(a)!!!

Elabore um programa que simule o lançamento de 7 (sete) dados simultaneamente, 20 vezes.

Quantas vezes nesse total a soma de pontos nos dados é menor ou igual a 21 pontos?

É um bom exercício, não acha?

A fórmula de Heron

Este é o nosso terceiro programa e inicialmente destacamos a importância do número três na Matemática e em geral.

O prestígio do número três decorre certamente de vários conjuntos notáveis que são trípliques.

Três são as partes do universo: céu, terra e inferno.

Três são as fases da existência humana: nascimento, vida e morte.

Três são as parcelas da eternidade: passado, presente e futuro.

Três são os reinos da natureza: animal, vegetal e mineral.

Três são as partes do corpo humano: cabeça, tronco e membros.

Três são as dimensões do espaço: comprimento, largura e altura.

Três são os elementos principais do mundo material: terra, água e mar (o homem levou muito tempo para descobrir o fogo).

Três são em Matemática, as condições possíveis para a situação de uma variável, ou cresce ou se mantém constante ou decresce.

Três são os pontos que formam o sinal gráfico chamado "reticência".

Três são os lados de um triângulo.

E nada melhor após tanto "três" mostrar como se pode obter a área de um triângulo, qualquer que ele seja quanto aos seus lados e ângulos.

Heron de Alexandria no ano 60 da nossa era desenvolveu uma fórmula através da qual se pode obter a área de um triângulo qualquer em função dos seus lados e em função do seu semi-perímetro ou seja a metade do perímetro.

A fórmula de Heron é

$$S_{\triangle abc} = \sqrt{p(p-a)(p-b)(p-c)} \quad (1)$$

onde

$$p = \frac{a + b + c}{2}$$

Há quem diga que Arquimedes conhecia esta fórmula muitos séculos antes de Heron...

Arquimedes nasceu na cidade de Siracusa no ano 287 a.C. e era descendente da família real.

Embora de época tão remota, pode-se considerar Arquimedes como um moderno em pensamento.

Realmente pode-se equipará-lo ao genial físico e matemático inglês Isaac Newton.

Arquimedes não foi só matemático.

Foi, também, inventor.

Seus inventos eram baseados no que hoje chamamos de máquinas simples (alavancas, roldanas, sarilhos, etc.).

É famosa a sua afirmação "dai-me um ponto de apoio e moverei o mundo".

Bem, com todo esse currículo não é para duvidar que Arquimedes (Figura 10) pudesse conhecer a fórmula (1) antes do próprio Heron.



Figura 10

De qualquer maneira se Arquimedes conhecia ou não a (1) antes de Heron não vai "atrapalhar" o seu esmerado TK-2000 COLOR a extrair a raiz quadrada da (1) após ter efetuado os produtos e indicar o valor correto da área de um triângulo.

Aliás é só em Portugal que o pessoal encheu a "santa terrinha" de buracos e não conseguiu extrair nenhuma raiz quadrada...

Aí vai o fluxograma (Figura 11) resumido do algoritmo:

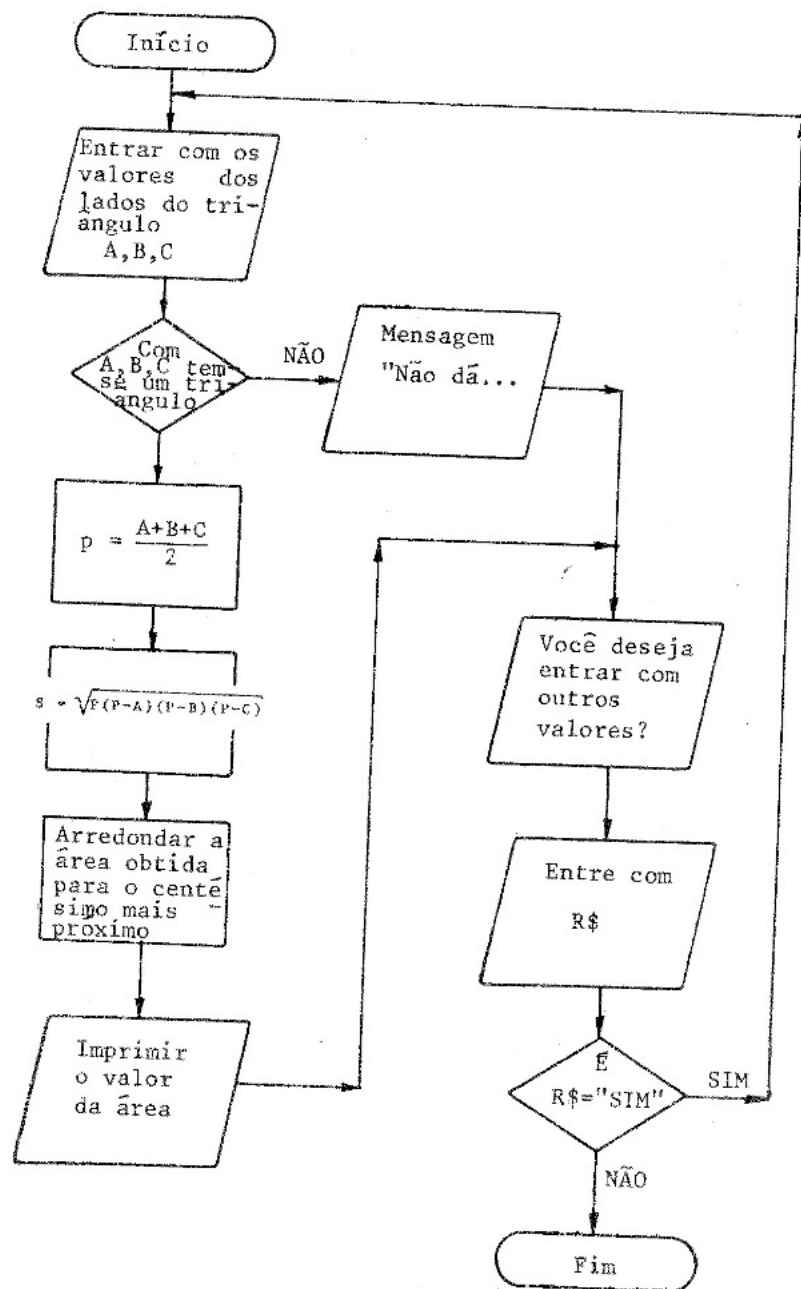


Figura 11

Programa em BASIC

```

10 PRINT "CÁLCULO DA ÁREA USANDO A FÓRMULA DE HERON"
15 FOR I = 1 TO 40
20 PRINT "*";
25 NEXT I
28 PRINT
30 INPUT "ENTRE COM OS COMPRIMENTOS DOS LADOS";A,B,C
40 IF A+B > C AND A+C > B AND (B+C > A) THEN 70
50 PRINT "NÃO DÁ PARA OBTER UM TRIÂNGULO"
60 PRINT: PRINT: GOTO 30
70 P = (A+B+C)/2
80 S = SQR(P*(P-A)*(P-B)*(P-C))
90 S = S + .005
100 S = INT (100 * S)/100
110 PRINT "A ÁREA É:";S
120 FOR I = 1 TO 40
125 PRINT "*";
130 NEXT I
140 PRINT "VOCÊ QUER ENTRAR COM OUTROS DADOS?(SIM/NÃO)"
150 INPUT R$
160 IF R$ = "SIM" THEN 30
170 END
  
```

Agora que você teclou todo o programa apli-
que o mesmo para os seguintes dados:

Caso	A	B	C	Área(S)
1	1	2	3	?
2	7	8	9	?
3	13,1	14,2	15,3	?
4	14,4	10,8	12,2	?
5	255,18	290,87	419,25	?

Tabela 2

As respostas que você deve obter são respectivamente:

- 1) NÃO DA PARA OBTER O TRIÂNGULO
 - 2) 26.83
 - 3) 86.26
 - 4) 64.26
 - 5) 36536.3
- └─ este é o ponto decimal

Comentários:

1) Na linha 40 dentro do IF temos várias operações lógicas usando o operador lógico ou Booleano AND (E).

Os operadores Booleanos dão aos programas a habilidade de fazer decisões lógicas (nem podia ser do outro jeito...) e no caso da linha 40 podemos testar de uma só vez todas as possibilidades para ver se a soma de dois lados ultrapassa um terceiro quando então podemos construir um triângulo.

Ocorrendo essa possibilidade (soma de quais quer dois lados é maior que um terceiro) o programa se desvia para a linha 70, começando aí os cálculos para se chegar a fórmula de Heron. Em caso contrário se vai para a linha 50.

Deve-se notar que após o último AND colocamos $B+C > A$ entre parênteses e isto para evitar a formação do AT que é uma palavra reservada pela junção do A com o THEN.

Experimente escrever sem o parênteses e veja a mensagem de erro que aparece.

Uma alternativa é não usar o THEN e sim o GOTO quando então o uso dos parênteses é desnecessário.

2) Você pode usar três operadores lógicos no seu TK-2000 COLOR ou seja AND(E), OR(OU) e NOT (NÃO).

Apresentamos a seguir as tabelas verdade para esses operadores.

Entrada	Saída
1 AND 1	1
1 AND 0	0
0 AND 1	0
0 AND 0	0

Tabela 3 - AND

Entrada	Saída
1 OR 1	1
1 OR 0	1
0 OR 1	1
0 OR 0	0

Tabela 4 - OR

Entrada	Saída
NOT 1	0
NOT 0	1

Tabela 5 - NOT

Caso existir alguma dificuldade para você entender o operador Booleano pense na seguinte situação: suponha que está num parque de diversões com os seus dois filhos e eles querem sorvete.

O operador AND diz que você vai comprar sorvete se ambos os filhos quiserem. O operador OR diz que você vai comprar sorvete se ao menos um dos filhos quiser.

Já o operador NOT é o da "contrariedade" isto é, se um filho X discordar do Y a sua decisão é o NOT dele.

Os micros não operam com analogia; eles trabalham com números.

No caso da lógica Booleana tem-se os valores 1 para verdadeiro e 0 para falso (veja as Tabelas 3, 4 e 5).

Como os operadores Booleanos trabalham com uns e zeros, eles comumente são usados dentro de expressões relacionais ou seja nas operações compara-

tivas.

Além das operações comparativas sempre resultam em "uns" e "zeros" sendo o 1 (um) para a condição verdadeira e o 0 (zero) para a condição falsa.

Os operadores Booleanos tem igual procedência.

Caso apareça mais de um operador Booleano na mesma expressão (é o caso da linha 40), o cálculo é feito da esquerda para a direita.

3) Na linha 80 aparece a função numérica $SQR()$ que tem por finalidade a extração da raiz quadrada do número que está entre os parênteses.

Pequeno trabalho para você!!!

O raio r de um círculo inscrito em um triângulo é dado pela fórmula

$$r = \sqrt{\frac{(p-a)(p-b)(p-c)}{p}} \quad \longrightarrow \quad (2)$$

onde

$$p = \frac{a + b + c}{2}$$

Faça as convenientes modificações no programa apresentado (Nº 3) para que além da área saia também calculado e impresso o valor do raio do círculo inscrito.

Aplique o programa para os dados da Tabela 6.

a	b	c	S	r
384,13	491,85	675,47	?	?
28,87	60,19	52,18	?	?
5,48	4,83	6,02	?	?

Tabela 6

O teorema de Pitágoras e as triplas pitagóricas

Chegamos ao quarto programa e antes demais nada devemos dizer que o *quatro* é um número pitagórico.

Cinco séculos antes de Cristo viveu na Grécia um filósofo, geômetra e moralista cujo nome fulgura com excepcional destaque na história da Matemática.

Chamava-se Pitágoras e nasceu na ilha de Samos.

Ensinava Pitágoras a seus aplicados e fiéis discípulos que os números governavam o mundo e que, por isso, todos os fenômenos que ocorriam na terra, no ar, no fogo ou na água, podiam ser expressos, avaliados e previstos por meio de números.

Os números, proclamava o mestre, exercem decisiva influência sobre a vida de cada ser.

Existem números felizes e números fatídicos.

As doutrinas do filósofo de Samos, atingiram no meio intelectual de Crotona, naquela época uma pequena colônia grega, um prestígio incalculável.

Para os pitagóricos o número quatro, imagem do sólido, é o primeiro quadrado perfeito par e representa a matéria nos quatro elementos que deveriam integrá-la: o fogo, o ar, a terra e a água.

Os famosos Versos Dourados, atribuídos a Pitágoras, começam por uma exaltação do número quatro: "Salve número famoso, gerador dos deuses e dos homens".

O número quatro, na linguagem mística, lembra a figura de um quadrado, simboliza a extensão limitada do mundo físico, pois segundo a crença dos antigos sacerdotes caldeus, a Terra era plana e tinha a forma perfeita de um quadrilátero regular.

Cada templo deveria ser a miniatura da Terra e a forma quadrangular era, sem dúvida, o que mais agradava aos deuses.

O templo dos maometanos, em Meca - a Caaba - tem a forma exata de um quadrado e os árabes veneram uma pedra negra de forma cúbica, isto é, com todas as faces regulares e quadrangulares.

Os sacerdotes chineses, vinte séculos antes do nascimento de Cristo, já ensinavam que "a terra era representada pela cor amarela e que tinha a forma de um quadrado".

O 4 (quatro) simboliza a criação, o quaternário terrestre da yoga oriental, as 4 Nobres Verdades de Buda, os 4 seres mitológicos do Celeste Império, ou seja, o dragão, o fenix, o unicornio e a tartaruga.

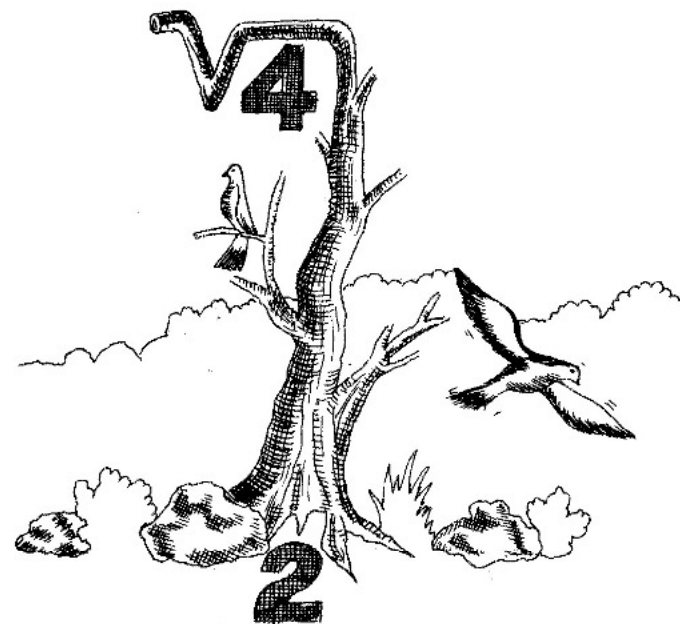


Figura 12

Mas a pergunta que ora fazemos é: o que Pitágoras, Euclides e o presidente dos Estados Unidos da América do Norte James A. Garfield tiveram em comum?

Resposta: Apesar de viverem em épocas totalmente diferentes Pitágoras no VI século a.C., Euclides (autor dos famosos "Elementos de Geometria") cerca de 325 a.C. e Garfield aproximadamente 100 anos atrás conseguiram todos e de forma diferente provar o famoso teorema de Pitágoras.

Só faltava o Pitágoras, o próprio, não ter conseguido provar que: "em um triângulo retângulo, a soma dos quadrados dos comprimentos dos catetos é igual ao quadrado da hipotenusa".

Se construirmos um triângulo e dermos aos lados os comprimentos 3, 4 e 5 cm, respectivamente, o dito triângulo será retângulo; o ângulo reto se

encontra entre os lados 3 e 4.

Além disso, vê-se facilmente (veja a Figura 13) que o quadrado construído sobre o lado maior, a "hipotenusa", é precisamente igual à soma dos quadrados formados sobre os outros dois lados os "catetos" [$9+16 = 25$].

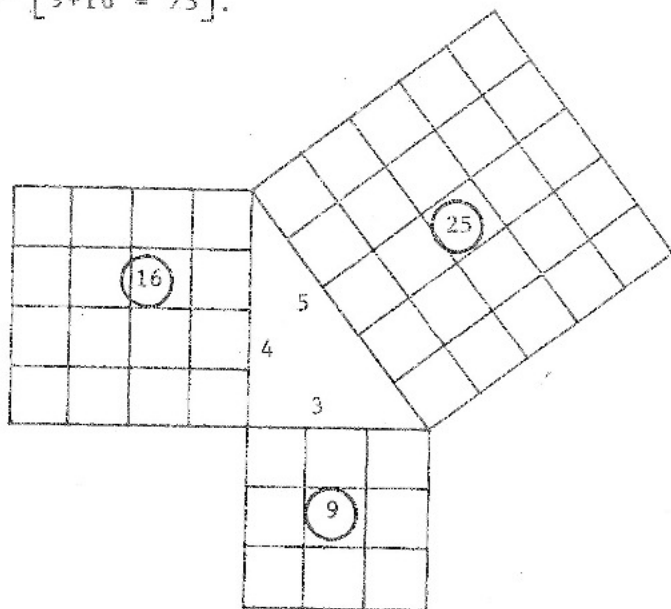


Figura 13

Um caso particular interessante, não é?

É um caso interessante porém não é tão particular pois se construirmos um triângulo de lados 12, 5 e 13 cm, o teorema continuará válido!!!

Quando obrigados a demarcar os campos lodosos, após o retraimento das águas do Nilo, os agricultores egípcios faziam uso prático desta relação e poderia alguém pensar que nisto se resumisse toda a utilidade do teorema.

Pitágoras provou que qualquer que seja o triângulo retângulo ter-se-á sempre:

$$\boxed{a^2 + b^2 = c^2} \longrightarrow (1)$$

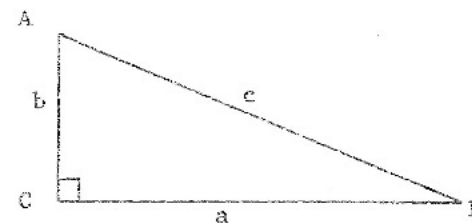


Figura 14 - Triângulo retângulo

Mas aí vai a demonstração da (1) feita pelo presidente Garfield em 1876 quando ainda era apenas um membro do Congresso e tinha um "tempinho" para se dedicar a Matemática.

Dado o triângulo retângulo ABC, constrói-se um trapézio DEBC com lados de comprimento a , b e c e os mesmos ângulos como mostrado na Figura 15.

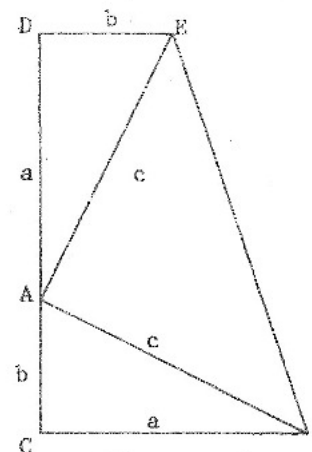


Figura 15

Daí surgem as perguntas:

- 1) Qual é a relação entre os triângulos $\triangle AED$ e $\triangle ABC$?
- 2) Como podemos saber que $AE = c$?
- 3) Qual é o tipo do ângulo $\angle BAE$?

As respostas são:

- 1) Tem ambos lados de mesmo comprimento e ângulos iguais.
- 2) Por construção feita. Só faltava não ser...
- 3) Agudo e de 45° .

A área do trapézio DEBC é a soma das bases vezes a altura dividida por 2.

Assim

$$\text{Área DEBC} = \frac{1}{2} (a+b)(a+b) = \frac{1}{2} (a^2 + 2ab + b^2) \rightarrow (2)$$

$$\begin{aligned} \text{Área DEBC} &= \text{Área EAD} + \text{Área BAE} + \text{Área BCA} \\ &= \frac{1}{2} ab + \frac{1}{2} ab + \frac{1}{2} ab = ab + \frac{c^2}{2} \rightarrow (3) \end{aligned}$$

(usou-se aqui a condição que a área de um triângulo é igual a base vezes altura dividida por 2).

De (2) e (3) vem:

$$(2) = (3) \rightarrow \boxed{c^2 = a^2 + b^2} \rightarrow \text{é a (1)}$$

O primeiro probleminha é obter a hipotenusa de um triângulo retângulo dados os valores dos seus catetos.

a) Análise

Para se obter a hipotenusa usa-se a fórmula

$$c = \sqrt{a^2 + b^2} \rightarrow (4)$$

O programa deve incluir uma verificação pa

ra se estar seguro que os comprimentos dos catetos são positivos.

b) Fluxograma sucinto.

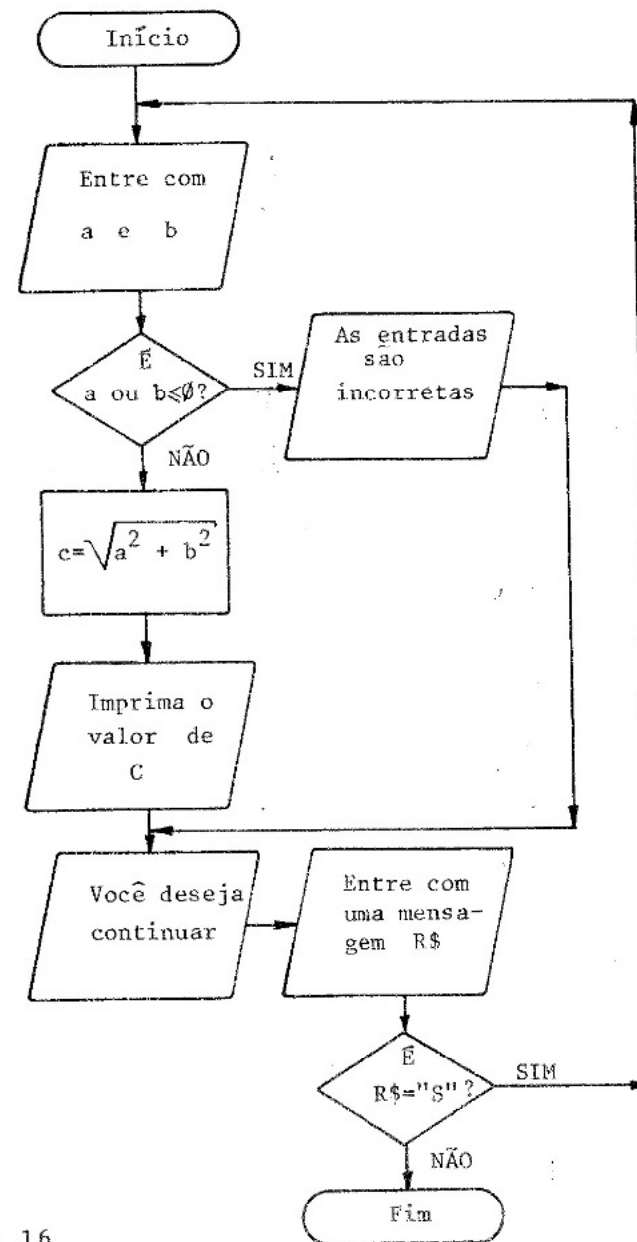


Figura 16

Aí vai o programa em BASIC para o seu TK-2000 COLOR.

```

10 PRINT "O TEOREMA DE PITÁGORAS"
15 PRINT: PRINT
20 INPUT "ENTRE COM OS COMPRIMENTOS DOS CATETOS";A,B
25 IF A <= 0 OR B <= 0 THEN 130
35 C = SQR(A*A + B*B)
50 PRINT "A HIPOTENUSA É: B"; C
60 PRINT: PRINT
65 PRINT "VOCÊ DESEJA CONTINUAR?"
70 PRINT "ENTRE COM S (DE SIM) EM CASO AFIRMATIVO"
80 INPUT R$
90 IF R$ = "S" THEN HOME: GOTO 20
100 PRINT: PRINT
110 PRINT "PITÁGORAS ESTÁ FELIZ"
120 GOTO 150
130 PRINT "OS CATETOS NÃO PODEM SER NEGATIVOS OU NULOS "
140 GOTO 60
150 END

```

Aplique o programa para os seguintes dados:

Caso	a	b	c
1	0	9	?
2	3	4	?
3	4	9	?
4	7	24	?
5	8	15	?

Tabela 7

Você deve obter as seguintes respostas:

- 1) "OS CATETOS NÃO PODEM SER NEGATIVOS OU NULOS";
 2) 5; 3) 9.84885781; 4) 25; 5) 17.

AS TRIPLAS PITAGÓRICAS

Já brincamos um pouco com o teorema de Pitágoras e agora podemos pensar em obter um conjunto de triplas ou trinças de números que satisfazem o citado teorema.

A trinça pitagórica mais popular é (3,4,5).

Para se obter as triplas de uma forma eficiente precisaremos de uma fórmula e aí vai a sequência.

Já sabemos que $c^2 = a^2 + b^2$ [veja a (1)] e obtém-se essa igualdade se tivermos:

$$\begin{cases} a = m^2 - n^2 \\ b = 2mn \\ c = m^2 + n^2 \end{cases} \text{ onde } m > n$$

Assim qual é a tripla que se pode obter se $m = 3$ e $n = 2$?

Resposta:

$$\begin{cases} a = 9 - 4 = 5 \\ b = 2 \cdot 3 \cdot 2 = 12 \\ c = 9 + 4 = 13 \end{cases}$$

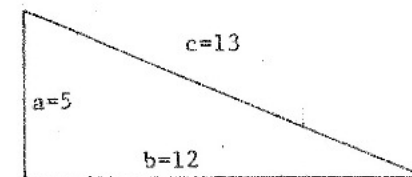


Figura 17

Gostou?!?!?

Vamos então obter todas as triplas pitagóricas com $2 \leq m \leq 7$ e $1 \leq n \leq 6$.

Aí vai o fluxograma resumido do algoritmo que permite resolver o problema.

O.I.- Para simplificar o fluxograma apenas com um símbolo representamos várias operações de atribuição

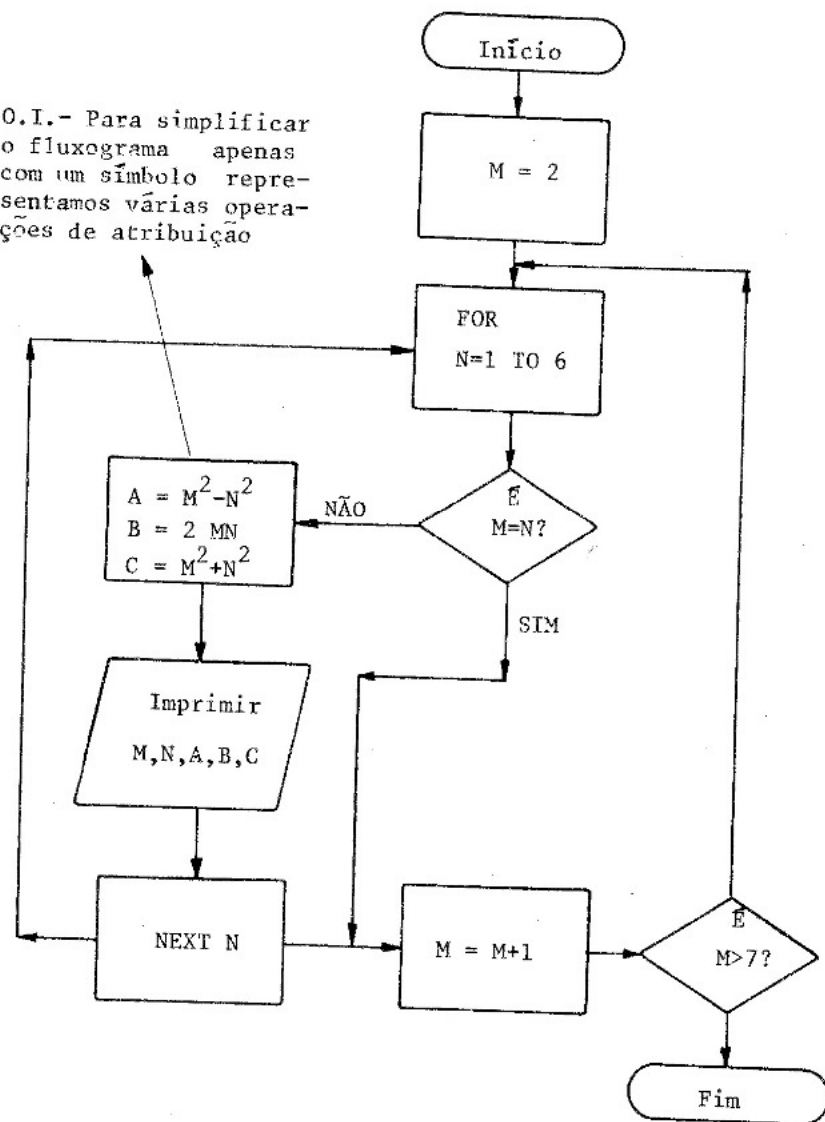


Figura 18

Programa para a obtenção das triplas pitagóricas.

```

10 PRINT "GERAÇÃO DE TRIPLAS PITAGÓRICAS"
20 PRINT "M"; TAB(7); "N"; TAB(16); "A"; TAB(24); "B";
   TAB(30); "C"
25 M = 2
30 FOR N = 1 TO 6
40 IF M = N THEN 100
50 A = M*M - N*N
60 B = 2*M*N
70 C = M*M + N*N
80 PRINT M; TAB(7); N; TAB(16); A; TAB(24); B; TAB(30); C
90 NEXT N
100 M = M + 1
110 IF M > 7 THEN 130
120 GOTO 30
130 END
  
```

Na saída você deve obter algo do tipo

GERAÇÃO DE TRIPLAS PITAGÓRICAS

	M	N	A	B	C
Posição →	1	7	16	24	30
Valores →	2	1	3	4	5
	3	1	8	6	10
	3	2	5	12	13

	7	5	24	70	74
	7	6	13	84	85

Sugestão de trabalho para você:

1) Escreva um programa para gerar triplas pitagóricas usando dois laços FOR-NEXT um embutido dentro do outro (veja por exemplo o programa N^o 14 Caleidoscópio).

2) Escreva um programa para obter as triplas pitagóricas sem usar nada mais do que o teorema de Pitágoras usando dois laços FOR-NEXT, um para o A e outro para o B e de tal forma que $1 \leq A \leq 30$, $1 \leq B \leq 32$ com A e B inteiros.

Cinemática

Aqui vai um programa para auxiliá-lo a dominar um pouco melhor a Cinemática.

Nele resolvem-se problemas que estudam movimentos retilíneos uniformemente variados.

Nesses problemas comumente aparecem cinco variáveis que aqui representaremos por:

$$\left\{ \begin{array}{l} s \rightarrow \text{distância percorrida} \\ v_i \rightarrow \text{velocidade inicial} \\ v_f \rightarrow \text{velocidade final} \\ a \rightarrow \text{aceleração} \\ t \rightarrow \text{tempo} \end{array} \right.$$

Neste programa serão usadas as seguintes fórmulas:

$$\left\{ \begin{array}{l} s = v_i t + \frac{1}{2} a t^2 \quad \longrightarrow (1) \\ v_f = v_i + a t \quad \longrightarrow (2) \\ a = \frac{2(v_f t - s)}{t^2} \quad \longrightarrow (3) \\ t = \frac{v_f - v_i}{a} \quad \longrightarrow (4) \\ v_i = \sqrt{v_f^2 - 2as} \quad \longrightarrow (5) \end{array} \right.$$

O.I. - O primeiro livro que escrevemos, com o auxílio da então nossa namorada e hoje a esposa de tantos anos felizes Nilza Maria, foi Cinemática cuja capa está aí na Figura 19.

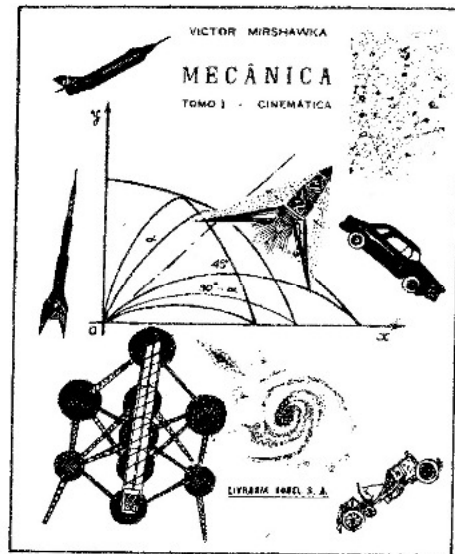


Figura 19

Neste livro, como em qualquer outro de Cinemática o(a) curioso(a) leitor(a) encontrará a justificativa das fórmulas (1), (2), (3), (4) e (5).

Bem vamos ao programa para o seu TK-2000 COLOR que por sinal, apesar de muito útil, é bem curto.

```

10 PRINT "CÁLCULOS CINEMÁTICOS"
20 PRINT: PRINT
30 S$ = "DISTÂNCIA"
40 VI$ = "VELOCIDADE INICIAL"
50 VF$ = "VELOCIDADE FINAL"
60 A$ = "ACELERAÇÃO"
70 T$ = "TEMPO"
75 PRINT "NÃO ESQUEÇA DE ENTRAR COM 0(ZERO) PARA A VARIÁ-
    VEL INCÓGNITA"
78 PRINT: PRINT
80 INPUT S, VI, VF, A, T
90 PRINT S$; "00"; S
100 PRINT VI$; "00"; VI
110 PRINT VF$; "00"; VF
120 PRINT A$; "00"; A
130 PRINT T$; "00"; T
135 PRINT: PRINT
140 PRINT "O VALOR DA INCÓGNITA PROCURADA É:"
150 FOR J = 1 TO 40
155 PRINT "?";
160 NEXT J
165 PRINT "AÍ VAI O VALOR DA INCÓGNITA QUE VOCÊ DESEJA"
170 PRINT: PRINT
180 IF S = 0 THEN PRINT S$; "0000"; VI*T+.5*A*T*T
190 IF VI = 0 THEN PRINT VI$; "0000"; SQR(VF*VF-2*A*S)
200 IF VF = 0 THEN PRINT VF$; "0000"; VI + A * T
210 IF A = 0 THEN PRINT A$; "0000"; 2*(VF*T-S)/(T*T)
220 IF T = 0 THEN PRINT T$; "0000"; (VF-VI)/A
230 FOR J = 1 TO 40
240 PRINT "#";
250 NEXT J
260 PRINT: PRINT
270 PRINT "VOCÊ QUER EXECUTAR UM NOVO CÁLCULO?(SIM/NÃO)"

```

```

280 INPUT R$
290 IF R$ = "SIM" THEN HOME: GOTO 75
300 END

```

Como aplicação desse programa obtenha os valores que estão com o ponto de interrogação (?) na Tabela 8.

S	VI	VF	A	T
?	100	<input type="checkbox"/>	10	10
300	?	180	2	<input type="checkbox"/>
<input type="checkbox"/>	20	?	30	185
5000	<input type="checkbox"/>	105	?	50
<input type="checkbox"/>	30	150	7	?

Tabela 8

O.I. 1 - Não esqueça que na linha 80 você deverá entrar com S, VI, VF, A e T.

A variável cujo valor você quer (casela com ?) deve receber o valor zero e a variável não necessária no cálculo (casela com) deve receber o valor *codificado* -1.

O.I. 2 - Para se utilizar esse programa, você carro(a) leitor(a) precisa conhecer ou informar todas as variáveis menos uma.

A variável não conhecida será representada pelo zero e o programa escolherá a equação apropriada para imprimir a "solução" para a sua dúvida.

Todas as variáveis precisam estar nas mesmas unidades, assim por exemplo se você "informar" a distância em metros (m), a velocidade em metros/segundo (m/s) o programa lhe dará a aceleração em metros por segundo ao quadrado (m/s^2).

O.I. 3 - Usamos no programa as seguintes constantes alfanuméricas: "DISTÂNCIA", "VELOCIDADE INICIAL", "VELOCIDADE FINAL", "ACELERAÇÃO", "TEMPO" as quais foram "guardadas" nas seguintes denominações respectivamente S\$, VI\$, VF\$, A\$, T\$.

Teremos a impressão de S\$ quando estivermos fazendo o cálculo da distância percorrida, de VI\$ se quisermos a velocidade inicial, etc.

O.I. 4 - Com as variáveis S, VI, VF, A e T representamos respectivamente a distância percorrida, a velocidade inicial, a velocidade final, a aceleração e o tempo gasto no percurso.

TRABALHO PARA VOCÊ perplexo(a) leitor(a)

I) Elabore um programa que estude a relação matemática entre tempo (t), distância (s) e velocidade (v) em um movimento uniforme.

É evidente que você sabe que $s = vt$, não é ?

Nesse programa você deve entrar com o tempo e a distância e ter o seguinte tipo de "conversa" com o seu TK-2000 COLOR:

a) Escolhe um valor para o tempo entre as seguintes unidades,

- $$\left\{ \begin{array}{l} 1 - \text{segundos} \\ 2 - \text{minutos} \\ 3 - \text{horas} \\ 4 - \text{dias} \\ 5 - \text{meses} \end{array} \right.$$

e o seu TK-20000 COLOR converte o mesmo para uma outra unidade.

Por exemplo você entra com o tempo em dias (4) e quer a conversão para minutos (2).

- b) Escolhe um valor para a distância entre as seguintes unidades,

- $$\left\{ \begin{array}{l} 1 - \text{milímetros} \\ 2 - \text{centímetros} \\ 3 - \text{metros} \\ 4 - \text{quilômetros} \\ 5 - \text{pés} \\ 6 - \text{milhas} \end{array} \right.$$

e o seu TK-20000 COLOR converte a mesma para uma outra unidade.

Por exemplo você entra com a distância em quilômetros (4) e quer a conversão para milhas (6).

- c) Escolha um dos códigos que permita calcular a terceira variável conhecida as outras duas.

As possibilidades são:

- $$\left\{ \begin{array}{l} 1 - \text{Dados o tempo e a distância achar a velocidade} \\ 2 - \text{Dados o tempo e a velocidade achar a distância} \\ 3 - \text{Dadas a velocidade e a distância achar o tempo} \end{array} \right.$$

Se por exemplo você entra com o código 1 devem surgir antes as perguntas sobre qual é a unidade que você deseja para o tempo e para a distância respectivamente.

Você deve obter a velocidade nas unidades escolhidas e ter ainda a possibilidade final de poder expressá-la em outra forma.

Caso você entre com o tempo em horas tendo escolhido o valor 3 e a distância em milhas tendo escolhido o valor 250 é evidente que para a velocidade deve-se obter o valor 83,3333334 milhas por hora.

O seu programa deve possibilitar a transformação desse resultado, digamos para metros por segundo quando então para os dados escolhidos acima deve-se obter 37,2523148 metros por segundo.

O.I. - Use o código 0 (zero) para poder parar automaticamente a "auto-conversação" com o seu TK-20000 COLOR quando deve sair uma mensagem dando-lhe "os parabéns e desejando um bom descanso"...

II) Elabore um programa para o cálculo das características físicas de um movimento cuja equação horária é

$$s = \frac{1}{2} at^2$$

onde s é a distância, t é o tempo e a é a aceleração.

Perceba que é um caso particular do nosso programa e portanto você imediatamente pode notar que uma possível expressão para a velocidade (v) é

$$v = \sqrt{2as}$$

visto que nesse caso a velocidade inicial é nula.

O programa deve permitir a solução de cada um dos seguintes problemas:

- 1 - dados a e t obter v e s
- 2 - dados a e s obter v e t
- 3 - dados a e v obter s e t
- 4 - dados v e s obter a e t
- 5 - dados v e t obter a e s
- 6 - dados s e t obter a e v

Use como unidade para a distância o metro, para o tempo, o segundo e portanto a velocidade (v) será expressa em metros por segundo (m/s) e a aceleração em metros por segundo por segundo (m/s^2).

No fim de cada resposta obtida o seu TK-2000 COLOR deve lhe fazer a clássica pergunta se você deseja ou não fazer mais cálculos.

Calorimetria

Quando dois corpos A e B, de temperaturas θ_A e θ_B ($\theta_A > \theta_B$) são postos em contato térmico, passados alguns instantes eles entram em equilíbrio térmico.

A temperatura de A diminui e a de B aumenta.

A energia térmica de A diminui e a de B aumenta.

Percebe-se daí que há uma transferência de energia de A para B.

A esta energia, enquanto está em trânsito, de um corpo para outro, dá-se o nome de calor.

Portanto, calor é energia em trânsito.

A Calorimetria classifica o calor como sendo tudo aquilo que se transfere de um corpo para outro, quando entre eles só há diferença de temperatura (noção calorimétrica de calor).

Observe-se que a única grandeza que rege o sentido da transferência de energia entre dois corpos sob a forma de calor, é a *temperatura*, com isso a energia se transfere do corpo de temperatura mais alta (A) para o de temperatura mais baixa (B).

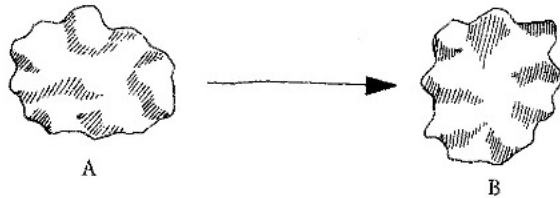


Figura 20

Se não houver diferença de temperatura não há transferência de energia sob a forma de calor.

Verifica-se esta identidade entre calor e energia no fato de podermos aquecer as mãos ou esfregando uma contra a outra (aquecimento pela realização de um trabalho) ou colocando-as perto de uma lareira (aquecimento pelo fornecimento de calor).

Torna-se clara assim uma perfeita analogia entre os conceitos de *trabalho e calor*.

Trabalho é uma modalidade de energia em trânsito que tem como causa fundamental o *exercício de forças entre os corpos*.

Calor é uma modalidade de energia em trânsito, que tem como causa fundamental unicamente a *diferença de temperatura entre os corpos*.

Diagramaticamente

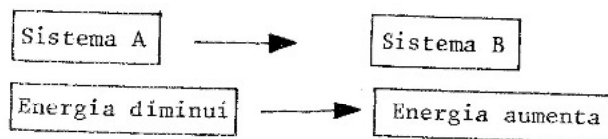


Figura 21 - Trabalho e/ou calor transferem energia de A para B.

É interessante fazermos aqui uma distinção bem nítida entre as duas modalidades de energia:

- a) energia localizada
- b) energia em trânsito

A energia localizada apresenta ainda duas modalidades.

1) *Energia externa*: é a parcela de energia que o sistema possui graças ao seu estado em relação ao ambiente externo. A sua *posição* caracteriza a *energia potencial* e a sua *velocidade* caracteriza a *energia cinética*.

2) *Energia interna* é a parcela de energia que o corpo possui em virtude do estado das partículas que o compõem, consideradas umas em relação às outras, independentes do ambiente externo. A energia térmica, a energia nuclear, a energia química, a energia de deformação elástica, etc., são exemplos de energia interna.

A energia em trânsito é aquela que só existe no ato de transferir-se de um sistema para outro. Além de calor e trabalho também as ondas eletromagnéticas constituem energia em trânsito.

Observe-se finalmente, que não tem sentido falar-se em calor contido em um corpo como não teria sentido falar-se em trabalho contido em um corpo.

Já falamos sobre as unidades de calor porém não é ruim relembrar que:

a) *Pequena caloria* ou *caloria (cal)* é a quantidade de calor necessária para aquecer 1 g de

água pura sob pressão normal, de 14,5 a 15,5°C.

b) *Grande caloria* ou *quilocaloria* (kcal) é a quantidade de calor necessária para aquecer 1 kg de água pura sob pressão normal de 14,5 a 15,5°C.

c) Chama-se B.T.U. (British Thermal Unit) a quantidade de calor necessária para aquecer 1 libra (massa) de água pura sob pressão normal de 64 a 65°F.

É evidente que:

$$\boxed{1 \text{ kcal} = 1\,000 \text{ cal}} \longrightarrow (1)$$

Como 1 libra = 453 g e $(65 - 64)^{\circ}\text{F} \approx \frac{5}{9}^{\circ}\text{C}$,

tiramos:

$$\boxed{1 \text{ BTU} = (453 \times \frac{5}{9}) \text{ cal} \approx 252 \text{ cal}} \longrightarrow (2)$$

Princípios fundamentais da Calorimetria

a) Princípio de transformação inversa

"Se na transformação sofrida por um corpo de um estado (1) a um estado (2) foi necessário fornecer, (ou retirar) uma quantidade de calor Q , na transformação inversa de (2) para (1) será necessário retirar (ou fornecer) a mesma quantidade de calor Q ".

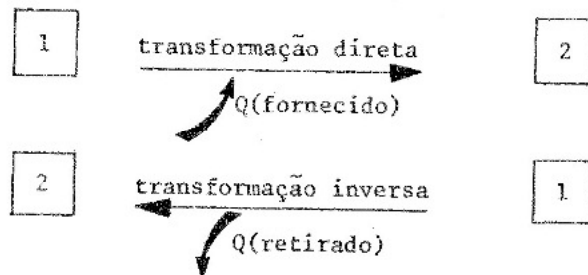


Figura 22

b) Princípio da igualdade das trocas de calor

"Quando um sistema termicamente isolado, sofre transformações devido somente às diferenças de temperatura entre os corpos que o constituem, a soma das quantidades de calor cedidas por uns é igual a soma das quantidades de calor recebidas pelos outros".

Podemos então enunciar que:

"Quando um sistema termicamente isolado sofre transformações devido apenas as diferenças de temperatura entre os corpos que o constituem, a soma algébrica das quantidades de calor trocadas é igual a zero".

$$\Sigma Q = 0 \longrightarrow (3)$$

Para massas iguais a experiência mostra que a variação da temperatura depende da natureza do corpo.

Para caracterizar esse aspecto térmico do fenômeno usa-se o calor específico (c).

Quando dois corpos com massas iguais mas com calores específicos diferentes são postos um em presença do outro, o de maior calor específico sofre menor variação de temperatura.

O calor específico, a experiência mostra, é variável com a temperatura. Entretanto, para pequenos intervalos, pode-se considerá-lo como constante. Por esse motivo, a quantidade de calor em jogo num fenômeno térmico, quando calculada pela fórmula

$$\boxed{Q = mc \Delta \theta} \longrightarrow (4)$$

resulta num valor aproximado. Nessa expressão, o calor específico é considerado constante no intervalo de temperatura (é o calor específico médio). Um melhor resultado seria conseguido considerando-se o calor específico numa dada temperatura.

A unidade geralmente usada para o calor específico é:

$$\left\{ \frac{\text{cal}}{\text{g}^{\circ}\text{C}} \right\}$$

ou

$$\left\{ \frac{\text{kcal}}{\text{kg}^{\circ}\text{C}} \right\}$$

ou ainda

$$\left\{ \frac{\text{j}}{\text{kg}^{\circ}\text{C}} \right\}$$

Calor específico da água

Conforme a definição de caloria, a quantidade de calor ΔQ necessária para aquecer 1g de água pura de 14,5 a 15,5°C é 1 cal. Logo:

$$\left\{ \begin{array}{l} \Delta Q = 1 \text{ cal} \\ m = 1 \text{ g} \\ \Delta\theta = 15,5 - 14,5 = 1^{\circ}\text{C} \end{array} \right\} \rightarrow c_{\text{H}_2\text{O}} = 1 \frac{\text{cal}}{\text{g}^{\circ}\text{C}} \rightarrow (5)$$

Interpretação física do calor específico em vista da (4).

"Calor específico de uma substância é numericamente igual à quantidade de calor que transferida à uma quantidade de massa unitária da substância, nela determina uma elevação unitária de temperatura".

Geralmente o calor específico de uma substância depende não só da temperatura como do estado de agregação. Nos corpos cristalinos, por exemplo, tem-se para cada pressão, uma temperatura de fusão bem determinada.

Na Tabela 9 temos o calor específico médio para algumas substâncias entre 0°C e 100°C.

Substância	c { $\frac{\text{cal}}{\text{g}^{\circ}\text{C}}$ }
Ferro (aço)	0,115
Alumínio	0,212
Latão	0,092
Cobre	0,093
Mercúrio	0,033
Pinho (madeira)	0,650
Mica	0,208
Ácido sulfúrico	0,330
Óleo lubrificante	0,400
Ouro	0,031
Prata	0,056

Tabela 9

Vamos elaborar um programa de tal forma que o seu TK-2000 COLOR escolha aleatoriamente uma das substâncias da Tabela 9 e aqueça a mesma desde 0 até uma temperatura aleatoriamente escolhida de 10 até 50.

A massa da substância também é aleatoriamente escolhida pelo TK-2000 COLOR no intervalo de 100 até 200 gramas.

Devem ser apresentados 15 cálculos diferentes.

Aí vai o programa

```

5 PRINT : PRINT : PRINT : PRINT
10 PRINT "QUANTIDADE DE CALOR RECEBIDA"
15 DIM N$(11), C(11)
20 FOR J = 1 TO 11
25 READ N$(J), C(J)
30 NEXT J
40 DATA "FERRO", 0.115, "ALUMÍNIO", 0.212, "LATÃO", 0.092,
"COBRE", 0.093, "MERCÚRIO", 0.033
50 DATA "PINHO", 0.650, "MICA", 0.208, "ÁCIDO SULFÚRICO",
0.330, "ÓLEO LUBRIFICANTE", 0.400
60 DATA "OURO", 0.031, "PRATA", 0.056
65 FOR I = 1 TO 15
70 T = INT (RND(1)*41) + 10
80 M = INT (RND(1)*101) + 100
90 K = INT (RND(1)*11) + 1
100 Q = M*C(K)*T
105 PRINT "A MASSA É:";M;" E A TEMPERATURA É:";T
110 PRINT "O(A) É:"; N$(K); " E RECEBEU:";
INT(Q*10 + .5)/10; " CALORIAS"
120 PRINT : PRINT
130 NEXT I
140 PRINT : PRINT
150 PRINT "TAREFA CUMPRIDA !!!!"
160 END

```

Trabalho para você respeitado(a) leitor(a)

Dã-se o nome de *calorímetro* a qualquer dispositivo que se presta para medir as trocas de calor entre corpos, bem como as conversões de energia sob as diversas formas em calor ou vice-versa.

Os calorímetros são feitos de forma a utilizar todo o calor sem deixar perder parte apreciável por condutibilidade ou irradiação.

Um calorímetro consta geralmente de dois vasos, colocados um dentro do outro.

Serve de isolante a camada de ar que existe entre as paredes, que são prateadas ou polidas, para diminuir as perdas por irradiação.

O calorímetro assenta sobre material isolante, como amianto ou cortiça, para diminuir as perdas por condutibilidade.

A tampa do calorímetro vem munida de duas aberturas: uma deixa passar a haste de um termómetro sensível; outra, o cabo de um agitador destinado a repartir uniformemente a temperatura na massa de água.

As garrafas térmicas são um bom exemplo prático de calorímetro e elas tem um envólucro duplo entre cujas paredes polidas ou prateadas se fez o vácuo.

Com o calorímetro, determina-se o calor específico, o calor absorvido ou desprendido nas mudanças de estado (calor latente, etc) o calor absorvido ou desprendido nas reações químicas (calor de combustão).

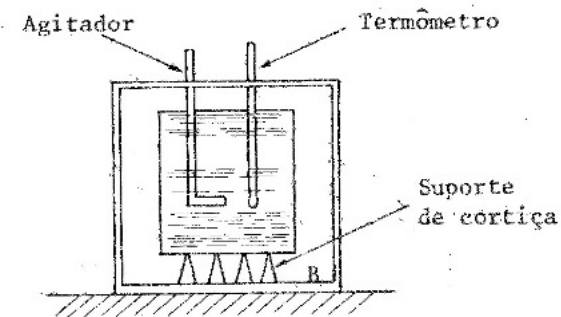


Figura 23

Suponha que você possui um calorímetro elétrico e no líquido contido no vaso calorimétrico coloca-se uma espiral metálica ligada de forma conveniente a um gerador elétrico, um medidor de corrente e a um voltímetro (veja a Figura 24).

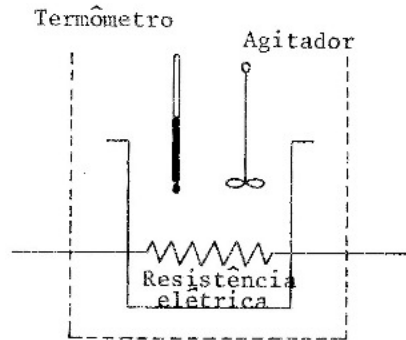


Figura 24

A corrente elétrica aquece a espiral determinando a libertação de uma quantidade de calor Q que é dada pela equação

$$Q = \frac{1}{J} V \cdot i \cdot t \quad (6)$$

onde

$$\begin{cases} J - \text{equivalente mecânico do calor } (J=4,18 \frac{\text{kcal}}{\text{s}}) \\ V - \text{tensão elétrica aplicada à espiral} \\ i - \text{intensidade da corrente na espiral} \\ t - \text{tempo durante o qual passa a corrente.} \end{cases}$$

Elabore um programa que permita obter a quantidade de calor transferida para a água, que supõe-se estar inicialmente a 20°C , para as seguintes condições:

$$\begin{cases} 100 \leq V \leq 110 & \text{em volts} \\ 5 \leq i \leq 10 & \text{em amperes} \\ 30 \leq t \leq 60 & \text{em segundos} \end{cases}$$

Ensine o seu TK-2000 COLOR escolher aleatoriamente um valor inteiro para cada uma das variáveis V , i , t nos intervalos indicados e entrar com os mesmos na (6) com o intuito de obter o valor de Q com uma casa decimal.

Faça com que o TK-2000 COLOR apresente 13 cálculos diferentes desse tipo.

0. Izão

Não é nosso interesse nesse livro dar grandes detalhes sobre conceitos físicos.

Nesse exemplo exageramos um pouco...

A finalidade maior do livro é apresentar o programa como aliás acontecerá na maior parte dos programas-exemplos e não se demorar, com as minúcias e detalhes do problema propriamente dito.

Para os programas de Física estamos elaborando o "TK-2000 na Física".

Quem sabe, quando você estiver lendo isto ele já esteja pronto e se você gostar de Física nada melhor que aprende-la com o auxílio do TK-2000 COLOR.

Adquira pois, neste caso, o "TK-2000 na Física"...

Resistências em séries e paralelo

Nesse programa vamos calcular tanto a resistência em série como em paralelo para até 100 resistores.

Você ainda lembra que no caso da resistência em série tem-se:

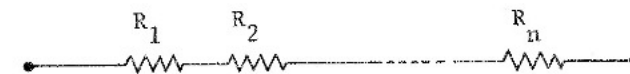


Figura 25

$$R_T = R_1 + R_2 + \dots + R_n = \sum_{i=1}^n R_i \quad \longrightarrow \quad (1)$$

Já para o caso de termos resistores em paralelo tem-se:

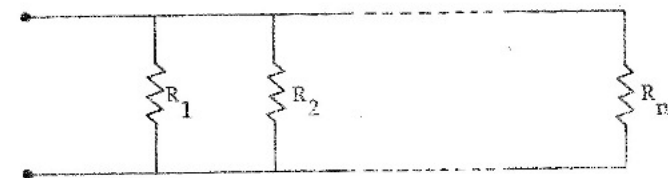


Figura 26

$$R_T = \frac{1}{\frac{1}{R_1} + \frac{1}{R_2} + \dots + \frac{1}{R_n}} \quad \longrightarrow \quad (2)$$

O fluxograma (Figura 27) explica claramente os passos que devem ser seguidos para se obter o valor da resistência equivalente em cada caso.

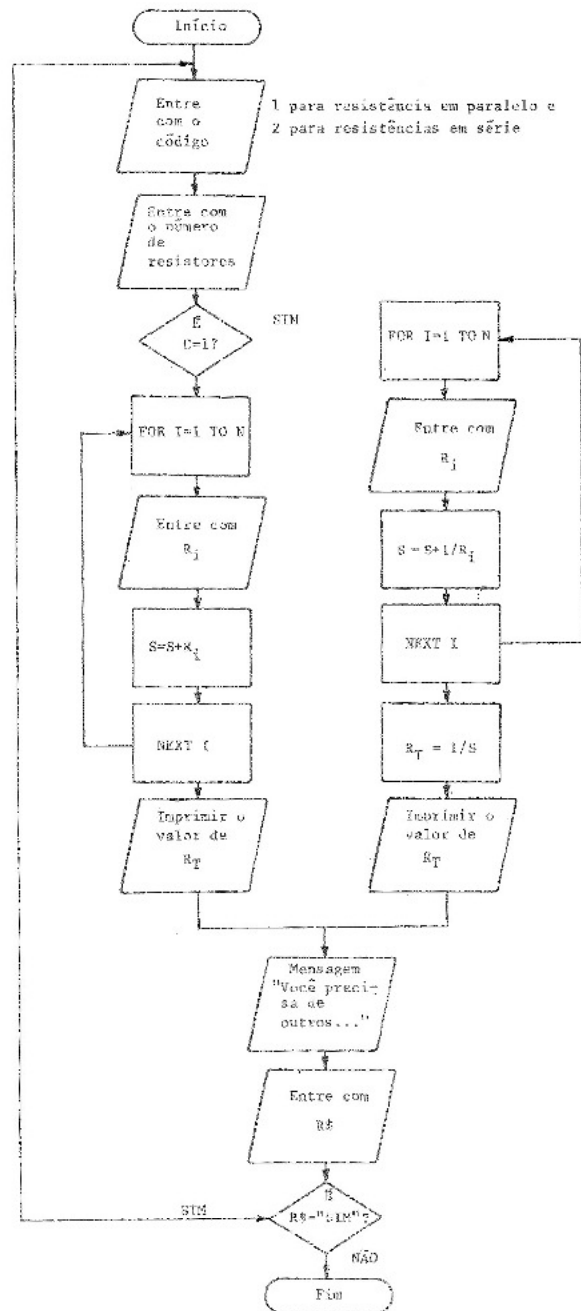


Figura 27

Aí vai o programa que o seu TK-2000 COLOR entende:

```

10 REM ESSE PROGRAMA CALCULA TANTO A RESISTÊNCIA DE UM
    CIRCUITO EM SÉRIE COMO DE UM CIRCUITO EM PARALELO
20 DIM R(100)
30 PRINT "TECLE O NÚMERO 1 PARA O CIRCUITO EM PARALELO E
    2 PARA O CIRCUITO EM SÉRIE"
40 PRINT
50 INPUT C
60 PRINT
70 PRINT "ENTRE COM O NÚMERO DE RESISTORES
    (NO MÁXIMO ATÉ 100)"
75 INPUT N
80 PRINT
90 IF C = 1 THEN 160
100 FOR I = 1 TO N
110 INPUT R(I)
120 S = S + R(I)
130 NEXT I
135 PRINT
140 PRINT "A RESISTÊNCIA TOTAL DO CIRCUITO EM SÉRIE É:
    "; S; "OHMS"
145 PRINT
150 GOTO 230
160 FOR I = 1 TO N
170 INPUT R(I)
180 S = S + 1/R(I)
190 NEXT I
200 RT = 1/S
210 PRINT "A RESISTÊNCIA TOTAL DO CIRCUITO EM PARALELO
    É: "; RT; "OHMS"
220 PRINT
230 PRINT "VOCÊ QUER REALIZAR NOVOS CÁLCULOS?(SIM/NÃO)"
  
```

```

240 INPUT R$
250 IF R$ = "SIM" THEN 270
260 END
270 CLEAR
280 HOME
290 GOTO 30

```

Aplique o seu programa (agora você já tomou conta e portanto o programa é seu) para:

a) *Circuito em série*

$$R_1 = 7\Omega, R_2 = 8\Omega, R_3 = 13\Omega, R_4 = 21\Omega, R_5 = 33.33\Omega$$

b) *Circuito em paralelo*

$$R_1 = 3\Omega, R_2 = 4\Omega, R_3 = 5\Omega, R_4 = 6\Omega, R_5 = 20\Omega, R_6 = 28\Omega$$

Você não esqueceu ainda que Ω é o símbolo de ohm, não é?

Pequenos problemas ou seja tarefas intelectuais para você estimado(a) leitor(a).

- 1) Modifique o programa apresentado para que ao mesmo tempo se possa obter a resistência total de um circuito em série e de um em paralelo com os mesmos resistores sem a necessidade de entrar novamente com o valor das suas resistências.
- 2) Elabore um programa que permita obter a capacitância total de capacitores em série e em paralelo.

Sugestão: Consulte algum livro de Eletrostática para ter a fórmula que dá a capacitância total em cada caso.

Lei de OHM

É muito interessante dominar rapidamente a relação e a interação entre voltagem, corrente, resistência e potência em um circuito elétrico de corrente contínua.

Acreditamos que quase todos(as) os(as) leitores(leitoras) tem ao menos uma vaga idéia da importante lei de Ohm.

Caso tenham esquecido isto, lembrem que a expressão matemática da lei de Ohm é:

$$\text{Voltagem} = \text{Intensidade de corrente} \times \text{Resistência}$$

$$\boxed{V = I \cdot R} \quad (1)$$

Por seu turno obtém-se a potência (P) através de uma das fórmulas

$$\boxed{P = I^2 R = VI = \frac{V^2}{R}} \quad (2)$$

Para que você entenda melhor o programa que vem a seguir apresentamos antes o fluxograma (Figura 28) que dá uma clara idéia do que se quer obter do TK-2000 COLOR.

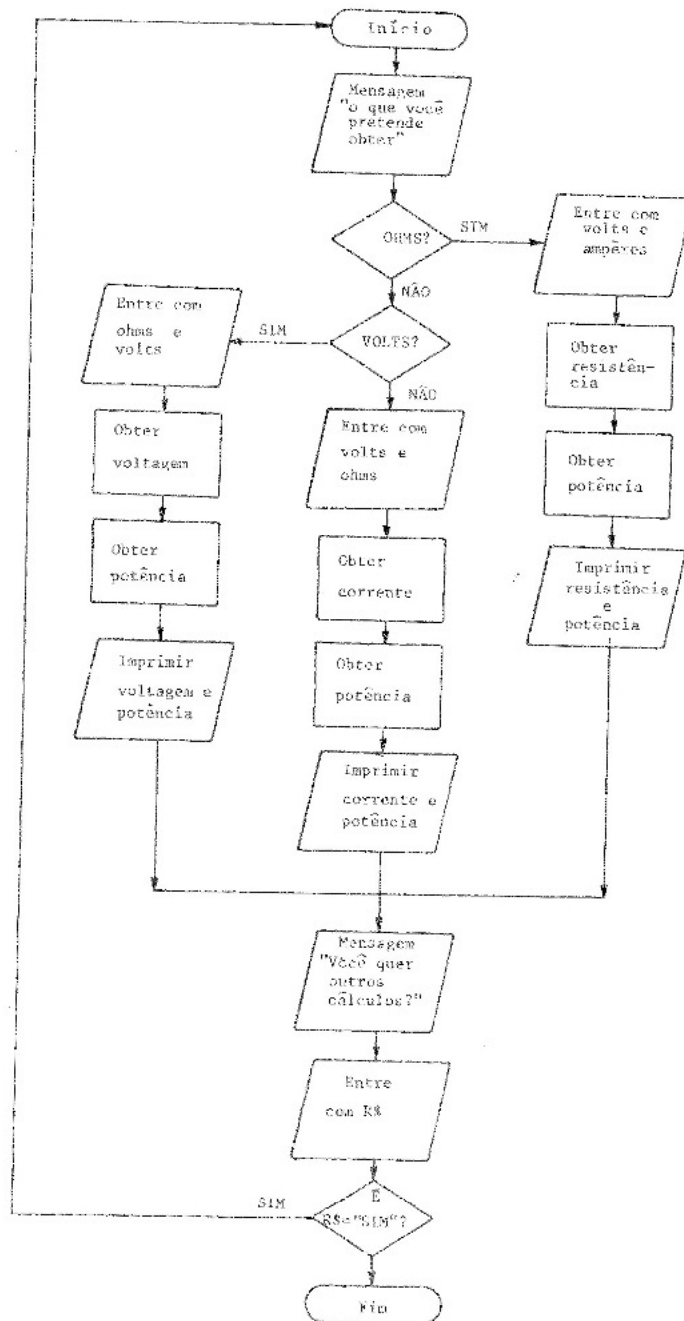


Figura 28

Aí vai o programa em BASIC:

```

10 HOME
20 PRINT "VOCÊ QUER OBTER OHMS (1), VOLTS(2) OU INTENSIDADE
  DE CORRENTE (3)?"
30 PRINT "ENTRE, POR FAVOR, COM O CÓDIGO CORRETO VIU..."
40 INPUT C
50 IF C = 1 THEN 140
60 IF C = 2 THEN 210
70 PRINT "ENTRE COM VOLTS (V) E OHMS (R)"
80 INPUT V, R
90 I = V/R
100 P = V*V/R
110 PRINT "A CORRENTE É: "; INT(I*100 + .5)/100; "AMPÈRES"
120 PRINT "A POTÊNCIA É: "; INT(P*100 + .5)/100; "WATTS"
130 GOTO 400
140 PRINT "ENTRE COM VOLTS(V) E AMPÈRES (I)"
150 INPUT V, I
160 R = V/I
170 P = V*I
180 PRINT "A RESISTÊNCIA É: "; INT(R*100 + .5)/100; "OHMS"
190 PRINT "A POTÊNCIA É: "; INT(P*100 + .5)/100; "WATTS"
200 GOTO 400
210 PRINT "ENTRE COM OHMS(R) E AMPÈRES(I)"
220 INPUT R, I
230 V = R*I
240 P = I * I * R
250 PRINT "A VOLTAGEM É: "; INT(V*100 + .5)/100; "VOLTS"
260 PRINT "A POTÊNCIA É: "; INT(P*100 + .5)/100; "WATTS"
400 PRINT : PRINT
420 PRINT "VOCÊ QUER FAZER NOVOS CÁLCULOS (SIM/NÃO)?"
430 INPUT R$
440 IF R$ = "SIM" THEN GOTO 10
450 END
  
```

O.I. - É evidente nessa altura dos acontecimentos, que você atento(a) leitor(leitora), já está em condições de sugerir uma série de alterações para melhorar esse programa, tanto na apresentação, minimizando o tempo de digitação, assim como obter outras informações sem "estragar" todo o programa.

Assim, é evidente que no lugar da instrução da linha 190 poderíamos ter escrito simplesmente

190 GOTO 120

O mesmo poderia ter sido feito na linha 260 na qual a instrução mais simples para teclar é

260 GOTO 120

Pensando um pouco mais poderíamos perceber que não há necessidade de fórmulas diferentes para a potência e poderíamos nos desviar para uma subrotina que nos calcula a potência após ter sido obtida a terceira variável a partir da lei de Ohm.

Bem é evidente que você já começou a notar essas possibilidades, não é?

Como um pequeno problema para você sugerimos que faça as seguintes modificações no programa "LEI DE OHM".

a) Se a corrente for pequena expresse a mesma em miliamperes.

Aliás se você não fizer essa mudança do jeito que o programa está você obterá o valor zero.

b) Se a resistência ou a potência forem grandes represente as mesmas respectivamente em quilo-ohms e quilowatts.

Fácil, não é?

c) Feitas as modificações preencha a Tabela 10.

Resistência (R) em ohms	Intensidade de corrente (I) em amperes	Voltagem (V) em volts	Potência (P) em watts
10	20	?	?
133	0,001	?	?
213	?	110	?
2130	?	55	?
?	0,00017	220	?
?	88,76	480	?
6666	?	330	?

Tabela 10

Desenhando com o PRINT

Com o uso direto e um pouco cansativo da instrução PRINT podemos fazer alguns desenhos.

Aí vão alguns exemplos

a) Vamos desenvolver a seguinte bicicleta

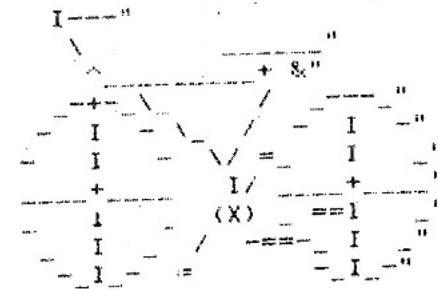


Figura 29

O programa é:

```

5 HOME
10 PRINT " I----"
20 PRINT " \
30 PRINT " ^ + &"
40 PRINT " -+--\ /-----"
50 PRINT " - I - \ / - I -"
60 PRINT " - I - \ / = I -"
70 PRINT " + I / + "
80 PRINT " I (X) ==I "
90 PRINT " I / ---- I "
100 PRINT " - I - = - I -"

```

O.I. Cuidado com os espaços em branco para obter essa bicicleta "super-sprint".

b) Para lembrar os velhos saudosos tempos de primário aĩ vai uma casa.

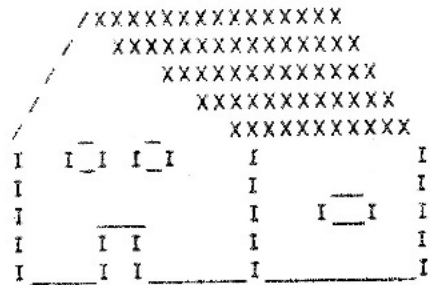


Figura 30

Aĩ vai o programa:

```

5 HOME
10 PRINT " /XXXXXXXXXXXXXXXXXX"
20 PRINT " / XXXXXXXXXXXXXXXXXXXX"
30 PRINT " / XXXXXXXXXXXXXXXXXXXX"
40 PRINT " / XXXXXXXXXXXXXXXXXXXX"
50 PRINT " / XXXXXXXXXXXXXXXXXXXX"
60 PRINT "I I_I I_I I I"
70 PRINT "I I I I I I"
80 PRINT "I I I I I I"
90 PRINT "I I I I I I"
100 PRINT "I I I I I I"
    
```

Aĩ vai um trabalho braçal com o intuito de que você desenvolva um pouco a sua paciência...

Elabore os programas para obter os seguintes desenhos sō com o auxílio da instrução PRINT:

a) de um transatlântico

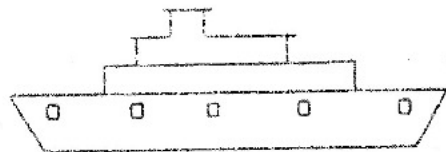


Figura 31

b)

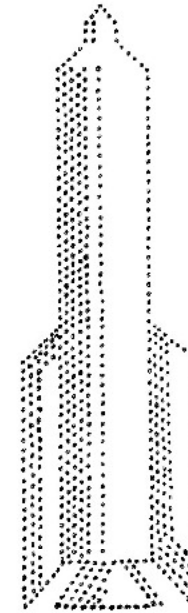


Figura 32

c)

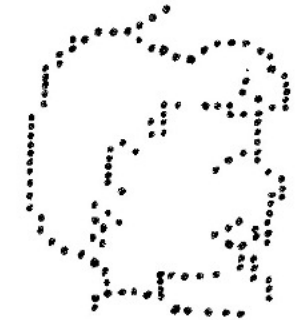


Figura 33

d)



Figura 34

e)

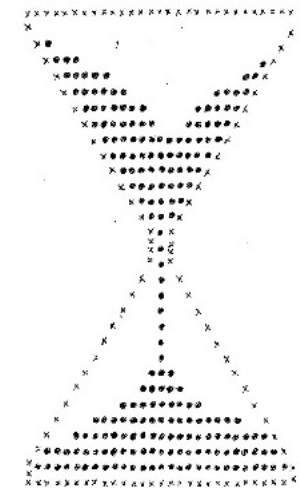


Figura 35

f)



Figura 36

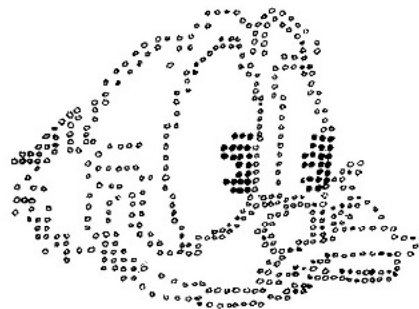


Figura 37

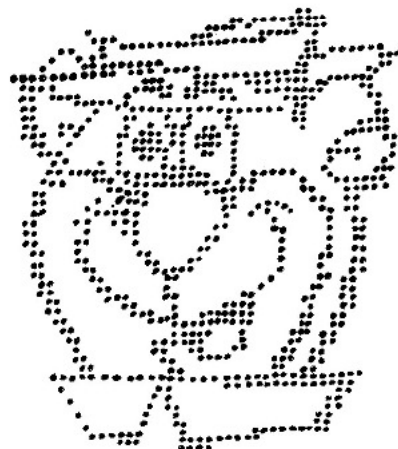


Figura 38

O.I. No "BASIC sem segredos" de nossa autopia, aparecem muitos exemplos como este, nos quais se obtêm desenhos apenas com a instrução PRINT.

Maçã estilizada

Vamos desenhar de forma estilizada a maçã da Figura 39 usando o "esquema" reticulado da Figura 40 onde os números em cada trecho indicam a cor a ser usada.

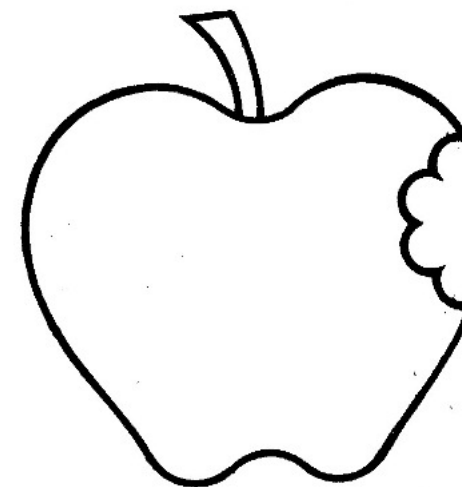


Figura 39

```

10 GR
20 COLOR = 8 → verde amarelado
30 PLOT 20,10 }
40 VLIN 11,14 AT 21 } aqui sai o cabo da maçã
50 COLOR = 13
60 HLIN 17,19 AT 13

```

```

70 HLIN 24, 26 AT 13
80 HLIN 16, 20 AT 14
90 HLIN 23, 27 AT 14
100 HLIN 15, 27 AT 15
110 COLOR = 9 → azul violeta
120 HLIN 15, 26 AT 16
130 HLIN 15, 25 AT 17
140 HLIN 14, 25 AT 18
150 COLOR = 7 → branca
160 HLIN 14, 25 AT 19
170 HLIN 14, 25 AT 20
180 HLIN 14, 26 AT 21
190 COLOR = 10 → verde
200 HLIN 14, 26 AT 22
210 HLIN 14, 27 AT 23
220 HLIN 14, 27 AT 24
230 COLOR = 14 → verde cian
240 HLIN 15, 26 AT 25
250 HLIN 16, 25 AT 26
260 HLIN 16, 25 AT 27
270 COLOR = 4 → azul
280 HLIN 17, 24 AT 28
290 HLIN 17, 24 AT 29
300 HLIN 18, 19 AT 30
310 HLIN 22, 23 AT 30
320 FOR K=1 TO 100: NEXT K → uma pausa para ver a
                             "beleza" de desenho.
330 TEXT: HOME:GOTO 10
    → aqui entramos em um "loop" sem fim

```

Comentários gerais

- 1) A instrução GR (linha 10) define o modo de baixa resolução na tela no formato de 40×40 blocos ou pixels e 4 linhas de texto.

Observe com muito cuidado a orientação dos eixos Ox e Oy (veja a Figura 40).

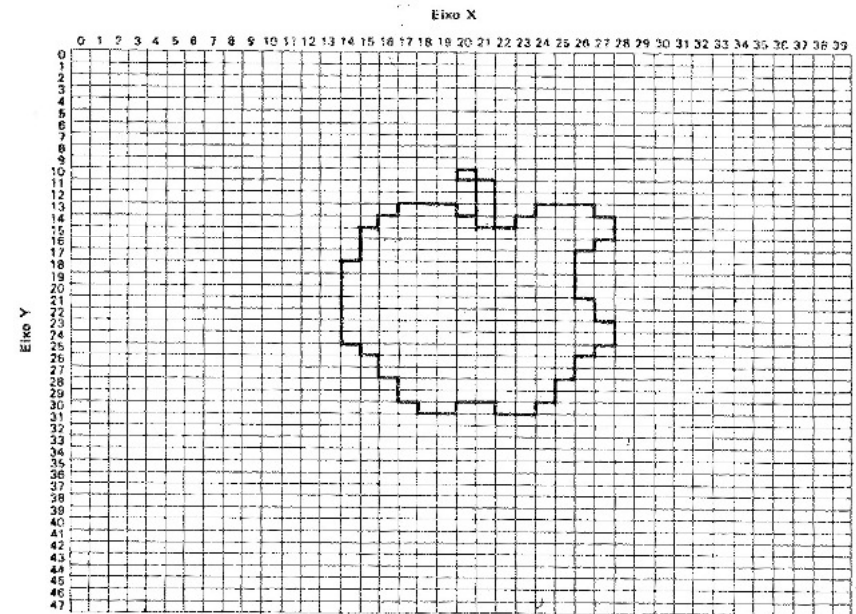


Figura 40

- 2) Usar o TK-2000 para desenhar ou melhor para "plotar" significa definir as coordenadas horizontal e vertical e então colocar um ponto nesta posição.

É com a instrução PLOT X,Y (linha 30) que se imprime um ponto no modo de baixa resolução.

X corresponde a coordenada horizontal (abscissa) e Y a coordenada vertical (ordenada).

3) A instrução COLOR permite a definição de cores no seu aparelho de TV ou no monitor.

É óbvio que se o seu TK-20000 COLOR estiver ligado a um televisor branco e preto esta instrução será inútil...

Quando o TK-20000 COLOR é usado para o traçado de gráficos coloridos em modo de baixa resolução pode-se obter até 6 cores diferentes conforme está indicado na Tabela 11.

0 - preta		
1 - azul	6 - cian	11 - branca
2 - verde	7 - branca	12 - branca
3 - branca	8 - verde	13 - vermelha
4 - azul	9 - azul	14 - cian
5 - vermelha	10 - verde	15 - branca

Tabela 11

A instrução COLOR (linhas 20, 50, 110, 150, 190, 230, 270) é usada necessariamente antes da instrução PLOT ou instrução HLIN ou ainda da VLIN.

4) HLIN coluna 1, coluna 2 AT linha

Com essa instrução obriga-se o TK-20000 COLOR a desenhar uma linha horizontal entre 2 colunas especificadas.

Assim por exemplo no nosso programa temos

100 HLIN 15, 27 AT 15

e isto significa que o TK-20000 COLOR vai desenhar uma linha horizontal na posição 15 (na realidade é na 16ª linha da tela) entre as colunas 15 e 27 (extremos incluídos).

5) VLIN linha 1, linha 2 AT coluna

Com essa instrução o seu TK-20000 COLOR imprime uma linha vertical na coluna indicada entre duas linhas, no modo de baixa resolução.

Assim por exemplo na linha 40 tem-se VLIN 11, 14 AT 21 o que quer dizer que ter-se-á na cor indicada (COLOR = 8) uma linha vertical na coluna 21 entre as linhas 11 e 14 (extremos incluídos).

6) Como se pode perceber claramente a instrução PLOT foi usada para desenhar o começo do "cabinho" da maçã.

Com a instrução VLIN desenhou-se o resto do cabo da maçã.

O resto da maçã estilizada com diferentes cores em cada "secção" foi obtido com o auxílio da instrução HLIN.

Realmente o programa foi feito meio na força bruta e para se ter um programa mais simplificado, uma boa técnica é colocar as instruções HLIN, dentro de um laço FOR-NEXT e ler os valores individuais de uma tabela de dados.

Assim o programa "MAÇÃ ESTILIZADA" poderia ter o seguinte aspecto:

```

10 GR
20 COLOR = 4
30 PLOT 20, 10
40 VLIN 11, 14 AT 21
50 COLOR = 13
60 FOR J = 1 TO 5
70 READ X1, X2, Y

```

```

80 HLIN X1, X2 AT Y
90 NEXT J
100 COLOR = 9
110 FOR K = 1 TO 3
120 READ X1, X2, Y
125 HLIN X1, X2 AT Y
130 NEXT K
:
1000 DATA 17,19,13,24,26,13,16,20,14,23,27,14,15,
      27,15
1100 DATA 15,26,16,15,25,17,14,25,18
1200 DATA 14,25,19,14,25,20,14,26,21
1300 DATA 14,26,22,14,27,23,14,27,24
1400 DATA 15,26,25,16,25,26,16,25,27
1500 DATA 17,24,28 17,24,29,18,19,30,22,23,30

```

- 7) No lugar da pausa criada com o laço FOR-NEXT na linha 320 que tal ter uma pausa musical!
 Você gostou da idéia? Ótimo então coloque no lugar da linha 320 o seguinte trecho:

```

5 DIM MUSIC(8)
314 GOSUB 3010
316 FOR N = 1 TO 8
318 SOUND MUSIC(N), 240
320 NEXT N
322 RESTORE
3010 FOR I = 1 TO 8
3020 READ MUSIC(I)
3040 NEXT I
3045 RETURN
5000 DATA 96,85,76,72,64,56,50,47

```

É oportuno dizer que com a instrução SOUND X,Y gera-se um acorde de frequência determinada por X com duração determinada por Y.

Por exemplo SOUND 96,240 nos dá um DO na escala 4 do piano com duração 240.

Esquentando a moringa

Baseando-se nesse programa obtenha os programas que permitem no modo gráfico de baixa resolução obter o desenho da borboleta da Figura 41 e do palhaço da Figura 42.

No caso do palhaço fica a seu critério definir as cores dos olhos, boca, corpo, sapatos, etc.

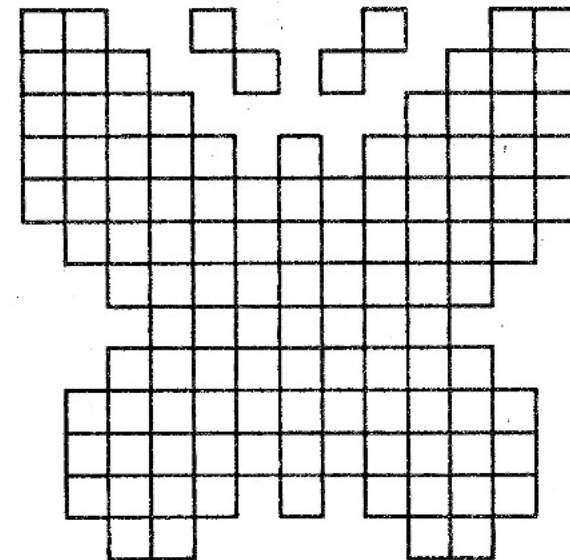


Figura 41

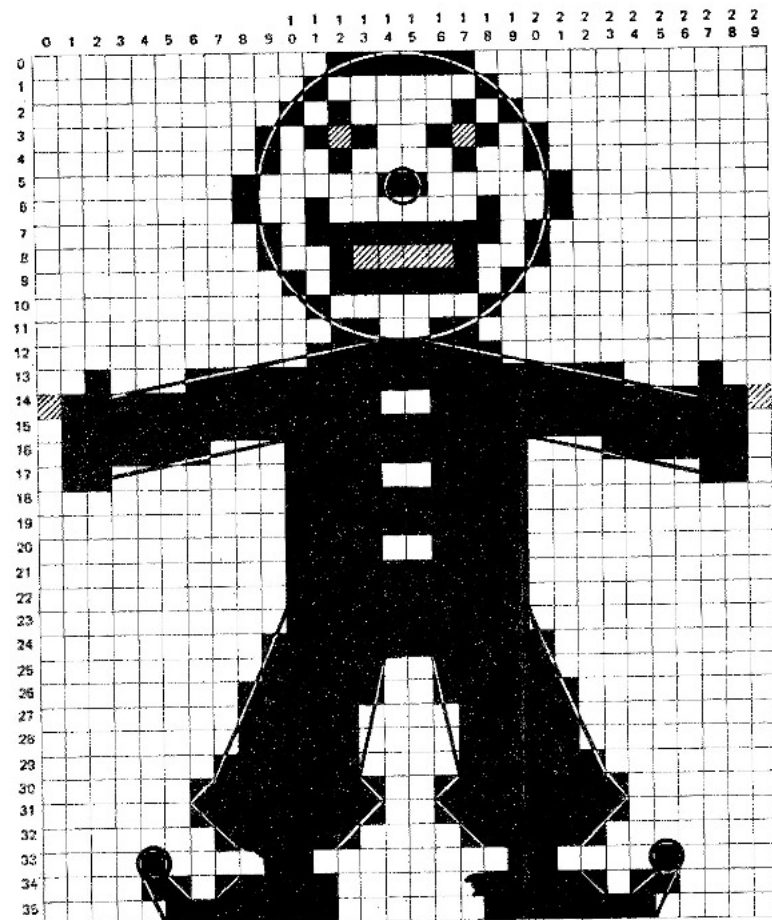


Figura 42

Bandeiras Nacionais

No Atlas Geográfico Melhoramentos de P. Geraldo José Pauwels você encontra caro(a) leitor(a) as bandeiras nacionais dos países do mundo.

Com o auxílio do seu TK-2000 COLOR vamos obter:

a) As bandeiras da Dinamarca, Finlândia e Grécia.

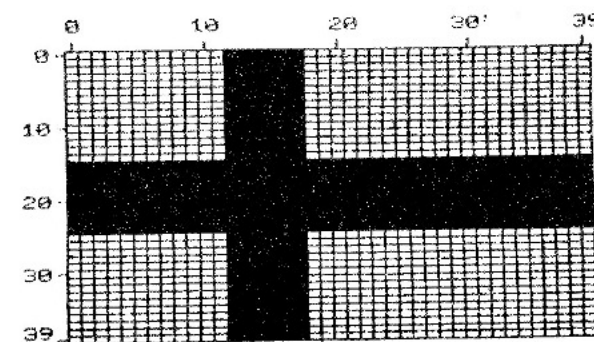


Figura 43a

País	Cruz	Fundo
Dinamarca	Branca	Vermelho
Finlândia	Azul	Branco
Grécia	Branco	Azul

Tabela 12

b) A bandeira do Kuwait

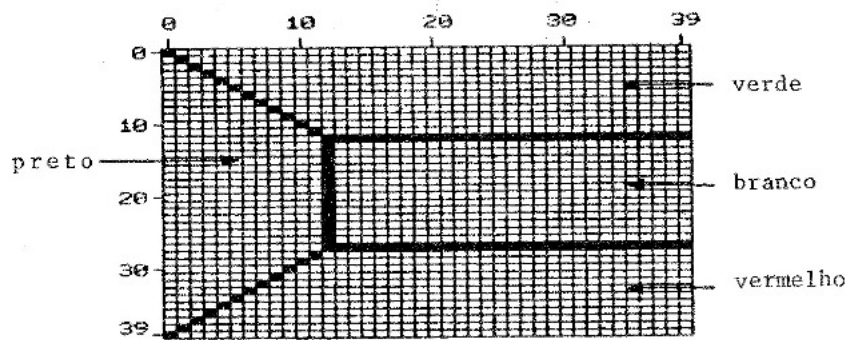


Figura 43b

c) A bandeira da Tanzânia onde o triângulo superior é verde e o inferior é azul.

Os dois triângulos são separados da parte central que é preta por duas faixas diagonais amarelas.

No lugar do amarelo use o verde amarelado (COLOR = 8).

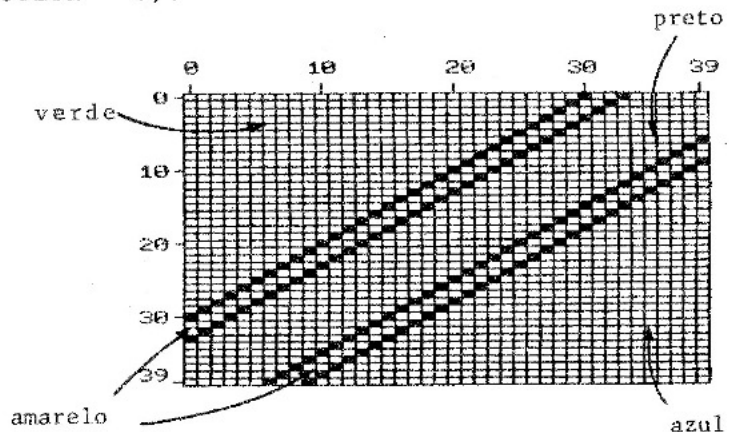


Figura 43c

a) Vamos inicialmente dar o programa para a bandeira da Dinamarca.

```

10 GR
20 REM AT VAI O FUNDO
30 GOSUB 500
40 REM AGORA VAI A CRUZ
50 GOSUB 750
60 END
500 COLOR = 5 → (cor branca)
510 FOR L = 0 TO 39
520 HLIN 0, 39 AT L
530 NEXT L
540 RETURN
750 COLOR = 3 → (cor vermelha)
760 FOR A = 0 TO 39
770 HLIN 12, 18 AT A
780 VLIN 15, 24 AT A
790 NEXT A
800 RETURN
    
```

Para obter as bandeiras dos outros países (Finlândia e Grécia) basta fazer as seguintes mudanças

País	Cruz	Fundo
Finlândia	750 COLOR = 4	500 COLOR = 3
Grécia	750 COLOR = 3	500 COLOR = 9

Tabela 13

Na realidade no caso da bandeira da Grécia também deve-se deslocar um pouco a cruz para a direita.

Será que você consegue fazer isto sozinho?

b) Vamos à bandeira do Kuwait

```

10 GR
20 REM AÍ VAI A PARTE DE CIMA - VERDE
30 GOSUB 500
40 REM AÍ VAI A PARTE DO MEIO - BRANCA
50 GOSUB 750
60 REM AÍ VAI A PARTE DE BAIXO - VERMELHA
70 GOSUB 1000
80 REM AÍ VAI A PARTE TRAPEZOIDAL - PRETA
90 GOSUB 1500
100 END
500 COLOR = 6 → verde cian
510 FOR V = 0 TO 12
520 HLIN 0, 39 AT V
530 NEXT V
540 RETURN
750 COLOR = 3 → branca
760 FOR V = 13 TO 26
770 HLIN 0, 39 AT V
780 NEXT V
790 RETURN
1000 COLOR = 5 → vermelha
1010 FOR V = 27 TO 39
1020 HLIN 0, 39 AT V
1030 NEXT V
1040 RETURN
1500 COLOR = 0
1510 FOR H = 0 TO 12
1520 VLIN H, 39-H AT H
1530 NEXT H
1540 RETURN

```

c) Bem, e agora chegou a vez da bandeira da Tanzânia.

```

10 REM AÍ VAI A BANDEIRA DA TANZÂNIA, SÓ QUE O NOSSO
    AMARELO NÃO É BEM UM AMARELO E SIM UM VERDE AMARELADO
20 GR
30 REM AÍ VÃO AS DIAGONAIS AMARELAS
40 GOSUB 500
50 REM AÍ VÃO OS TRIÂNGULOS AZUL E VERDE
60 GOSUB 1000
70 END
500 COLOR = 8
510 FOR H = 0 TO 30
520 VLIN H, H+3 AT 30-H
530 VLIN H+6, H+9 AT 39-H
540 NEXT H
550 FOR H=0 TO 2
570 VLIN 0,2 - H AT 31+H
580 VLIN 37+H, 39 AT 8-H
590 NEXT H
600 RETURN
1000 FOR V = 0 TO 29
1010 COLOR 6
1020 HLIN 0, 29 -V AT V
1030 COLOR = 9
1040 HLIN 39-V, 39 AT V+10
1050 NEXT V
1060 RETURN

```

Aí vai uma pequena e agradável tarefa para você estimado(a) leitor(a).

Obtenha o programa que permita que seja exibida na tela a bandeira:

a) da Tchecoslovaquia

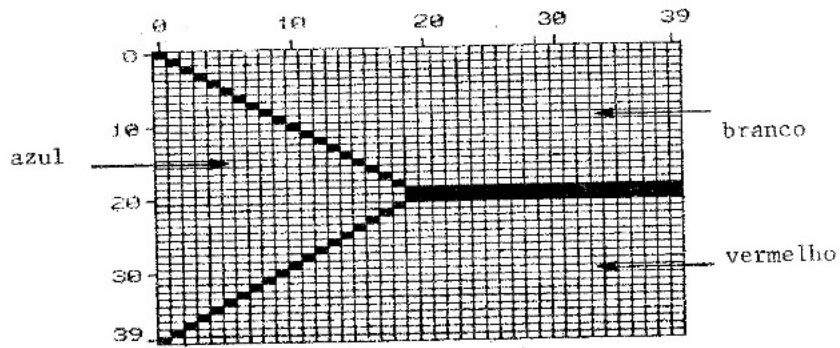


Figura 44

b) do Sudão

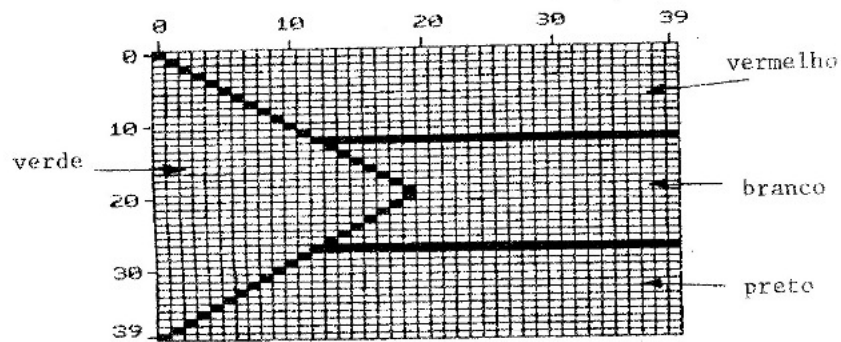


Figura 45

c) de Tonga

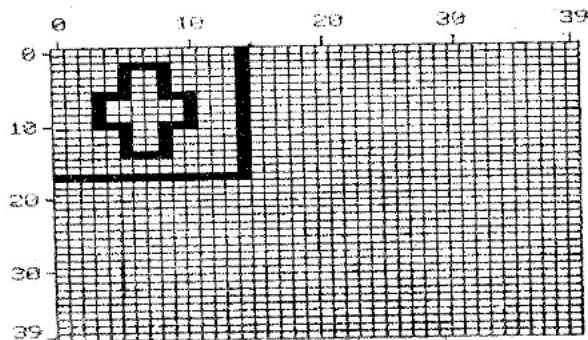


Figura 46

A bandeira de Tonga tem uma cruz vermelha em um campo branco no canto superior da esquerda. O resto da bandeira é todo vermelho.

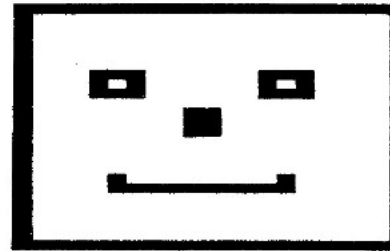
Bem agora você já pode pegar o Atlas do Pauwels e "estraçalhar", digo programar muitas outras bandeiras.

Logo, logo você já estará apto a desenhar a bandeira da nossa querida pátria.

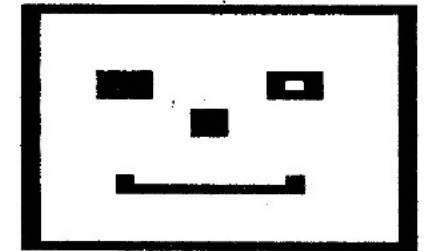
O pisca-pisca

Vamos elaborar um programa que permita ter o rosto mostrado na Figura 47 a e b ou seja na Figura 47a com os olhos abertos e na Figura 47b com o olho esquerdo fechado.

A idéia é obter um "piscador".



a)



b)

Figura 47

```

10 GR : COLOR = 13
20 REM AÍ VAI O DESENHO DO ROSTO COM OS OLHOS ABERTOS
30 HLIN 10,30 AT 5 : HLIN 14,16 AT 14 : HLIN 23,25 AT 14
40 VLIN 16,18 AT 19 : VLIN 16,18 AT 20 : HLIN 15,24 AT 24
50 VLIN 5,30 AT 10 : VLIN 5,30 AT 30 : HLIN 10,30 AT 30
60 PLOT 15,23 : PLOT 24,23
70 HLIN 14,16 AT 12 : HLIN 23,25 AT 12
80 PLOT 14,13 : PLOT 16,13
90 PLOT 23,13 : PLOT 25,13

```

```

100 REM VAMOS AGORA FECHAR O OLHO ESQUERDO
110 COLOR = 3 : PLOT 15,13
120 REM VAMOS DAR UMA PAUSA PARA QUE O OLHO PERMANEÇA
    FECHADO
130 FOR K = 1 TO 43 : NEXT K
140 REM VAMOS ABRIR O OLHO QUE ESTAVA FECHADO
150 COLOR = 0 : PLOT 15,13
160 REM VAMOS DAR UM TEMPO PARA ELE FICAR ABERTO
170 FOR K = 1 TO 100 : NEXT K
180 GOTO 110

```

É evidente que para se fazer um programa como esse convém antes de mais nada ter um papel quadriculado como o indicado na Figura 48 que é como realmente está dividida a tela do seu TK-2000 COLOR no caso de gráficos em modo de baixa resolução.

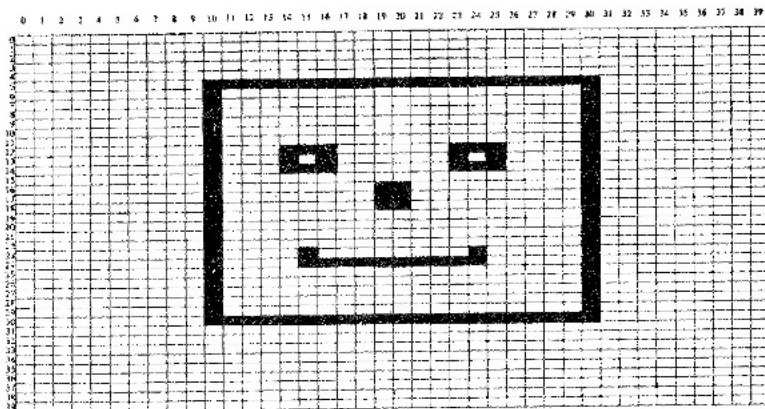


Figura 48

Aí fica bem mais simples o entendimento de todas as instruções.

Dessa forma com HLIN 10,30 AT 5 (linha 30) desenha-se a linha de topo do rosto, com VLIN 16,18 AT 19 : VLIN 16,18 AT 20 (linha 40) desenha-se o nariz, etc.

Lá vai um pequeno "raxakuka" (como diz o amigo Wilson J. Tucci no seu excelente "Por dentro do APPLE").

- a) Que tal você melhorar esse programa fazendo também o olho direito piscar?
- b) Que tal você fazer com que ambos os olhos pisquem alternadamente?
- c) Faça a boca mexer um pouco e os olhos abrirem e fecharem 50 vezes consecutivamente.

O.I. - Observação importante

Para simplificar futuros programas seus apresentamos uma folha quadriculada que lhe será muito útil para gráficos de baixa resolução (Figuras 49 e 50).

Que tal você fazer um programa homenageando a Fundação Armando Álvares Penteado (F.A.A.P.) escrevendo as letras em branco em um fundo azul ocupando toda a tela (veja a Figura 49).

Você consegue fazer com que a sigla se movimente tanto no sentido horizontal como no vertical?

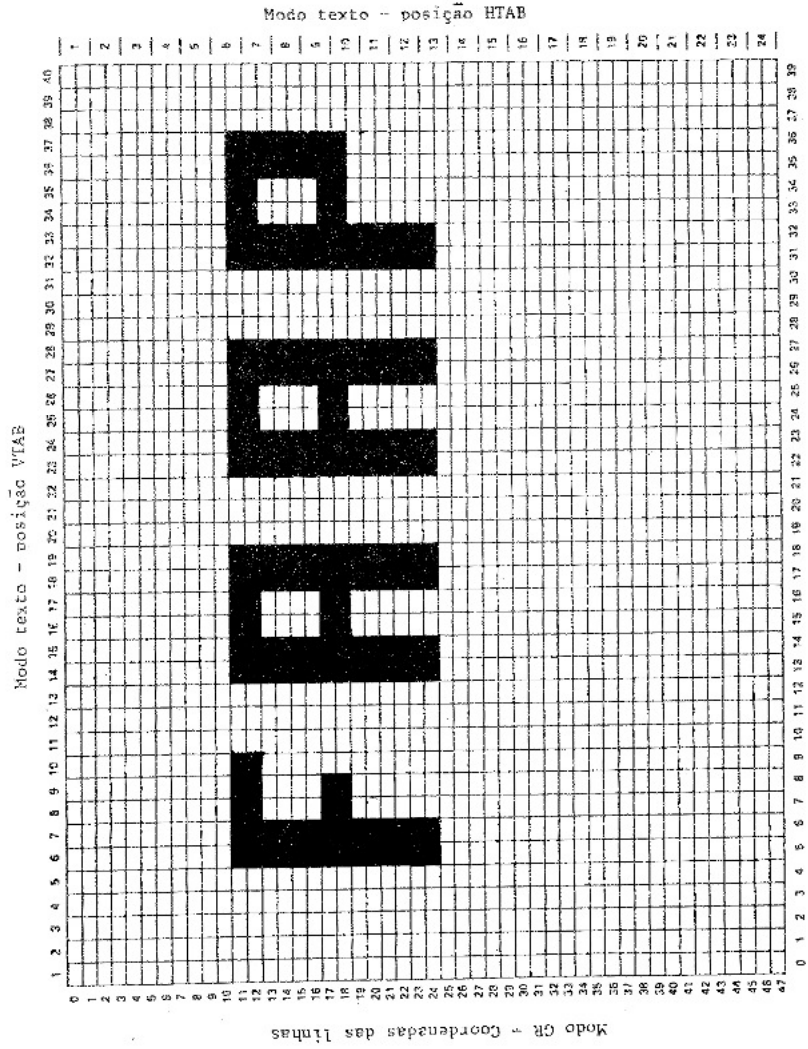


Figura 49

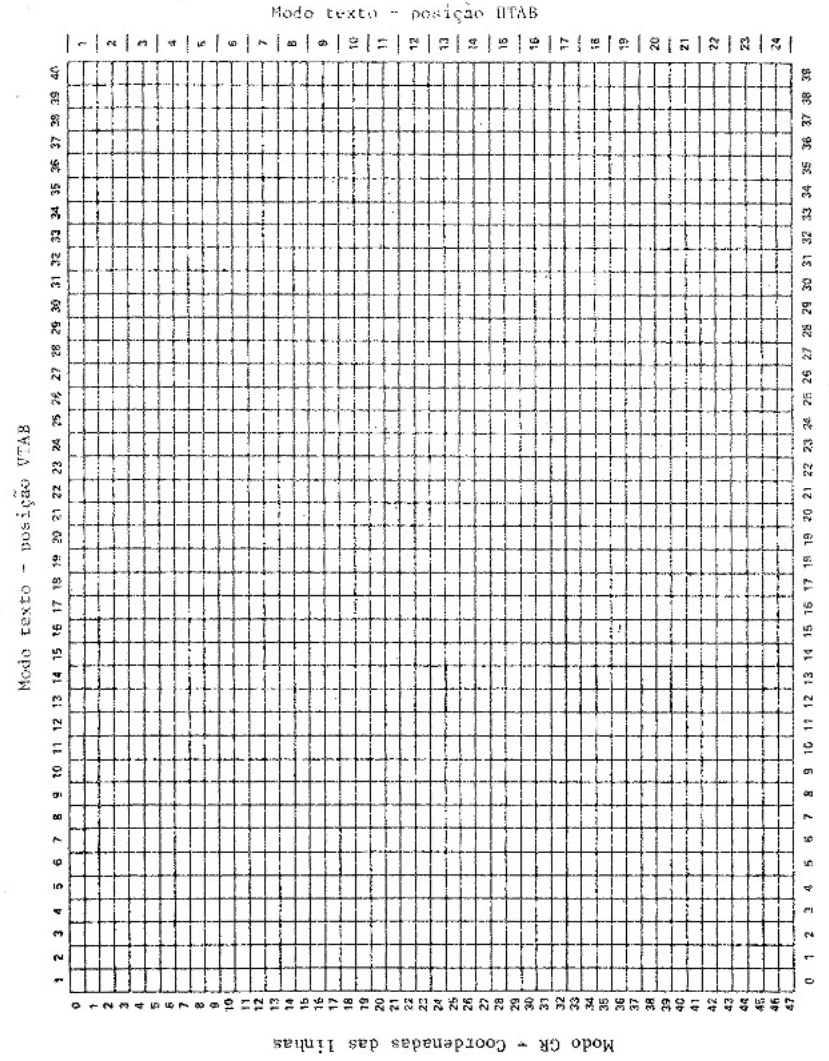
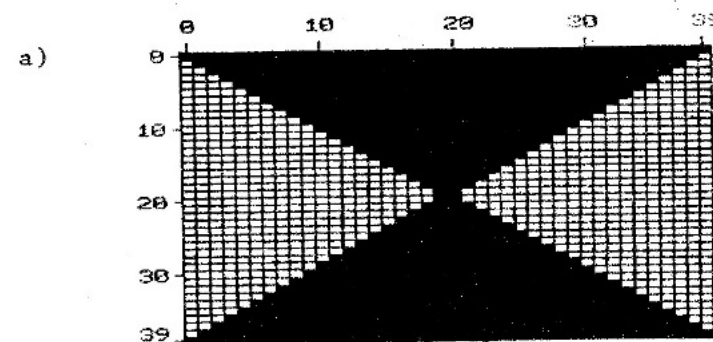
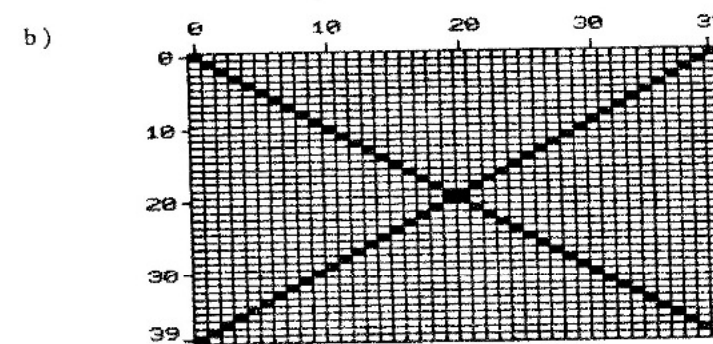


Figura 50

Simetrias gráficas

Faremos um programa único que permita ao gosto do leitor(a) obter qualquer um dos seguintes gráficos:

Figura
51Figura
52

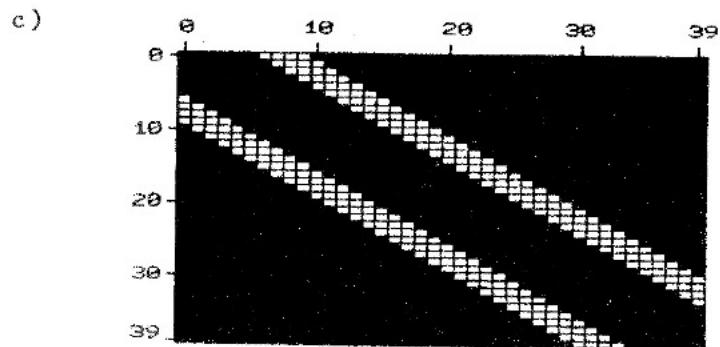


Figura 53

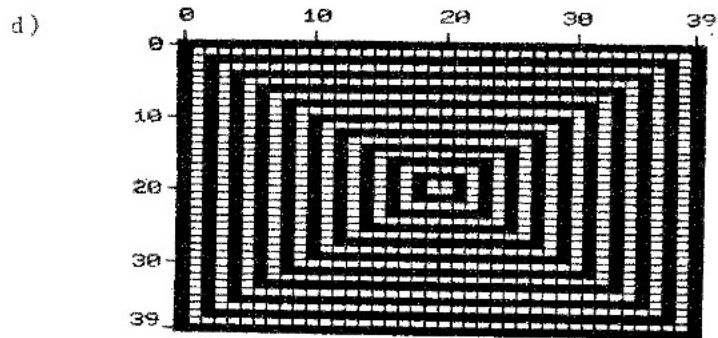


Figura 54

Bem aí vai o programa que terá a cor escolhida aleatoriamente.

```

5 HOME
10 PRINT "ENTRE COM A,B,C OU D CONFORME O DESENHO QUE VOCÊ
  QUEIRA"
20 INPUT R$
30 IF R$ = "A" THEN 80
40 IF R$ = "B" THEN 150
50 IF R$ = "C" THEN 220
60 IF R$ = "D" THEN 350
70 PRINT "NÃO BRINQUE, ENTRE COM A LETRA CORRETA": GOTO 5
75 FOR Z = 1 TO 499 : NEXT Z
77 GOTO 5

```

```

80 GR
90 COLOR = INT(15 * RND(1))
100 FOR J = 0 TO 19
110 HLIN J, 39-J AT J
120 HLIN 39-J, J AT 39-J
130 NEXT J
140 GOTO 430
150 GR
160 COLOR = INT(15 * RND(1))
170 FOR X = 0 TO 39
180 PLOT X, X
190 PLOT 39-X, X
200 NEXT X
210 GOTO 430
220 GR
225 COLOR = INT(15 * RND(1))
230 FOR K = 0 TO 19
235 REM: AÍ VAI A PARTE DE CIMA DA SECÇÃO CENTRAL
240 HLIN K, K+5 AT K
250 VLIN K, K+5 AT K
255 REM: AÍ VAI A PARTE DE BAIXO DA SECÇÃO CENTRAL
260 HLIN 39-K, 34-K AT 39-K
270 VLIN 39-K, 34-K AT 39-K
280 NEXT K
290 FOR K = 0 TO 29
300 REM AÍ VAI O TRIÂNGULO SUPERIOR
310 HLIN K+10, 39 AT K
320 REM AÍ VAI O TRIÂNGULO INFERIOR
325 HLIN 0,K AT K+10
330 NEXT K
340 GOTO 430
350 GR
360 COLOR = INT(15 * RND(1))

```

```

370 FOR I = 0 TO 19 STEP 2
380 HLINE I, 39-I AT I
390 VLINE I, 39-I AT 39-I
400 HLINE 39-I, I AT 3-I
410 VLINE 39-I, I AT I
420 NEXT I
430 FOR Z = 1 TO 999 : NEXT Z
440 TEXT: HOME
450 PRINT "VOCÊ QUER VER ALGUM DESENHO MAIS (S/N)?"
460 INPUT A$
470 IF A$ = "S" THEN 5
480 END

```

Mais um "raxakuka" para você culto(a) leitor(a).

Faça os programas para obter

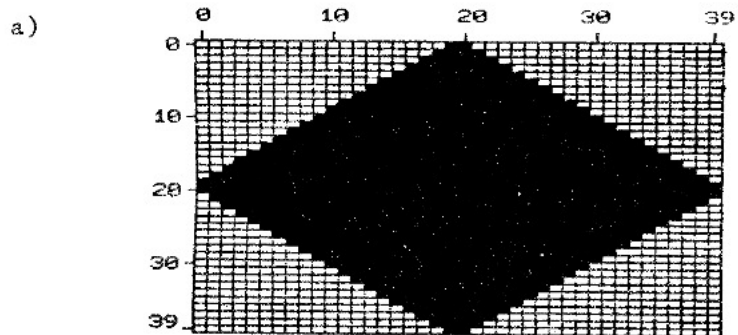


Figura 55

O.I. Aí está uma parte para se chegar a bandeira do nosso país.

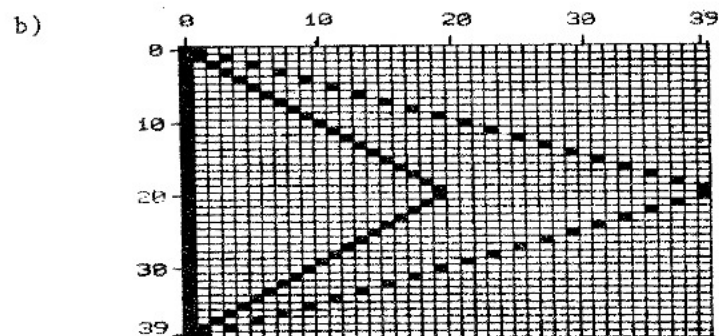


Figura 56

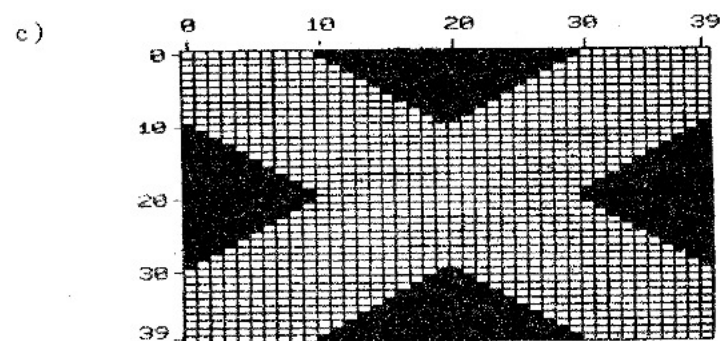


Figura 57

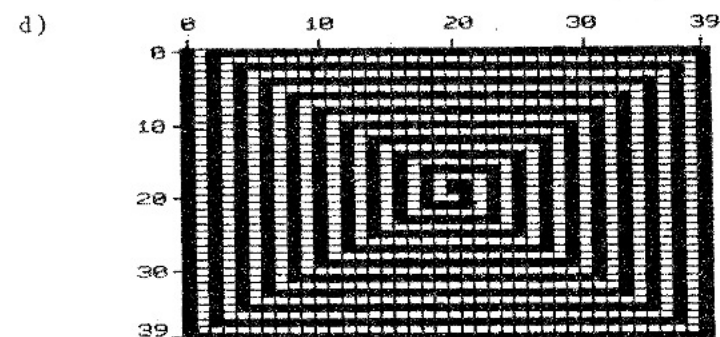


Figura 58

Caleidoscópio

O gráfico colorido é realmente uma área exuberante e de interesse para todos.

O programa CALEIDOSCÓPIO talvez seja o mais rico em cores de todos os que apresentamos neste livro destinado fundamentalmente para divulgar as grandes qualidades do TK-2000 COLOR.

Usaremos no programa "CALEIDOSCÓPIO" mais uma vez o modo gráfico de baixa resolução com o que ganharemos muito em colorido.

Vamos fazer com que cada um dos $40 \times 40 = 1600$ pixels mude de cor rapidamente de uma forma cíclica e simétrica.

Dois laços FOR-NEXT aninhados fazem com que se passe pelos pixels e se ajuste o número da cor (COLOR), através de uma função seno ou cosseno das coordenadas do pixel ou seja do retângulo ativado na tela.

Essas coordenadas são sempre truncadas para um valor inteiro.

A função SCRN é usada para se achar as cores dos pixels.

Em outras palavras a instrução SCRN, em baixa resolução, fornece o código da cor cujo ponto é determinado pelas coordenadas dadas.

Os laços (loops) são indexados com as variáveis I e J que são as variáveis mais importantes do programa.

```

10 REM ***CALEIDOSCÓPIO***
15 GR:REM**AVANÇO PARA O CENTRO
20 FOR I=0 TO 19
30 FOR J=I TO 20
40 COLOR = INT (ABS(15*SIN(I+J))) : GOSUB 230
50 NEXT J : NEXT I
60 REM ***AVANÇO PARA AS BORDAS DA TELA***
70 FOR I=19 TO 0 STEP -1
80 FOR J=I TO 20
90 COLOR = INT (ABS(15*COS(I+J))) : GOSUB 230
100 NEXT J : NEXT I
110 REM **RETROCEDENDO PARA DENTRO***
120 FOR I=0 TO 19
130 FOR J=I TO 20
140 COLOR = 15 - SCRN(I,J) : GOSUB 230
150 NEXT J
160 NEXT I
170 REM **VOLTANDO PARA FORA**
180 FOR I=19 TO 0 STEP -1
190 FOR J=I TO 20
200 COLOR = SCRN(I,J)+1 : GOSUB 230
210 NEXT J : NEXT I
220 GOTO 120 : REM **ASSIM SEGUE O CICLO**
230 REM SUBROTINA QUE DESENHA POR VEZ OITO PONTOS DISPOSTOS
SIMETRICAMENTE EM RELAÇÃO AO CENTRO DA TELA
240 PLOT I,J : PLOT J,I : PLOT 39-I, J : PLOT 39-J, I
250 PLOT 39-I, 39-J : PLOT 39-J, 39-I : PLOT I,39-J :
PLOT J, 39-I
260 RETURN

```

Exercício para a sua massa cinzenta

Faça um programa que permita obter inicialmente o boneco apresentado na Figura 57.

O corpo do boneco deve ser azul.

Acrescente ao seu boneco olhos, nariz e um sorriso que devem ter cor preta.

Além disto coloque na sua cabeça um chapéu verde e braços de cor vermelha.

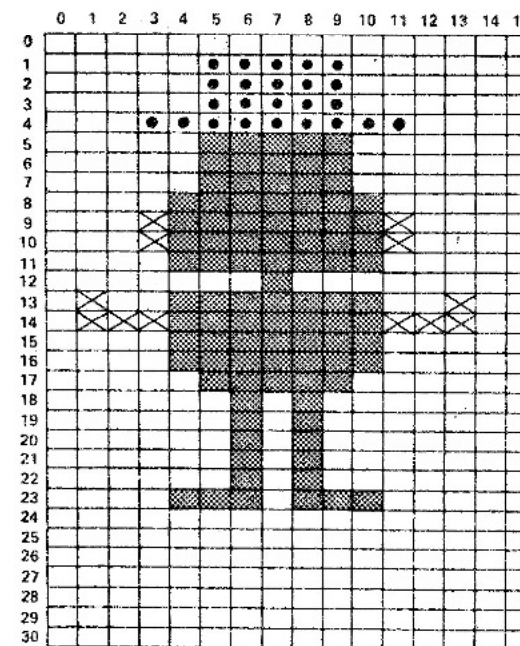


Figura 57

Traçando em alta resolução

Inicialmente, usando o traçado em alta resolução, desenharemos um quarto a três dimensões com uma porta na parede.

A alta resolução refere-se a traçados com mais detalhes e melhor definição.

A capacidade de alta resolução no TK-2000 COLOR pode ser de $280 \times 160 = 44800$ pontos (veja a Figura 58) obtida com a instrução HGR1 ou de $280 \times 192 = 53760$ pontos acionada pela instrução HGR2 (veja a Figura 59).

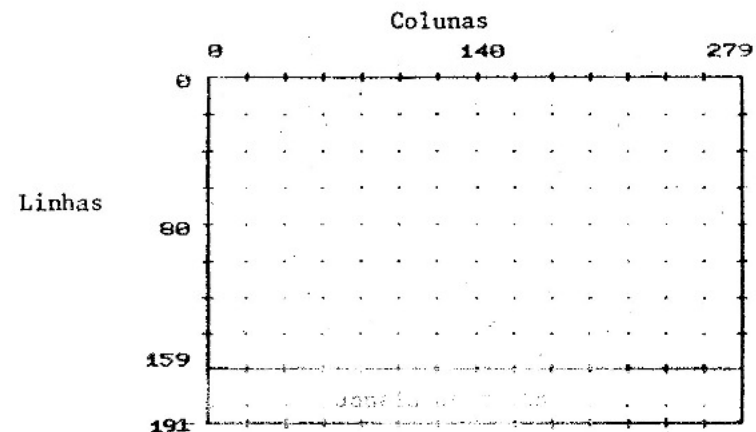


Figura 58

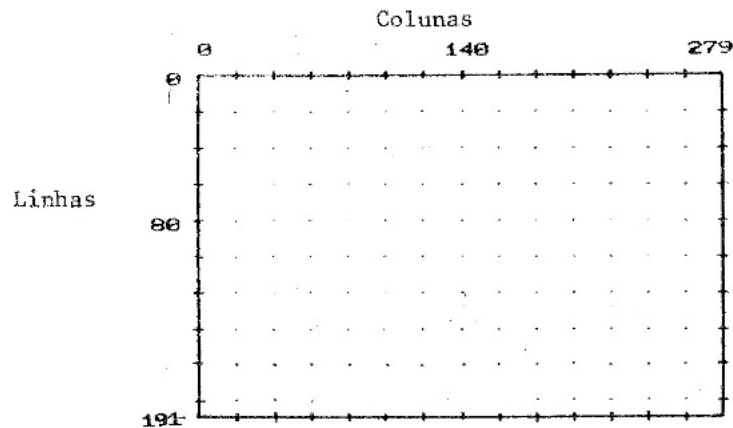


Figura 59

Como se pode observar facilmente no modo de alta resolução gráfica acionado por HGR as linhas finais da tela permanecem no modo texto.

Já o HGR2 faz com que o modo de alta resolução gráfica passe a preencher a tela toda ou seja permitindo a definição de 280 x 192 pontos ou pequenos pixels na tela.

A instrução HCOLOR é usada no modo alta resolução da mesma maneira que o COLOR no modo de baixa resolução.

A forma geral dessa instrução é

```
HCOLOR = número
```

e o número da cor em alta resolução está na Tabela 13.

0 - preto	4 - azul
1 - azul	5 - vermelho
2 - verde	6 - cian
3 - branco	7 - branco

Tabela 13

De forma análoga a PLOT, HPLOT permite o traçado isto é a plotagem em alta resolução; porém com bem mais pontos como já sabemos.

Porém é com a instrução HPLOT X1,Y1 TO X2, Y2 que se conseguem resultados rápidos no traçado de alta resolução.

Com essa instrução pôde-se traçar uma reta entre os pontos de X1,Y1 e X2,Y2.

Bem aí vai um programinha tridimensional (o do quarto...)

```
a) 10 HGR
    20 HOME
    30 HCOLOR = INT(7*RND(1))
    40 HPLOT 10,35 TO 10,110 TO 150, 110 TO 150,35 TO 10,35
    50 HPLOT 55,55 TO 55,80 TO 105,80 TO 105,55 TO 55,55
    60 HPLOT 10,35 TO 55,55
    70 HPLOT 105,55 TO 150,35
    80 HPLOT 105,80 TO 150,110
    90 HPLOT 10,110 TO 55,80
    100 HPLOT 80,80 TO 80,70 TO 90,70 TO 90,80
    110 HPLOT 87,76
    120 FOR Z=1 TO 399:NEXT Z
    130 GOTO 10
```

Com essas instruções se ligam os vértices dos retângulos

Esse pontinho é o trinco

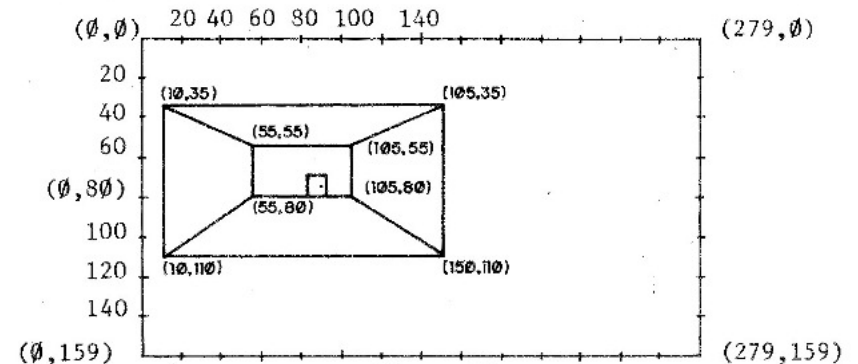


Figura 60

b) Vamos agora mostrar um carro estilizado o qual mais tarde vai se mover (veja o ítem c).

```

10 HGR: HCOLOR = INT(7*RND(1))
15 TEXT: HGR
20 INPUT X,Y → como exemplo entre com X=60 e Y=130
30 HPLOT X,Y TO X+30, Y TO X+37, Y+13 TO X+48, Y+18 TO
   X+48, Y+25 TO X-12, Y+25 TO X-12, Y+18 TO X, Y+18 TO
   X,Y →————→ (Com a linha 30
40 HPLOT X-2, Y+26 TO X-2, Y+28 }   desenha-se o con-
50 HPLOT X-1, Y+29 TO X+3, Y+29 }   torno do carro)
60 HPLOT X+4, Y+28 TO X+4, Y+26 }
70 HPLOT X+28, Y+26 TO X+28, Y+28 }   Com essas instru-
80 HPLOT X+29, Y+29 TO X+33, Y+29 }   ções desenha-se
90 HPLOT X+34, Y+28 TO X+34, Y+26 }   as rodas
100 FOR Z=1 TO 499 : NEXT Z
110 HOME
120 GOTO 10

```



Figura 61

O.I. É bom o caro(a) leitor(a) ver o exemplo no qual está um esboço de como se obtém um avião e com isso clarearão as idéias para obter um carro.

c) Vamos colocar agora um carro do tipo de Figura 61 em movimento.

```

10 HGR
20 FOR Y=20 TO 160 STEP 20
30 FOR X=230 TO 25 STEP -15
40 HCOLOR = 3 : GOSUB 500
50 HCOLOR = 0 : GOSUB 500
60 NEXT X
70 NEXT Y

```

```

80 GOTO 10
500 HPLOT X,Y TO X+20, Y TO X+27, Y+8 TO X+38, Y+8 TO
   X+38, Y+15 TO X-22, Y+15 TO X-22, Y+8 TO X-10,
   Y+8 TO X,Y
510 HPLOT X-12, Y+16 TO X-12, Y+13
520 HPLOT X-11, Y+19 TO X-7, Y+19
530 HPLOT X-6, Y+18 TO X-6, Y+16
540 HPLOT X+18, Y+16 TO X+18, Y+18
550 HPLOT X+19, Y+19 TO X+23, Y+19
560 HPLOT X+24, Y+18 TO X+24, Y+16
570 RETURN

```

Vamos agora despertar (assim espero...) o seu dom artístico!!!

Faça um programa para cada uma das seguintes situações ou imagens de objetos

- uma pessoa esquiando;
- um peixe;
- um avião a jato;
- uma pessoa pescando em um bote.

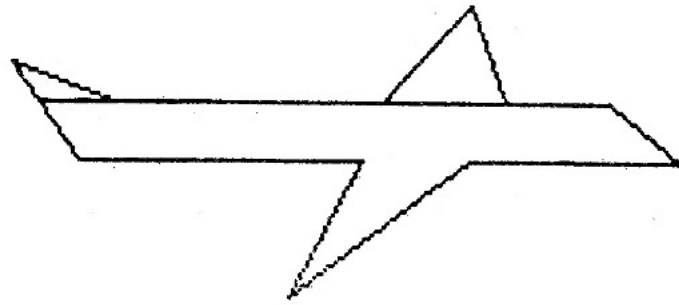


a)

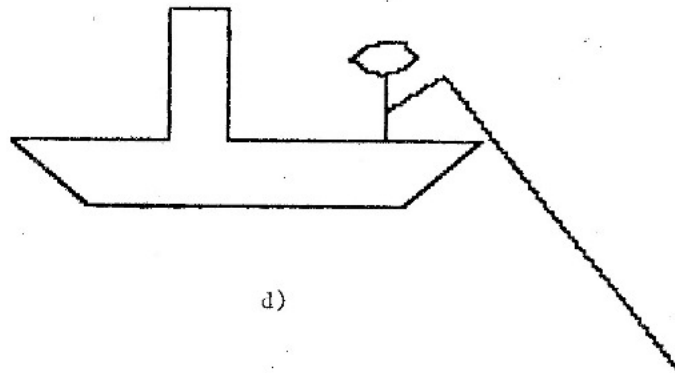


b)

Figura 62



c)



d)

Figura 62

Gráficos Radiais

Vamos agora elaborar, usando e abusando da instrução HPLLOT alguns gráficos radiais como os mostrados na Figura 63 a e b.

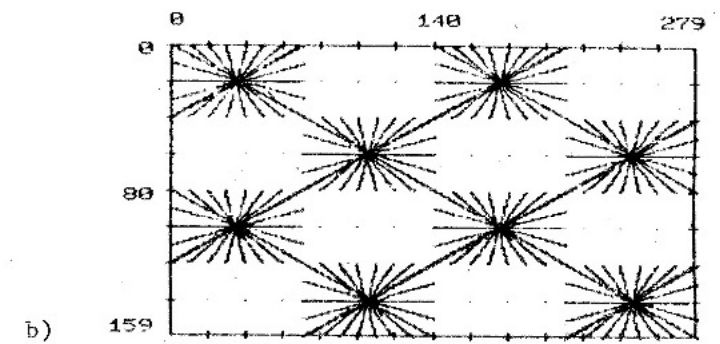
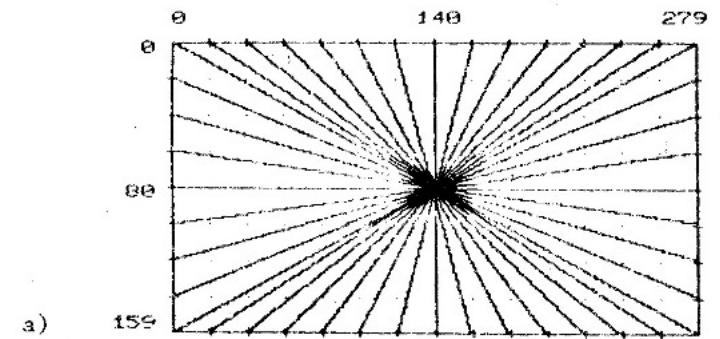


Figura 63

```

10 REM ESTE É UM PROGRAMA PARA OBTEN NO MODO DE ALTA
RESOLUÇÃO UM DESENHO RADIAL
15 REM AÍ VAI A PRIMEIRA METADE
20 HGR
30 HCOLOR = 3
40 H PLOT 0,0 TO 279,159 → Com essa instrução
50 FOR X=20 TO 260 STEP 20 → é desenhada a dia-
60 H PLOT X,0 TO 280-X,159 → gonal principal
70 NEXT X
80 REM AÍ VAI A SEGUNDA METADE
90 H PLOT 0,159 TO 279,0 → Com essa instrução
100 FOR Y=20 TO 140 STEP 20 → é representada na
110 H PLOT 0,Y TO 279,160-Y → tela a diagonal se-
120 NEXT Y → cundária
130 END

```

b) Agora vamos para o 2º desenho (Figura 63b) no qual tem-se 8 desenhos radiais menores.

Antes de mais nada vamos definir as variáveis que serão usadas no problema.

{	XE, YE	→ coordenadas do canto superior da esquerda
	H	→ amplitude do desenho na horizontal
	V	→ amplitude do desenho na vertical
	EH	→ espaçamento na direção horizontal
	EV	→ espaçamento na direção vertical

Como o canto superior da esquerda está especificado por (XE, YE) e o "vão" na horizontal é H então o canto superior da direita tem coordenadas (XE+H, YE) e os outros cantos inferiores tem coordenadas (XE+H, YE+V) e (XE, YE+V) conforme está mostrado na Figura 64.

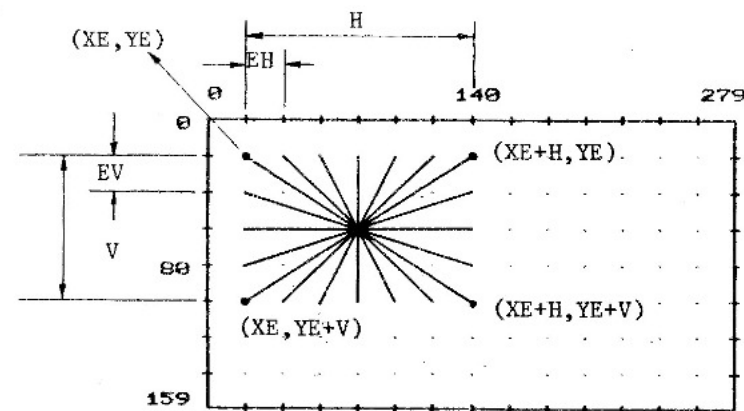


Figura 64

Aí vai o programa

```

10 REM H,V,EH E EV SÃO OS MESMOS PARA TODOS OS DESENHOS
RADIAIS
20 HGR
30 HCOLOR = INT(7*RND(1))
40 H=70
50 V=40
60 EF=10 : EV=EH
70 REM AÍ VÃO OS DESENHOS DA PRIMEIRA E DA TERCEIRA LINHAS
80 FOR XE=0 TO 140 STEP 2*H
90 FOR YE=0 TO 30 STEP 2*V
100 GOSUB 1000
110 NEXT YE
120 NEXT XE
130 REM AÍ VÃO OS DESENHOS DA SEGUNDA E DA QUARTA LINHAS
140 FOR XE=70 TO 210 STEP 2*H
150 FOR YE=40 TO 120 STEP 2*V
160 GOSUB 1000
170 NEXT YE
180 NEXT XE

```

Mude depois os valores de H e V e veja o que ocorre!!!
Cuidado para não cair fora da tela!?

```

190 FOR Z=1 TO 499 : NEXT Z
200 GOTO 20
1000 REM PRIMEIRO VAMOS DESENHAR AS RETAS DE CIMA PARA
      BAIXO
1005 D=EH
1010 FOR X=XE + EH TO XE+H - EH STEP EH
1020 HPLLOT X, YE TO XE+H-D, YE+V
1030 D=D+EH
1040 NEXT X
1050 REM AÍ VÃO AS RETAS DA ESQUERDA PARA A DIREITA
1060 D=EV
1070 FOR Y=YE+EV TO YE+V-EV STEP EV
1080 HPLLOT XE, Y TO XE+H-1, YE+V-D
1090 D=D+EV
1100 NEXT Y
1110 REM AÍ VAI A DIAGONAL PRINCIPAL
1120 HPLLOT XE, YE TO XE+H-1, YE+V
1130 REM AÍ VAI A DIAGONAL SECUNDÁRIA
1140 HPLLOT XE, YE+V TO XE+H-1, YE
1150 RETURN

```

Que tal você, com o auxílio das "dicas" das nesse programa obter os seguintes gráficos:

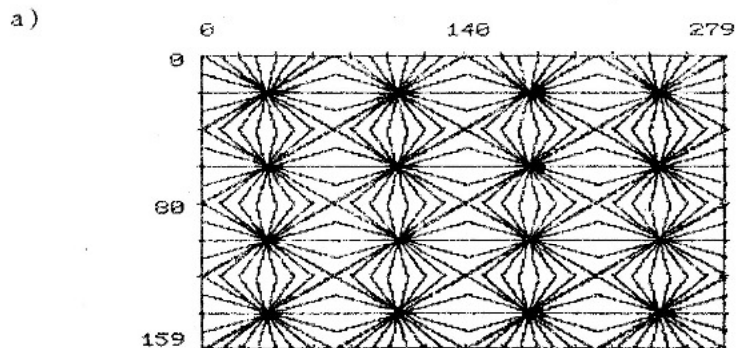


Figura 65

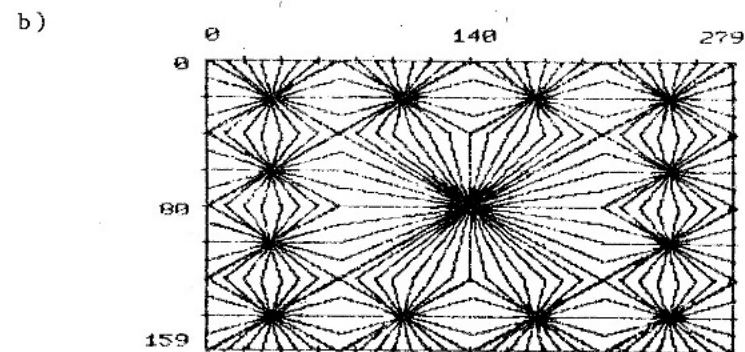


Figura 66

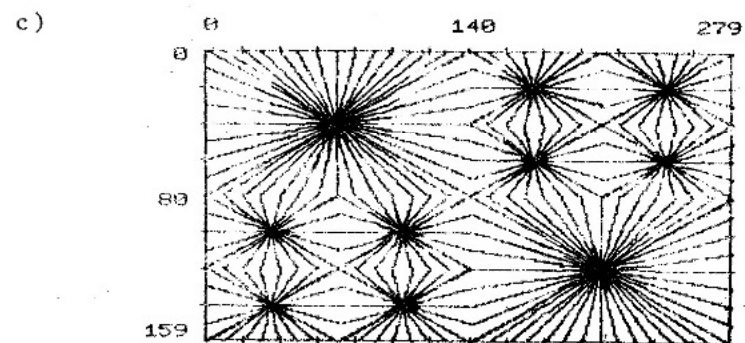


Figura 67

O.I. - É muito fácil obter todos esses gráficos radiais e para isto basta que você use uma GRAFIX.

Leia o que está escrito na página 189, viu!!

Reticulados sofisticados

Vamos apresentar o programa RETICULADOS SOFISTICADOS com o qual obtêm-se um desenho do tipo mostrado na Figura 68.

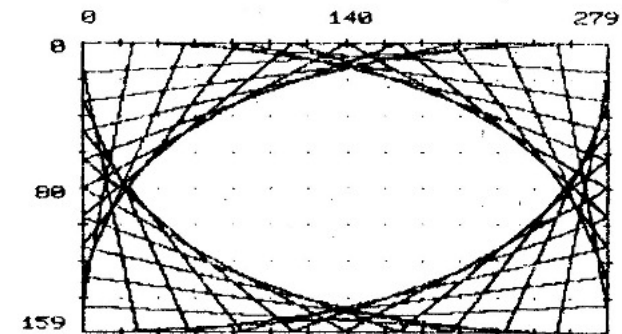


Figura 68

Aí vai o programa:

```

10 REM A ARTE SÓ COM O USO DAS RETAS
20 REM RETICULADOS SOFISTICADOS
30 HGR
40 HCOLOR = 5
50 RETAS=10
60 REM AÍ VAO OS INCREMENTOS
70 IX = 280/RETAS: IY = 160/RETAS

```

aqui se define quantas
retas serão traçadas
de cada bordo

75 REM VAMOS AGORA DO LADO DIREITO E PARA CIMA E DO LADO
ESQUERDO E PARA BAIXO

80 Y = 0

90 FOR X=0 TO 279 STEP IX

100 HPLOT X,0 TO 279,Y

105 HPLOT X,159 TO 0,Y

110 Y=Y+IY

120 NEXT X

130 REM AGORA VAI O DESENHO DO
LADO DIREITO E PARA BAIXO
E DA ESQUERDA E PARA CIMA

140 X=0

150 FOR Y=159 TO 0 STEP -IY

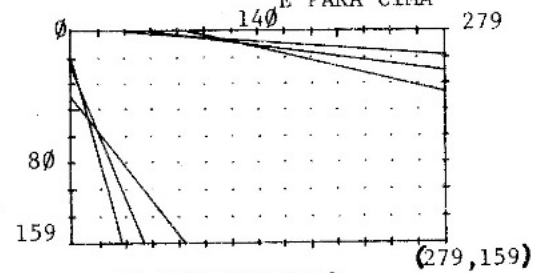
160 HPLOT 0,Y TO X,0

165 HPLOT 279, Y TO X, 159

170 X=X+IX

180 NEXT Y

190 END



DO LADO ESQUERDO
E PARA BAIXO

Figura 69

Alguns exercícios para você respeitado(a)
leitor(a).

a) Mude a linha 50 para RETAS = 5

b) Introduza uma outra cor para o laço FOR-NEXT de
140-180.

Faça por exemplo 135 HCOLOR = 6

c) Faça um programa para obter o desenho da Figura
70.

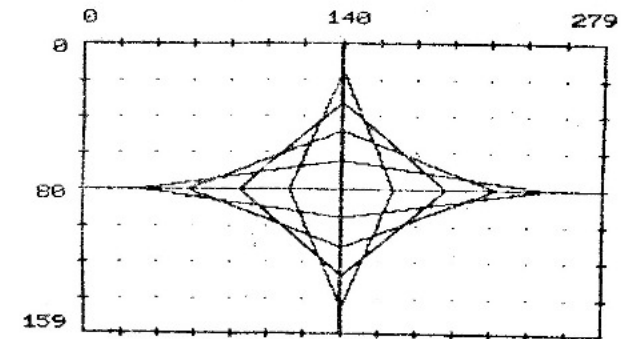


Figura 70

d) Refaça algumas instruções do programa do item c)
para se ter mais de 4 segmentos de reta em cada
quadrante e escolha uma cor diferente para cada
um.

O.I. - Novamente lhe recordo que estes gráficos são
o que existe de mais simples de se tirar do TK-2000
COLOR quando ele está conectado a uma GRAFIX.

Na página 189 você encontra algumas especifica-
ções sobre esta magnífica impressora.

Manipulação de triângulos retângulos e paralelogramos

Vamos agora, em modo de alta resolução, obter lindos desenhos usando apenas triângulos, retângulos e paralelogramos.

Aí vão os programas para obter os desenhos mostrados nas Figuras 71, 72 e 73.

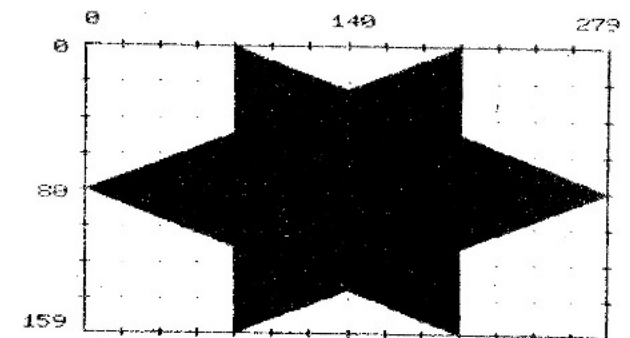


Figura 71

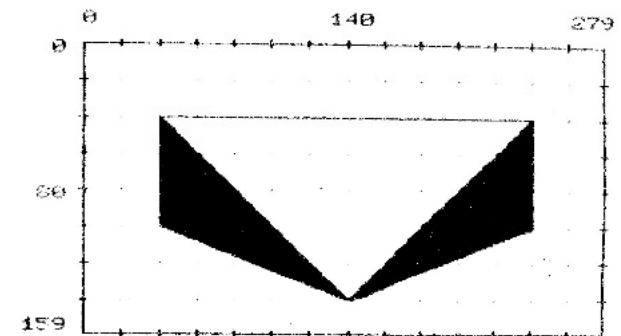


Figura 72

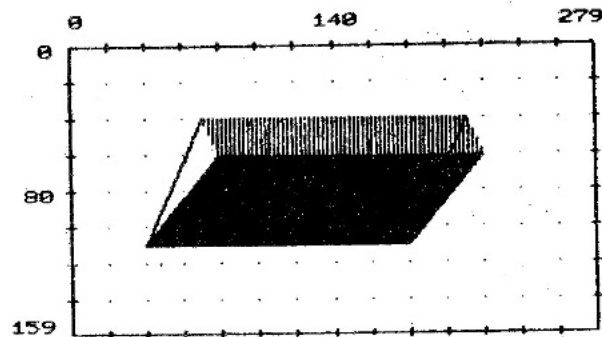


Figura 73

a) 10 REM AÍ VAI O DESENHO DA ESTRELA DE SEIS PONTAS

```

20 HGR
30 HCOLOR = INT(7*RND(1))
40 FOR Y = 0 TO 159
50 HPLOT 0,80 TO 200,Y
60 HPLOT 279,80 TO 80,Y
70 NEXT Y
80 FOR Z=1 TO 499: NEXT Z
90 GOTO 20

```

Daqui sai o feixe de retas com o vértice fixo em (0,80)

Daqui sai o feixe de retas com o vértice fixo em (279,80)

b) 10 REM UM DESENHO COM TRÊS DIMENSÕES FEITO COM AUXÍLIO DE DOIS TRIÂNGULOS

```

20 HGR
30 HCOLOR = 2
40 FOR Y = 40 TO 100
50 HPLOT 40,Y TO 140,140 TO 240,Y
60 NEXT Y
70 HPLOT 40,40 TO 240,40
80 END

```

c) 10 REM DESENHO DE UM TRIÂNGULO SÓLIDO

```

20 HGR
30 REM O FUNDO CONSTITUIRÁ UMA FACE DO SÓLIDO
40 HCOLOR = 6
45 REM (X1,Y1) E O VÉRTICE SUPERIOR DA ESQUERDA, (X2,Y2)
   E O VÉRTICE INFERIOR DA ESQUERDA E (X3,Y3) E O VÉRTICE
   SUPERIOR DA DIREITA

```

50 INPUT X1, Y1, X2, Y2, X3, Y3 → Para o nosso desenho (Figura 73) deve-se

entrar com

$$\begin{cases} X1=70 \\ Y1=40 \end{cases} \begin{cases} X2=40 \\ Y2=110 \end{cases} \begin{cases} X3=210 \\ Y3=40 \end{cases}$$

60 GOSUB 1200

70 REM AÍ VAI O TOPO

80 HCOLOR = 1

90 INPUT X1, Y1, X2, Y2, X3, Y3 → Para o nosso caso particular entre agora com

$$\begin{cases} X1=70 \\ Y1=40 \end{cases} \begin{cases} X2=80 \\ Y2=60 \end{cases} \begin{cases} X3=210 \\ Y3=40 \end{cases}$$

100 GOSUB 1200

110 REM AÍ VAI A PARTE DE BAIXO

120 HCOLOR = 5

130 INPUT X1, Y1, X2, Y2, X3, Y3 → Para o nosso caso entre aqui com os seguintes valores

$$\begin{cases} X1=80 \\ Y1=60 \end{cases} \begin{cases} X2=40 \\ Y2=110 \end{cases} \begin{cases} X3=230 \\ Y3=60 \end{cases}$$

140 GOSUB 1200

150 FOR Z=1 TO 499: NEXT Z

160 GOTO 20

1200 IF Y1=Y3 THEN GOSUB 1300

1210 IF X1=X2 THEN GOSUB 1400

1220 RETURN

1300 REM TEM-SE UM PARALELOGRAMO HORIZONTAL

1310 FOR X=0 TO X3-X1

1320 HPLOT X1+X, Y1 TO X2+X, Y2

```

1330 NEXT X
1340 RETURN
1400 REM TEM-SE UM PARALELOGRAMO VERTICAL
1410 FOR Y=0 TO Y2-Y1
1420 HPLLOT X1, Y1+Y TO X3,X3+Y
1430 NEXT Y
1440 RETURN

```

Elabore agora em modo de alta resolução os programas que permitam obter os gráficos das Figuras 74a), b) e c).

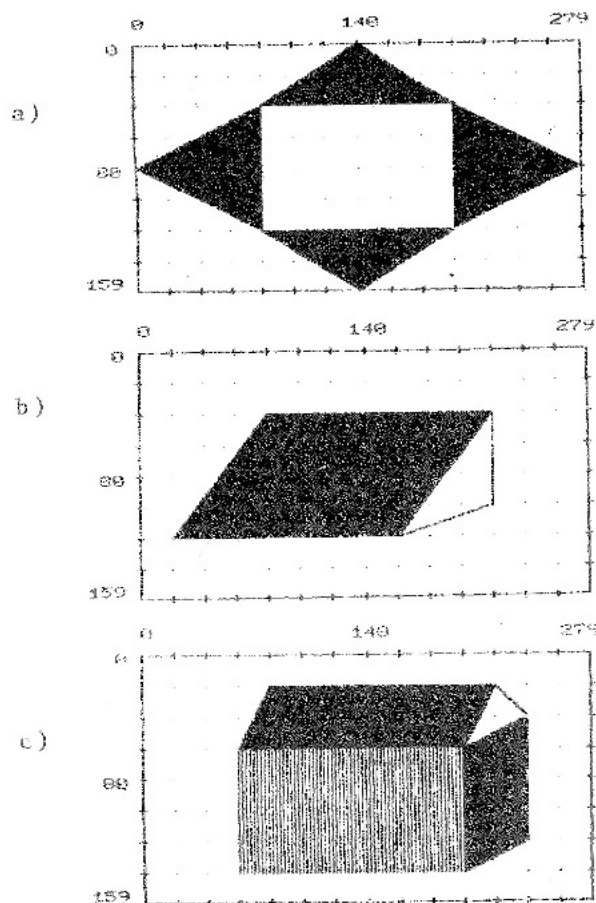


Figura 74

Divertindo-se com a geometria

a) Inicialmente vamos elaborar um programa que nos permita obter a circunferência de círculo (Figura 75).



Figura 75

usando as equações:

$$\begin{cases} x = x_c + r \cos\theta \\ y = y_c + r \sin\theta \end{cases} \quad (1)$$

onde (x,y) são as coordenadas de um ponto qualquer da circunferência de um círculo com um raio r e centro no ponto (x_c, y_c) .

- 10 HCR: HCOLOR = 5
- 20 XC = 139 : YC = 80 → centro da circunferência
- 30 R = 30 → raio da circunferência
- 40 I = 0 → valor inicial em radianos para começar o desenho da circunferência
- 50 F = 6.4 → valor final em radianos
- 60 H = 0.2 → incremento que é usado como passo enquanto se desenha

```

70 X1 = XC + R : Y1 = YC
80 FOR A = I TO F STEP H
90 X2 = R*COS(A) + XC
100 Y2 = -R*SIN(A) + YC
110 HPLOT X1,Y1 TO X2,Y2
120 X1=X2 : Y1 = Y2
130 NEXT A
140 END

```

Caso você queira desenhar várias circunferências de círculo concêntricas basta fazer as seguintes modificações

```

30 FOR R = 10 TO 50 STEP 5
135 NEXT R

```

Que tal no lugar de várias circunferências termos várias elipses com o eixo maior na direção vertical?

Bem, para isto basta fazer a seguinte mudança:

```
100 Y2 = -1.35 * R * SIN(A) + YC
```

Que tal agora ter elipses com o eixo maior no eixo horizontal?

Para isto basta fazer a seguinte mudança:

```
90 X2 = 1.55*R*COS(A) + XC
```

O.I. Para notar bem essa elipse deixe a linha 100 como estava no início.

b) Vamos desenhar agora uma linda flor com o auxílio das funções seno e cosseno (veja a Figura 76).

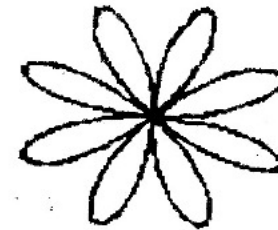


Figura 76

```

10 HGR: HCOLOR = INT(7*RND(1))+1
20 INPUT N,A,PI → Entre aqui com os valores N=4;
                    A=55; PI = 3.141592
30 M=N*PI*2
40 X1=139: Y1=80
50 REM AÍ COMEÇA O DESENHO DA LINDA FLOR
60 FOR T=0 TO M STEP .1
70 R = A*SIN(N*T)
80 X2=R*COS(T) + 139
90 Y2 = -R*SIN(T)+80
100 HPLOT X1,Y1 TO X2,Y2
110 X1=X2 : Y1=Y2
120 NEXT T
130 FOR Z=1 TO 499 : NEXT Z
140 GOTO 10

```

O.I. Se N é par tem-se uma "flor" de $2*N$ pétalas como é o caso da Figura 76 e caso N seja ímpar a flor terá N pétalas!!!

Não está acreditando nisso?

Então entre com diversos valores de N na linha 20 (N = 3,4,5,6 etc.).

Na linha 70 utiliza-se uma equação chamada "rosa de N folhas" para obter um valor para R.

Você caro(a) leitor(a) deve notar que o valor de R muda toda vez que o laço FOR-NEXT é executado.

Aí vai o "racha a moringa" baseado nos programas "DIVERTINDO-SE COM A GEOMETRIA".

Elabore o programa que permita obter o desenho de:

- a) um triângulo com um vértice obrigatoriamente em (13,13);
- b) sete triângulos retângulos concêntricos;
- c) um quadrado com um vértice obrigatoriamente em (77,55) e lado 30;
- d) 13 quadrados concêntricos;
- e) um polígono regular de n lados;
- f) uma circunferência, fazendo o n do item e) ter um valor bem grande.

Espiraís

No programa ESPIRAIS você poderá manipular o tamanho inicial (R), o incremento de cada lado (D) e a cor desejada (C) e obter lindas espirais como a da Figura 77.

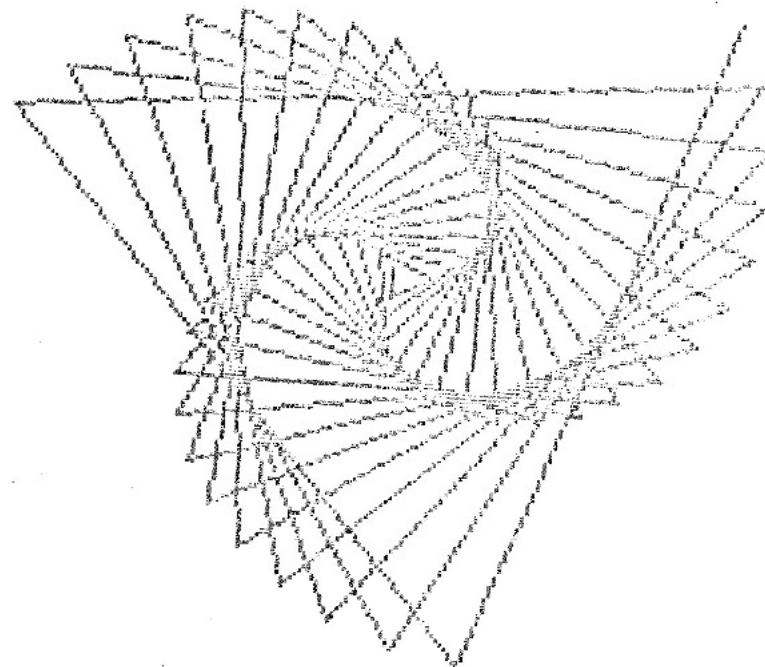


Figura 77

```

10 TEXT: HOME
20 INPUT R, D, C, N
30 INPUT "ENTRE COM O ÂNGULO EM GRAUS"; A
40 HGR
50 HCOLOR = C
60 A = A*(3.141592/180)
70 X1 = 110 : Y1 = 70
80 I = 0
90 X2 = R*COS(A*I)+X1
100 Y2 = -R*SIN(A*I)+Y1
110 IF X2 < 0 OR X2 > 279 THEN 180
120 IF Y2 < 0 OR Y2 > 159 THEN 180
125 IF I > 10 * N THEN 180
130 HPLOT X1,Y1 TO X2,Y2
140 SOUND 96,8 TO 85,8 TO 76,8
145 SOUND 128,8 TO 114,8 TO 108,2
150 X1 = X2 : Y1 = Y2
160 R = R + D
170 I = I + 1
180 FOR Z = 1 TO 888 : NEXT Z
190 GOTO 10

```

tamanho inicial
 incrementa o lado
 cor desejada
 regula as voltas

converte graus em radianos

aqui determina-se o valor das próximas coordenadas

com essas comparações para-se o programa quando a imagem representada atingir os limites da tela

um pouco de "música" enquanto se desenha

Com essa pausa "congela-se" a imagem durante um certo tempo

Aí vai um conjunto de "probleminhas" para "fundir a cuca".

a) Que tal você fazer um programa que permita um pássaro voar sobre uma flor (Figura 78).

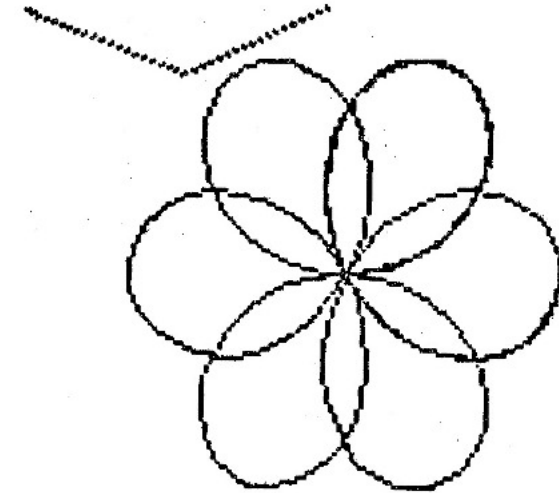


Figura 78

b) Que tal você fazer uma onda que mude de posição e um peixinho dando saltos (veja a Figura 79) em arco.

Você é capaz de desenhar mais de um peixinho?



Figura 79

c) Use as equações da elipse para representar esquematicamente uma macieira.

Coloque na mesma 14 maçãs das quais 6 já caídas no chão (veja a Figura 80).

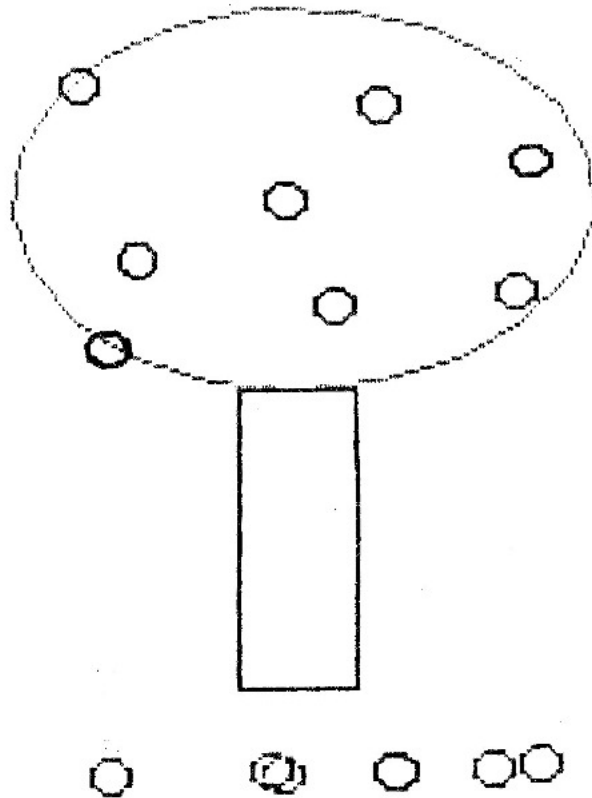


Figura 80

O.I. - Está na hora de você pensar seriamente em adquirir uma GRAFIX para não apenas ver os gráficos, mas poder guardar os mesmos e com isto completar tanto os seus trabalhos escolares assim como os relatórios de serviço.

Vá a página 189 e conheça melhor a GRAFIX.

21

Movimentando a ponta de uma seta

Vamos agora elaborar um programa que desenhe a ponta de uma seta (Figura 81).

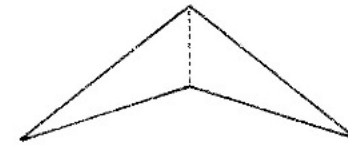


Figura 81

O programa é interativo permitindo independentemente o controle dos fatores de escala para X e Y assim como a possibilidade de aplicar a rotação e/ou a translação a ponta da seta.

Um dos métodos mais simples de guardar os dados de uma figura ou forma desejada é através de variáveis coletivas.

Aí vai o trecho do programa que permite is

```

30 READ N
40 DIM X(N), Y(N)
50 FOR I=1 TO N : READ X(I) : NEXT I
60 FOR I=1 TO N : READ Y(I) : NEXT I

```

```

1000 DATA 5
1100 DATA 0, 4, 2, 4, 0
1200 DATA 0, 4, 0, -4, 0

```

Será que realmente temos a ponta de uma flecha?

Para se ter certeza disto basta representar os pontos (x_i, y_i) em um papel quadriculado e unir os mesmos.

Veja a confirmação na Figura 82.

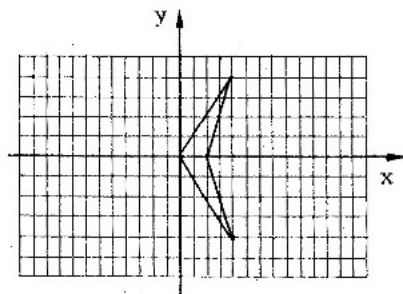


Figura 82

O.I. Cuidado que a orientação dos eixos na Figura não é a mesma do seu TK-2000 COLOR

É evidente que para que a forma em questão esteja na tela é necessário geralmente adicionar um valor, digamos PX, a todos os valores X (abscissas) da forma (figura) e um PY para todas as ordenadas Y.

Uma coisa o(a) leitor(a) deve ter sempre em mente, para evitar que apareçam quantidades ilegais, que PX e PY devem seguir dois critérios.

- 1) PX e PY precisam ser ao menos maiores que os respectivos valores negativos de X e Y.
- 2) PX e PY quando somados respectivamente aos valores positivos de X e Y não podem superar os limites da tela que como você sabe no caso de HGR são de 279 para X e 159 para Y.

```

100 PRINT "VOCÊ QUER DESLOCAR A SUA FIGURA PARA ALGUM PONTO
      DA TELA OU SEJA FAZER UMA TRANSLAÇÃO (S/N)?"
110 INPUT R$
120 IF R$ = "S" THEN 150
130 PRINT "VOCÊ QUER MUDAR OS FATORES DE ESCALA FX E FY PA-
      RA X E Y (S/N)?"
133 INPUT Q$
135 IF Q$ = "S" THEN 230
140 GOTO 310
150 INPUT PX, PY
160 X1 = X(1) + PX : Y1 = Y(1) + PY
170 HGR : HCOLOR = 5
175 FOR I=2 TO N
180 X2 = X(I) + PX : Y2 = Y(I) + PY
190 HPLOT X1, Y1 TO X2, Y2
200 X1 = X2 : Y1 = Y2
210 NEXT I
220 GOTO 340
230 INPUT FX, FY
240 FOR I = 1 TO N
250 X(I) = FX * X(I)
260 Y(I) = FY * Y(I)
270 NEXT I
280 PRINT "VOCÊ QUER FAZER AGORA UMA TRANSLAÇÃO DA SUA FIGU-
      RA PARA UM PONTO PX, PY DA TELA (S/N)?"
290 INPUT S$
300 IF S$ = "S" THEN 150
310 PRINT "VOCÊ GOSTARIA DE APLICAR UMA ROTAÇÃO DE ÂNGULO AR
      A SUA FIGURA ANTES DE EFETUAR A TRANSLAÇÃO (S/N)?"
315 INPUT T$
320 IF T$ = "S" THEN 240
330 END
340 INPUT "QUAL É O ÂNGULO DE ROTAÇÃO (GRAUS)?: "; AR

```

```

350 AR = AR * 3.141592/180 → transformação de graus para
                             radianos
360 INPUT "ENTRE COM O PONTO (RX, RY) EM TORNO DO QUAL SERÁ
FEITA A ROTAÇÃO"; RX, RY
370 FOR I = 1 TO N
380 TEMP = X(I) * COS(AR) - Y(I) * SIN(AR) + RX
390 Y(I) = X(I) * SIN(AR) + Y(I) * COS(AR) + RY
400 X(I) = TEMP
410 NEXT I
420 PRINT "NÃO ESQUEÇA DE ENTRAR COM PX E PY"
430 PRINT
440 GOTO 150

```

Para a visualização do programa faça inicialmente apenas uma translação com $PX = 130$ e $PY = 75$.

A seguir aplique inicialmente a mudança de escala $FX = 2.5$ e $FY = 4.5$ e depois faça a translação para $PX = 120$, $PY = 125$.

Finalmente aplique uma mudança de escala $FX = 4$ e $FY = 6$, uma rotação de 45° graus em torno do ponto $RX = 8$, $RY = 10$ e a seguir faça uma translação da figura obtida para $PX = 70$, $PY = 80$.

Como sugestão de figuras para você "brincar" de criativo(a) programador(a) aí vão algumas:

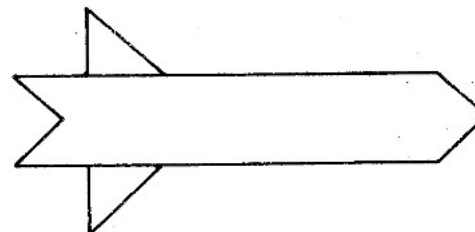


Figura 83

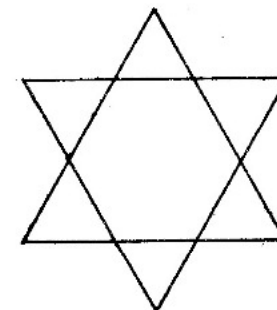


Figura 84

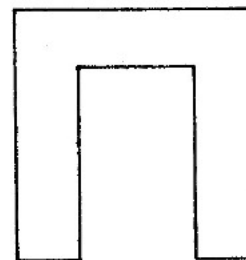


Figura 85

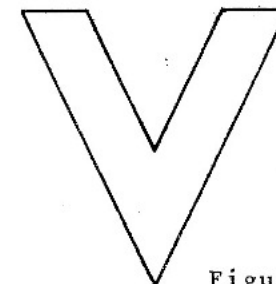


Figura 86

A escolha das coordenadas fica por sua conta porém para auxiliá-lo aí vão duas folhas quadriculadas (sem uso ainda) para os gráficos de alta resolução no seu TK-2000 COLOR.

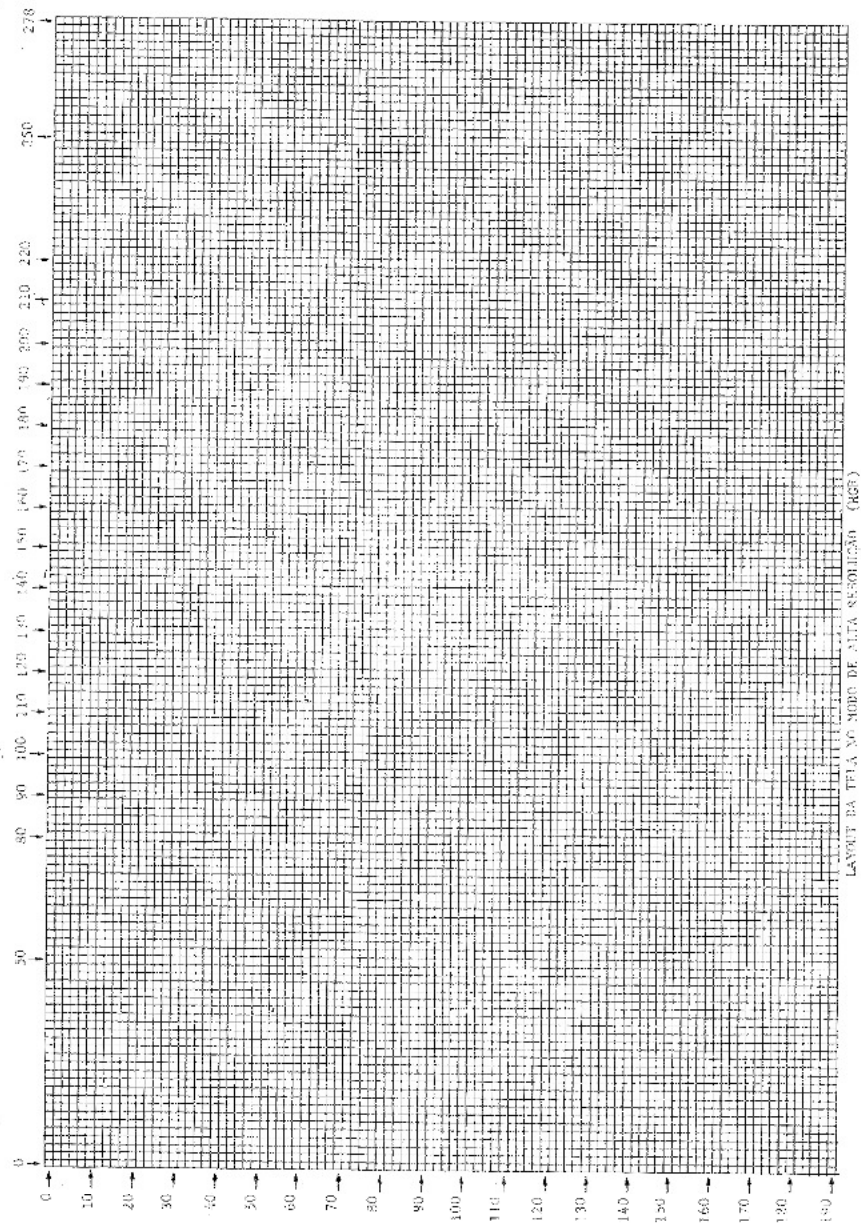


Figura 87

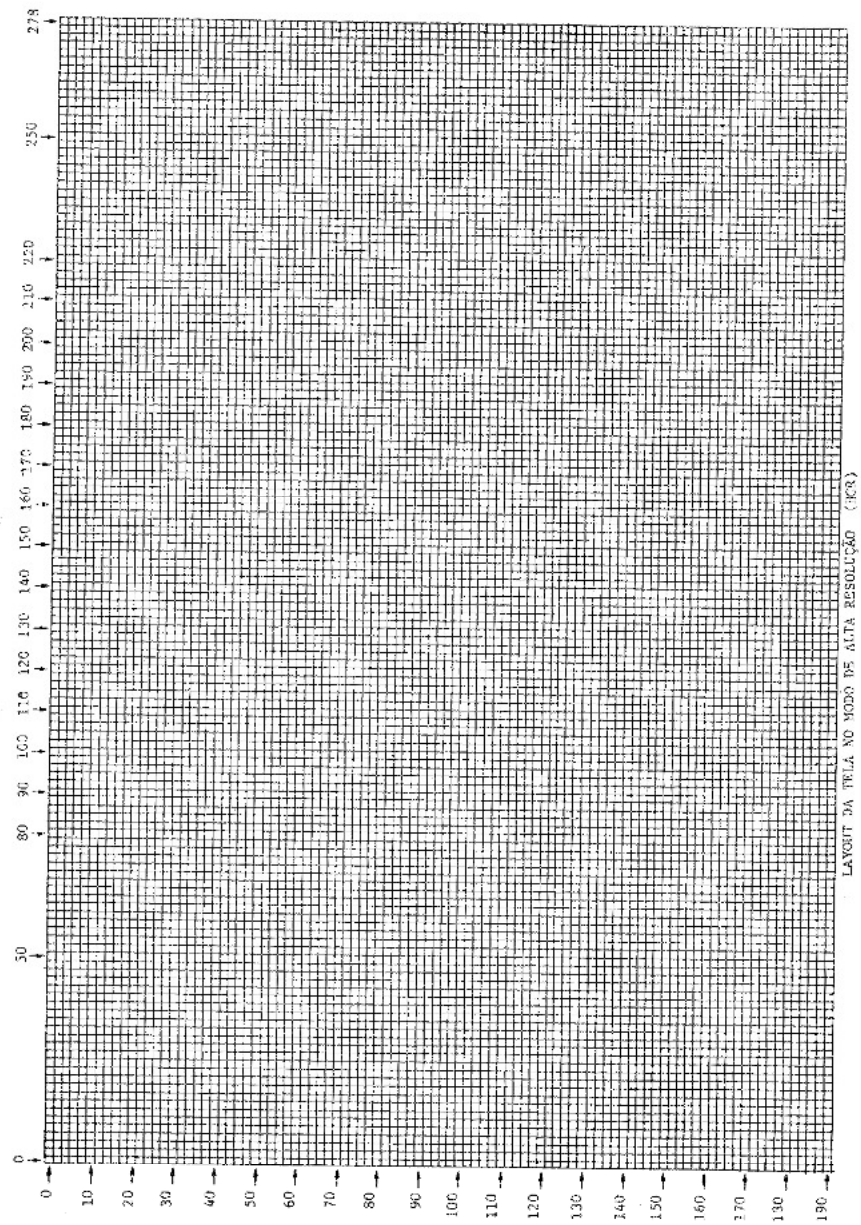


Figura 88

O.I. - Recentemente estivemos participando de um estágio na Pennsylvania, na PennState University e lá conhecemos o simpático mestre inglês Richard Devon.

Além de conhecermos através do prof. Devon todas as novidades da linguagem BASIC dentro de microcomputadores da linha APPLE recebemos também o seu livro "The First Few Bytes".

O programa "MOVIMENTANDO A PONTA DE UMA SETA" é uma adaptação de um programa que existe no livro do prof. Richard Devon.

Aliás o mesmo ocorre com o programa seguinte ao qual denominamos de "VOANDO".

O prof. Devon usa muito as transformações lineares nos gráficos tanto a duas como a três dimensões.

No caso de duas dimensões para qualquer ponto com coordenadas (X,Y) um novo ponto (X',Y') pode ser "criado" como uma função do ponto original (X,Y) e de constantes numéricas.

As duas equações gerais para essa finalidade são:

$$\begin{cases} X' = A.X + B.Y + C & \longrightarrow (1) \\ Y' = D.X + E.Y + F & \longrightarrow (2) \end{cases}$$

Assim todas as transformações lineares de (X,Y) para (X',Y') podem ser obtidas usando-se convenientes valores de A,B,C,D,E,F nas equações (1) e (2).

Translação

$$\begin{cases} \text{Em X} & \rightarrow X' = X+C & \rightarrow A=1, & B=0 & 0 < C < 279 \\ \text{Em Y} & \rightarrow Y' = Y+F & \rightarrow D=0, & E=1 & 0 < F < 159 \end{cases}$$

Mudança de escala usando os fatores de escala FX e FY

$$\begin{cases} \text{Em X} & \rightarrow X' = X.FX & \rightarrow A=FX, & B=0, & C=0 \\ \text{Em Y} & \rightarrow Y' = Y.FY & \rightarrow D=0, & E=FY, & F=0 \end{cases}$$

Rotação de um ângulo T

$$\begin{cases} X' = X.COS(T) - Y.SEN(T) \\ Y' = X.SEN(T) + Y.COS(T) \end{cases} \rightarrow \begin{cases} A=COS(T), & B=-SEN(T), & C=0 \\ D=SEN(T), & E=-COS(T), & F=0 \end{cases}$$

↳ Aqui não está em BASIC não é?

Reflexão ou rebatimento

$$\begin{cases} \text{Em torno do eixo OY} & \rightarrow X' = -X & \rightarrow A=-1, & B=0, & C=0 \\ \text{Em torno do eixo OX} & \rightarrow Y' = -Y & \rightarrow D=0, & E=-1, & F=0 \end{cases}$$

Cisalhamento

$$\begin{cases} \text{Cisalhando Y sobre X} & \rightarrow X' = X + Y.CY & \rightarrow A=1, & B=CY, & C=0 \\ \text{Cisalhando X sobre Y} & \rightarrow Y' = Y + X.CX & \rightarrow D=CX, & E=1, & F=0 \end{cases}$$

Observações finais:

- 1) Para a rotação e para o cisalhamento quando ambos os cisalhamentos estiverem presentes é necessário usar uma variável temporária ao se escrever um programa para a transformação.

Por exemplo

$$\begin{cases} TEMP = X + Y.CY \\ Y' = X.CX + Y \\ X' = TEMP \end{cases} \rightarrow \text{(veja a linha 380 do programa "MOVIMENTANDO A PONTA DE UMA SETA")}$$

- 2) Embora o tratamento acima seja para um único ponto, uma figura ou forma pode consistir de um conjunto de tais pontos e obriga-se todos esses pontos a uma transformação sistemática.

Uma forma geral de se apresentar uma rotina para essa finalidade é:

```
510 INPUT "A,B,C,D,E,F"; A,B,C,D,E,F
520 FOR I=1 TO N
530 TEMP = A * X(I) + B * Y(I) + C
540 Y(I) = D * X(I) + E * Y(I) + F
550 X(I) = TEMP
560 NEXT I
```

- 3) O par de equações gerais para as transformações lineares pode ser expresso de uma forma mais simples com o auxílio da álgebra matricial, isto é:

$$\begin{bmatrix} X' \\ Y' \\ 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \begin{bmatrix} A & D & 0 \\ B & E & 0 \\ C & F & 1 \end{bmatrix}$$

- 4) As equações das transformações nos gráficos tridimensionais são

$$X' = A.X + B.Y + C.Z + D$$

$$Y' = E.X + F.Y + G.Z + H$$

$$Z' = I.X + J.Y + K.Z + L$$

Damos detalhes sobre o uso de coordenadas homogêneas (observação 3) e de gráficos tridimensionais no nosso livro "TK-2000 nos gráficos e na música".

Você está agora convidado a efetuar ou a comandar o vôo do avião mostrado na Figura 89.

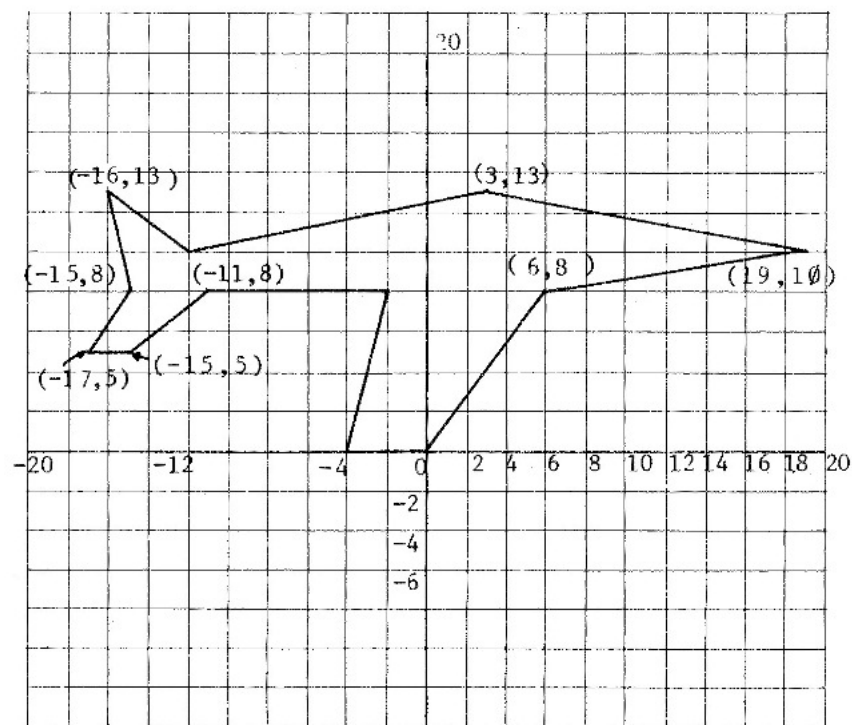


Figura 89

Aí vai o programa:

```

5 DIM X(13), Y(13)
10 HGR2
20 GOSUB 1000
30 REM AQUI COMEÇA A DECOLAGEM
40 FOR I=20 TO 260 STEP 20
50 V = I ^ 2 / 500
60 V = 185 - V
70 DECLIV = I/250
80 HCOLOR = 7
90 GOSUB 600
100 FOR K=1 TO 199 : NEXT K → essa é uma pequena pausa
    para ver o avião
110 HCOLOR = 0
120 GOSUB 600
130 NEXT I
140 REM AGORA VAI COMEÇAR A ATERRISSAGEM
150 TEXT : HOME
160 PRINT "VELOCIDADE DE DESCIDA : APERTE UM NÚMERO DE 1
    (DEVAGAR) ATÉ 4 (RÁPIDO)"
170 GET S → Com a instrução GET aceita-se um único caractere
    do teclado sem mostrar o mesmo na tela.
180 HGR2
190 DV = 10 * S
200 FOR P = 260 TO 20 STEP - DV
210 V = 185 - P ^ 2 / 500
220 I = 279 - P
230 DECLIV = - P/250
240 HCOLOR = 5
250 GOSUB 600
260 FOR Z=1 TO 299 : NEXT Z → uma nova pausa para ver o
    avião descendo
270 IF P = 20 GOTO 444
280 HCOLOR = 0

```

```

290 GOSUB 600
300 NEXT P
440 END
600 REM: SUBROTINA REPRESENTANDO A FORMA
610 HPLOT I,V
620 X1=I : Y1=V
630 FOR L=2 TO 13
640 X2=I+X(L) : Y2=V+Y(L) - DECLIV*X(L)
650 HPLOT X1,Y1 TO X2,Y2
660 X1=X2 : Y1=Y2
670 NEXT L
680 RETURN
1000 REM É AQUI QUE VÃO OS DADOS PARA O AVIÃO
1010 FOR I=1 TO 13 : READ X(I) : NEXT I
1020 FOR I=1 TO 13 : READ Y(I) : Y(I) = -Y(I) : NEXT I
1030 DATA 0, -4, -2, -11, -15, -17, -15, -16, -12, 3, 19,6,0
1040 DATA 0, 0, 8, 8, 5, 5, 8, 13, 10, 13, 10, 8, 0
1060 RETURN

```

Nesse programa inicialmente tomam-se as coordenadas dos pontos para desenhar o avião (veja a Figura 89 e as linhas do programa 1000 - 1050).

Continuamente a imagem do avião é representada e depois apagada.

Para isto se usam as instruções 110 e 280 onde aparece o comando HCOLOR = 0.

Na linha 1020 as ordenadas y são invertidas pois o TK-2000 COLOR mede o y para baixo.

Isto simplesmente ajusta a figura e não a sua posição na tela.

A posição na tela é ajustada com o auxílio das instruções 60 e 210.

Como se pode notar facilmente a trajetória do vôo do avião é uma parábola.

$$Y = X^2 / 500 \quad (\text{linha } 50)$$

O declive em qualquer ponto de $y = x^2/500$ é $\frac{dy}{dx} = \frac{x}{250}$ (veja as instruções de números de linha 70 e 230).

O ajuste na linha 64 para a representação da figura na declividade conveniente é uma aproximação grosseira.

É por isso que você verá (ou já viu...) uma grande distorção na forma do seu avião a medida que a declividade aumenta.

Caso a figura fosse desenhada corretamente os cálculos necessários para isso, tornariam o movimento de desenho tão vagaroso que se perderia o efeito do movimento do avião.

Aí vai um "problema" (aliás aqui são dois problemas pois o primeiro é que problema não é com r...).

Que tal você desenhar um outro tipo de avião e fazer com que ele ande segundo uma outra trajetória parabólica ou uma outra trajetória que no início seja ascendente e depois descendente.

Cuidado para não cair fora da tela!!!?!!!

Fogos de artifícios e figuras de Lissajou

a) Inicialmente vamos elaborar um programa que permita a obtenção dos fogos de artifício da Figura 90.

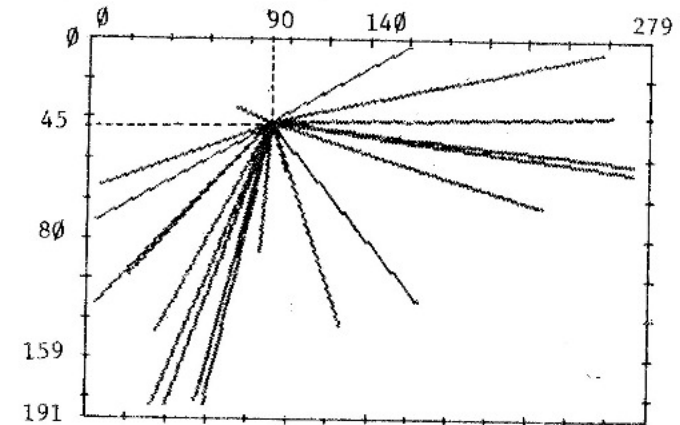


Figura 90

```

10 REM FOGOS DE ARTIFÍCIO
20 FOR I=1 TO 7
30 HGR2
40 X0 = 90 + INT(80 * RND(1)) : Y0 = 45 + INT(90 * RND(1))
50 FOR A=1 TO INT(199 * RND(1))
60 HCOLOR = INT(7 * RND(1))
70 X = INT(280 * RND(1)) : Y = INT(191 * RND(1))
80 HPLOT X0, Y0 TO X, Y
90 SOUND 96, 30 TO 47, 30 TO 23, 60 TO 67, 8 TO 33, 8 TO 56,
240

```

veja a página 179 do seu manual de TK-2000 COLOR


```

100 NEXT A
110 NEXT I
120 FOR Z=1 TO 399 : NEXT Z
130 TEXT : HOME
140 PRINT "VOCÊ QUER OUTRAS EXPLOSÕES (S/N)?"
150 INPUT R$
160 IF R$ = "S" THEN 20
170 END

```

- b) Os engenheiros eletricitistas obtêm os gráficos de Lissajous como aqueles da Figura 91 aplicando dois sinais senoidais de frequências diferentes nas "entradas" X e Y de um osciloscópio. (Você sabe o que é um osciloscópio?).

Esses sinais controlam o movimento vertical e horizontal da imagem.

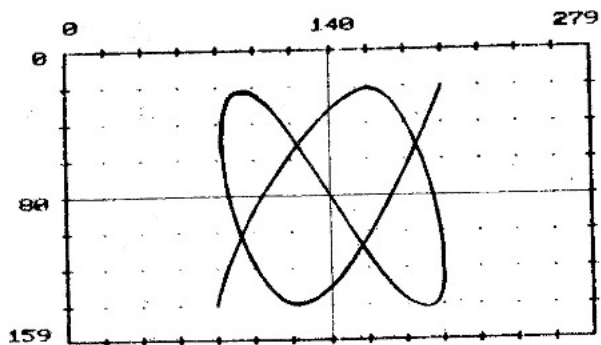


Figura 91

```

b) 10 TEXT : HOME
20 HTAB 10 : PRINT "FIGURAS DE LISSAJOUS"
30 VTAB 7 : HTAB 7
40 INPUT "ENTRE COM A RAZÃO HARMÔNICA: ISTO É DOIS
NÚMEROS H E V"; H,V
50 HGR

```

```

60 HCOLOR = 5
70 X1 = 140 : Y1 = 80
80 FOR K=0 TO 6.35 STEP .05
90 X = 140 + 80 * SIN(H*K)
100 Y = 80 + 80 * SIN(V*K)
110 HPLLOT X1,Y1 TO X,Y
120 X1 = X : Y1 = Y
130 NEXT K
140 VTAB 24
150 PRINT "QUER OUTRA FIGURA (S/N)?"
160 GET R$
170 IF R$ = "S" THEN 20
180 END

```

Trabalhinho intelectual para você obnubila do(a) leitor(a):

- a) Elabore um programa que desenhe uma série de retângulos conforme mostrado na Figura 92.

Note que os vértices de cada conjunto de retângulos deve estar sobre a mesma circunferência de círculo.

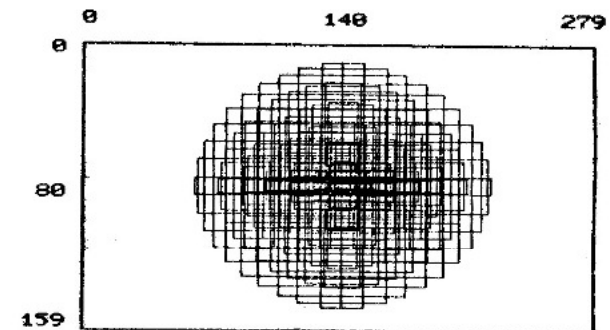


Figura 92

- b) Elabore um programa que desenhe um conjunto de estrelas de seis pontas posicionadas aleatoriamente na tela como indicado na Figura 93.

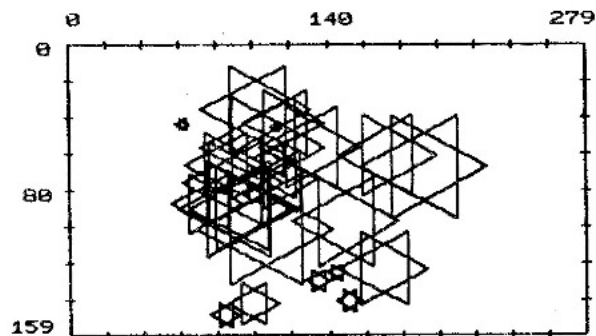


Figura 93

- c) Elabore um programa que permita obter uma espiral como a mostrada na Figura 94.

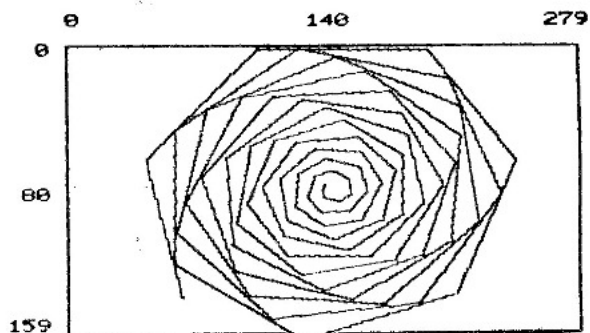


Figura 94

O.l. Caso queira pode fazer também todos esses programas em HGR2!!! Gostou da idéia?.

Além disto é bom guardá-los para mostrar aos amigos e isto só é possível se você tiver uma GRAFIX...

Orbitando

Com o auxílio desse programa você vai ter uma visão animada de satélites em órbita ou elétrons movimentando-se em torno de um núcleo ou qualquer coisa girando em torno de um centro em trajetórias que podem ser circunferências ou elipses.

No programa "ORBITANDO", no centro temos o Sol, o núcleo ou qualquer coisa que "atraia" e "rodando" a sua volta pode-se imaginar que estão os satélites elétrons ou quaisquer outros tipos de objetos.

Esses objetos são representados por pequenos quadrados que rodopiam em torno do seu eixo à medida que fazem o movimento de rotação em torno de um centro (que é o centro da tela).

O programa "ORBITANDO" irá lhe perguntar qual é o número de objetos orbitando, a velocidade e o tamanho de cada um, assim como pede dois números que definem a órbita de cada trajetória.

Quando a órbita é circular ambos os números são iguais ao raio da órbita.

Quando a órbita é uma elipse os números são os comprimentos dos semi-eixos horizontal e vertical.

Tem-se uma saída bem interessante quando se especificarem 5 objetos com velocidade, tamanho e valores dos eixos nos seguintes intervalos:

velocidade angular \longrightarrow (2 - 11)
 tamanho \longrightarrow (2 - 7)
 eixos $\left\{ \begin{array}{l} a = \text{horizontal} \quad (15 - 120) \\ b = \text{vertical} \quad (10 - 70) \end{array} \right.$

O programa usa um sistema de coordenadas com a sua origem no centro da tela.

Ele calcula as coordenadas de um objeto em órbita a partir do sistema:

$$\begin{cases} x = a \cos\theta \\ y = b \sin\theta \end{cases} \longrightarrow (1)$$

onde a e b são os números que serão pedidos na entrada para definir a órbita e o ângulo θ é aumentado de uma quantidade proporcional a velocidade do objeto, para cada ciclo de movimento ou animação.

Aliás, isto é aproximadamente correto para planetas movendo-se em torno do Sol e luas movendo-se em torno de planetas com órbitas que são *levemente elípticas*.

Contudo para uma maior precisão e principalmente para cometas, o Sol (centro de atração) deveria estar não no centro da tela e sim no foco de uma elipse.

Caso você caro(a) leitor(a) não esteja habituado com algum termo recorra ao seu dicionário...

No programa "ORBITANDO" vamos ter algo semelhante ao mostrado na Figura 95.

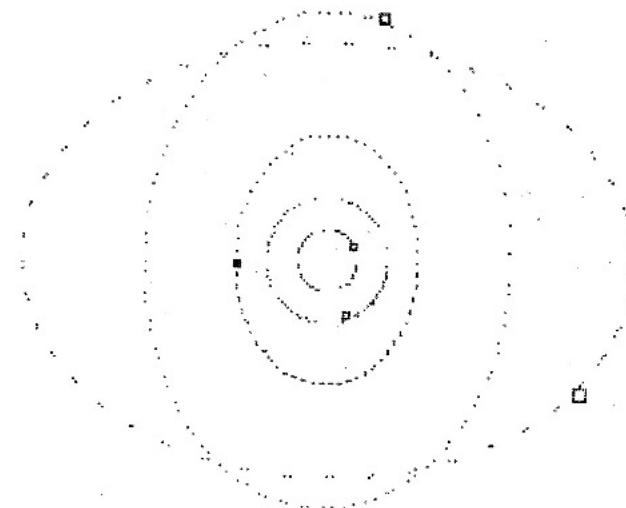


Figura 95

As variáveis principais do programa são:

- $T(2,314)$ \longrightarrow variável coletiva bidimensional que contém os valores de senos e cossenos necessários para calcular as posições dos objetos em órbita.
- $N0$ \longrightarrow número de objetos em órbita
- K \longrightarrow índice ou variável contadora $K=1,2,\dots,N0$
- $\left\{ \begin{array}{l} XE(K) \longrightarrow \text{comprimento do semi-eixo horizontal do } k\text{-ésimo objeto} \\ YE(K) \longrightarrow \text{comprimento do semi-eixo vertical do } k\text{-ésimo objeto} \end{array} \right.$
- $\left\{ \begin{array}{l} XN(K) \longrightarrow \text{coordenada } x \text{ do } k\text{-ésimo objeto} \\ YN(K) \longrightarrow \text{coordenada } y \text{ do } k\text{-ésimo objeto} \end{array} \right.$
- $\left\{ \begin{array}{l} X0(K) \longrightarrow \text{valor anterior de } XN(K) \\ Y0(K) \longrightarrow \text{valor anterior de } YN(K) \end{array} \right.$
- $AN(K)$ \longrightarrow ângulo de rotação do k -ésimo objeto em torno do centro de atração.

RN(K) → ângulo de rotação do k-ésimo objeto em torno do próprio eixo

RØ(K) → valor anterior de RN(K)

No programa "ORBITANDO" inicialmente se armazenam os códigos para um pequeno quadrado como uma tabela de forma (figura) e calcula-se uma tabela de senos e cossenos para permitir que o movimento seja feito de forma rápida.

A seguir pede-se o número de objetos em órbita e suas propriedades e aí então começa o movimento "sem fim"...

Nesse movimento cíclico interno, AN(K) é aumentado pela velocidade angular W(K) para se obter uma nova posição angular do k-ésimo objeto em unidades que correspondem aos sucessivos elementos de T().

O ângulo de rotação em torno do próprio eixo RN(K), é também "influenciado" pela velocidade angular W(K).

As coordenadas atuais são calculadas por fórmulas dadas anteriormente [sistema (1)].

O objeto é desenhado na sua nova posição e orientação pelo uso dos comandos ROT e DRAW e é apagada a sua posição imediatamente anterior.

```
1Ø REM ***ORBITANDO***
2Ø LOMEM : 16384 → O comando LOMEM é usado para definir o limite inferior da área utilizável pelo BASIC
3Ø DIM T(2,314)
4Ø HGR: HCOLOR=5 : POKE 34,2Ø : HOME : PRINT "***POR FAVOR ESPERE ESTOU FAZENDO CÁLCULOS***"
```

```
45 SCALE = 7 : ROT = Ø : J = Ø
47 REM NESSE LOOP SE ARMAZENA A TABELA DA FIGURA OU FORMA
5Ø FOR I = 7676 TO 7682
52 READ V
54 POKE I,V
56 NEXT I
58 DATA 1, Ø, 4, Ø, 44, 62, Ø
6Ø REM AQUI SE ARMAZENA O ENDEREÇO DA BASE
65 POKE 232, 252
7Ø POKE 233, 29
8Ø FOR A = Ø TO 6.28 STEP .Ø2
9Ø T(1,J) = COS (A) : T(2,J) = SIN(A) : J = J+1
1ØØ NEXT A
11Ø HOME: INPUT "QUANTOS OBJETOS EM ÓRBITA DESEJA?";NØ
12Ø IF NØ > 1Ø THEN PRINT "NÃO DEVEM SER MAIS DO QUE 1Ø":
NØ = 1Ø
13Ø FOR K = 1 TO NØ
14Ø DIM XE (NØ), YE(NØ), W(NØ), G(NØ)
15Ø DIM XN(NØ), YN(NØ), AN(NØ), AØ(NØ), RN(NØ)
16Ø DIM RØ(NØ), XØ(NØ), YØ(NØ)
17Ø FOR K = 1 TO NØ
18Ø HOME : PRINT "OBJETO"; K; "DE"; NØ
185 INPUT "ENTRE COM OS COMPRIMENTOS DOS SEMI-EIXOS=";XE(K),
YE(K)
190 INPUT "ENTRE COM A VELOCIDADE ANGULAR E TAMANHO =";
W(K), G(K)
195 NEXT K
197 HOME
2ØØ FOR K = 1 TO NØ
2Ø5 RN(K) = RN(K) + W(K)
21Ø IF RN(K) > 64 THEN RN(K) = RN(K) - 64
215 AN(K) = AØ(K) + W(K)
22Ø IF AN(K) > 314 THEN AN(K) = AN(K) - 314
23Ø NEXT K
```

```

240 FOR K=1 TO N0
250 XN(K) = 140 + XE(K) * T(1, AN(K))
260 YN(K) = 80 + YE(K) * T(2, AN(K))
270 NEXT K
280 FOR K=1 TO N0
290 SCALE = G(K) : ROT = R0(K)
300 XDRAW 1 AT X0(K), Y0(K)
310 ROT = RN(K)
320 XDRAW 1 AT XN(K), YN(K)
330 HPLOT X0(K), Y0(K)
340 NEXT K
350 FOR K=1 TO N0
360 X0(K) = XN(K) : Y0(K) = YN(K)
370 R0(K) = RN(K) : A0(K) = AN(K)
380 NEXT K
390 GOTO 200

```

O.I. Não entraremos em grandes detalhes (aliás em nenhum), sobre ROT, SCALE, XDRAW e POKE.

Recomendamos consultar os nossos livros "Dê um APPLE a sua vida" ou então "TK-2000 nos gráficos e na música" onde abundam explicações sobre essas instruções. Que comercial e que saída pela tangente...

Para fazer com que a temperatura da sua "cuca" fique um pouco acima de 37°C sem que você esteja doente faça os seguintes programas:

a) Obtenha um programa que permita traçar a função de Bezier.

A expressão geral para a função de Bezier pode ser escrita na notação fatorial e o seu programa deve calcular cada função separadamente.

O mais simples é começar com

$$b(n, 0, u) = (1 - u)^n \longrightarrow (1)$$

função de Bezier

e usar a fórmula

$$b(n, i, u) = \binom{n-i+1}{i} \left(\frac{u}{1-u}\right) b(n, i-1, u) \longrightarrow (2)$$

Fazendo $i=1$ pode-se obter a função de Bezier (famoso engenheiro francês) $b(n, 1, u)$ a partir de $b(n, 0, u)$.

Por outro lado com $i=2$, calcula-se $b(n, 2, u)$ de $b(n, 1, u)$ e assim por diante até um valor genérico i .

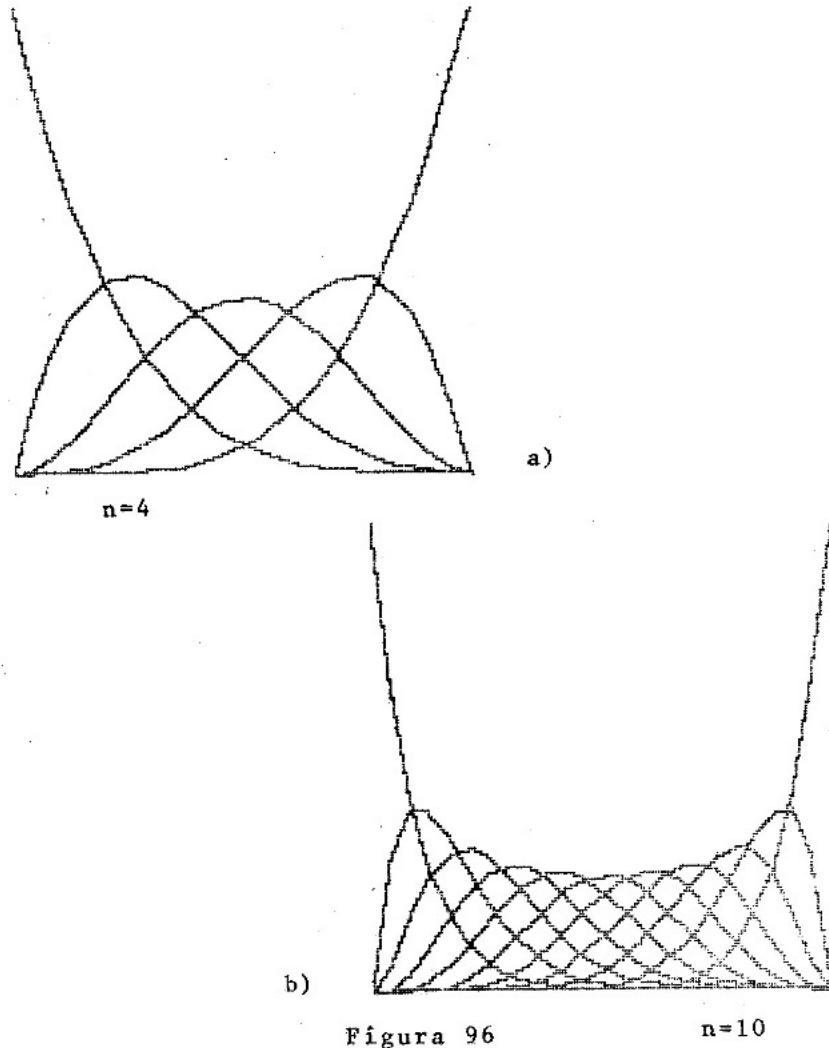
O cálculo termina quando $u=0$ ou $u=1$ pelo fato de $b(n, i, 0)$ é zero a menos que $i=0$ e $b(n, i, 1)$ é zero a menos que $i=n$.

Note também estimado(a) leitor(a) que:

$$\begin{aligned}
 n = 1 & \longrightarrow \begin{cases} b(1, 0, u) = 1-u \\ b(1, 1, u) = u \end{cases} \\
 n = 2 & \longrightarrow \begin{cases} b(2, 0, u) = (1-u)^2 \\ b(2, 1, u) = 2u(1-u) \\ b(2, 2, u) = u^2 \end{cases} \\
 n = 3 & \longrightarrow \begin{cases} b(3, 0, u) = (1-u)^3 \\ b(3, 1, u) = 3u(1-u)^2 \\ b(3, 2, u) = 3u^2(1-u) \\ b(3, 3, u) = u^3 \end{cases} \text{ etc.}
 \end{aligned}$$

É evidente que as funções de Bezier são de senhadas em eixos cujas escalas vão de 0 a 1.

Na Figura 96 você tem idéia de algumas formas das funções de Bezier.



b) Que tal você agora, após essa beleza de programa astronômico que é o "ORBITANDO", elaborar o programa para ver um Toro, ou seja uma rosquinha ou ainda um "donut" em suas várias secções?

Topou a idéia? Bom...

As fórmulas que você irá precisar não são muitas.

A bolacha "tipo polvilho" será representada por um conjunto de discos verticais de raio a com os seus centros sobre uma circunferência de círculo de raio b .

Esta última circunferência de círculo tem o seu centro na origem do sistema de coordenadas com o eixo Oy na vertical.

O seu programa deve usar os valores de a e b para se obter as coordenadas $X(I)$ e $Y(I)$ dos pontos em torno de um dos círculos, na posição local do círculo ou seja do disco.

As correspondentes coordenadas no sistema que definem uma "linha de vista" são:

$$\begin{cases} x = X(I) \cdot \cos\left(\frac{J}{2\pi}\right) \\ y = Y(I) \\ z = X(I) \cdot \sin\left(\frac{J}{2\pi}\right) \end{cases} \longrightarrow (3)$$

onde $J = 1, 2, 3, \dots, 100$

As coordenadas (x_r, y_r, z_r) do mesmo ponto em um sistema que tem o eixo Oz ao longo da linha da vista (perpendicular a tela) são:

$$\begin{cases} x_r = x \cos\theta \cos\psi + y \cos\theta \sin\psi - z \sin\theta \\ y_r = -x \sin\psi + y \cos\psi \\ z_r = x \sin\theta \cos\psi + y \sin\theta \sin\psi + z \cos\theta \end{cases} \longrightarrow (4)$$

Como se quer a projeção sobre um plano perpendicular à linha de vista basta que você, no seu programa, obtenha o par (x_r, y_r) .

"Dicas" finais

Caso conseguir elaborar o programa para $a = 15$, $b = 40$, $\theta = \frac{\pi}{3}$ e $\psi = \frac{\pi}{3}$ deve obter um desenho como o da Figura 97a e caso opte pela entrada $a = 40$, $b = 40$, $\theta = \frac{\pi}{3}$ e $\psi = \frac{\pi}{4}$ terá um desenho como o da Figura 97b.

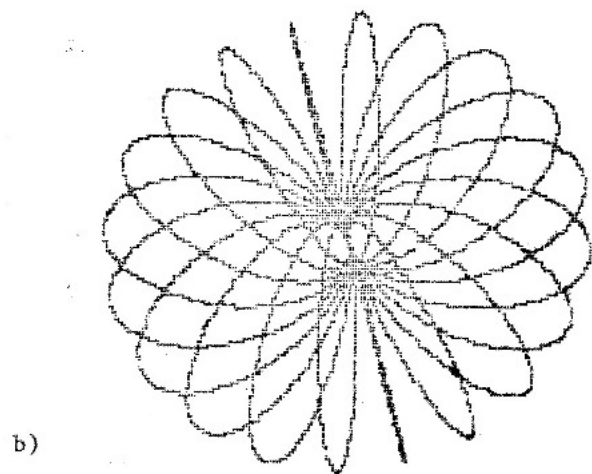
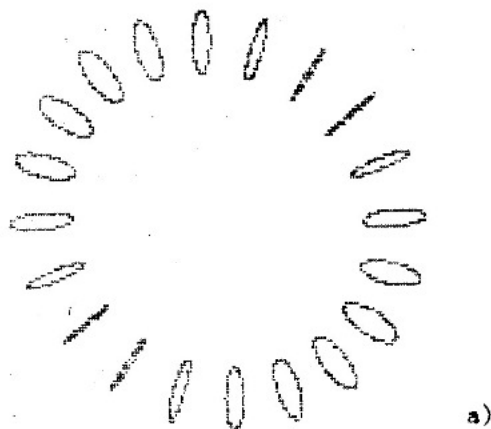


Figura 97

Rolando

Com o programa "ROLANDO" você, considerando(a) leitor(a), obterá muitas curvas interessantes.

Alguns exemplos dessas curvas estão mostradas na Figura 98 a,b,c e na Figura 99 a,b,c.

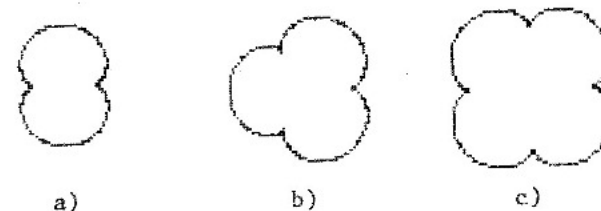


Figura 98

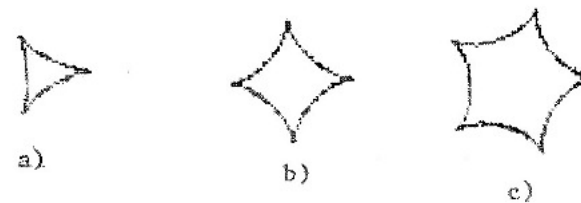


Figura 99

O programa começa lhe perguntando pela letra código E ou H para escolher o estilo geral do 1º ou do 2º conjunto de curvas e depois disso pede-se o número de cuspídes e o tamanho da curva.

Para as curvas da Figura 98a,b,c a entrada, na ordem, deve ser (E, 2,5); (E,3,5);(E,4,5).

Já para as curvas da Figura 99a,b,c a entrada deve ser, na ordem, (H,3,5); (H,4,5); (H,5,5).

O tamanho deve estar sempre no intervalo 2 a 8 pois caso contrário corre-se o perigo de cair fora da tela.

As letras E e H são as escolhidas como código pois representam respectivamente a epicicloide e hipocicloide.

Você sabe o que é epiciclóide e hipociclóide?

Em caso negativo, consulte antes um dicionário ou um livro de Geometria Analítica ou ainda um de Desenho Geométrico para estar mais "enfreado" sobre a importância das mesmas.

O programa "ROLANDO" é bem curto e ele calcula as coordenadas (x,y) dos pontos na epiciclóide relativa ao seu centro, a partir das equações:

$$\begin{cases} x = na \cos t - a \cos nt \\ y = na \sin t - a \sin nt \end{cases} \longrightarrow (1)$$

onde n é o número de cuspídes mais 1, a determina o tamanho da curva e t é um ângulo que varia de 0 até 2π .

Ter-se-á uma cardióide quando $n=2$ e uma nefróide quando $n=3$.

As correspondentes equações para a hipociclóide são:

$$\begin{cases} x = nr \cos t + a \cos nt \\ y = nr \sin t - a \sin nt \end{cases} \longrightarrow (2)$$

onde n é o número de cuspídes menos 1.

Ter-se-á uma deltóide quando $n=2$ e uma astróide quando $n=3$.

As principais variáveis nesse programa são N, A e T correspondendo a n, a e t nos sistemas de equações paramétricas (1) e (2).

Aí vai o programa:

```
10 HOME: VTAB 13
20 PRINT "TECLE E PARA EPICICLÓIDE"
30 PRINT "TECLE H PARA HIPOCICLÓIDE"
40 INPUT "ENTRE COM FIM PARA TERMINAR :";A$
50 HOME: VTAB 21
60 IF A$ = "E" THEN 100
70 IF A$ = "H" THEN 190
80 IF A$ = "FIM" THEN END
90 GOTO 10
100 REM AÍ VAI O DESENHO DA EPICICLÓIDE
110 INPUT "CUSPIDES, TAMANHO: ";N,A
120 N = N+1
125 HGR : HCOLOR=5
130 FOR T=0 TO 6.28 STEP .02
140 X = 140 + N*A*COS(T) - A*COS(N*T)
150 Y = 80 + N*A*SIN(T) - A*SIN(N*T)
160 HPLOT X,Y ————— Como você pode perceber a representação da curva será ponto por ponto
170 NEXT T
180 GOTO 10
```



```

190 REM AÍ VAI O DESENHO DA HIPOCICLÓIDE
200 INPUT "CÚSPIDES, TAMANHO:";N,A
210 N = N-1
215 HGR : HCOLOR=6
220 FOR T=0 TO 6.28 STEP .02
230 X=140 + N*A*COS(T) + A*COS(N*T)
240 Y=80 + N*A*SIN(T) - A*SIN(N*T)
250 HPLOT X,Y
260 NEXT T
270 GOTO 10

```

Pequeno raxakuka

Eu disse raxakuka e não racha a "xuxa"...

- a) Desenhe várias nefrôides aninhadas como está mostrado na Figura 100.

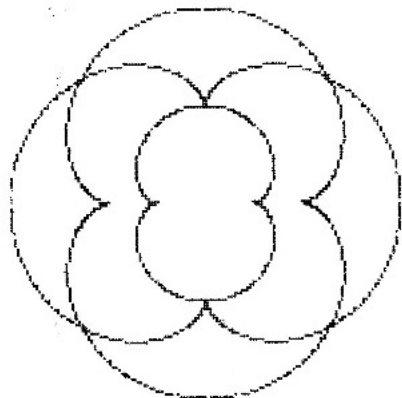


Figura 100

e para tanto use os seguintes sistemas de equações paramétricas.

$$\begin{cases} x = a(3\cos t - \cos 3t) \\ y = a(3\sin t - \sin 3t) \end{cases} \longrightarrow (3)$$

$$\begin{cases} x = \frac{a}{2}(3\cos t + \cos 3t) \\ y = \frac{a}{2}(3\sin t + \sin 3t) \end{cases} \longrightarrow (4)$$

$$\begin{cases} x = \frac{a}{4}(3\cos t - \cos 3t) \\ y = \frac{a}{4}(3\sin t - \sin 3t) \end{cases} \longrightarrow (5)$$

- b) Elabore um programa para obter a onda senoidal da figura 101.

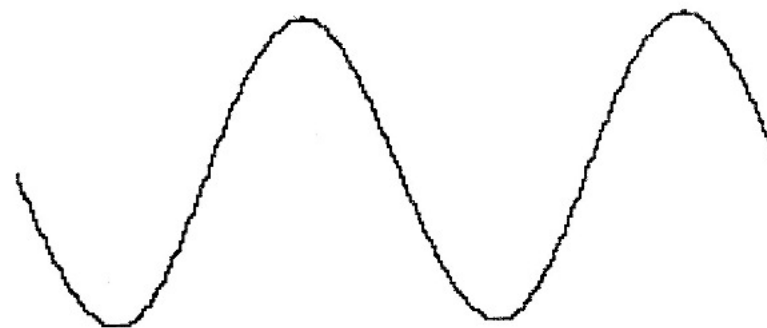


Figura 101

Use as expressões $y = \sin x$, $y = 0,5\sin x$,
 $y = 3\sin 3x$ e $y = \sin(0,5x)$.

O.I. - Pela última vez, antes de você chegar à página 189, você já pensa na possibilidade de ter ou pelo menos usar em algum lugar uma GRAFIX para dar um outro "status" aos resultados dos seus programas e principalmente para obter os mais sofisticados desenhos?

Esquiando

Aí vai uma pequena aventura nos esquis...

Um(a) esquiador(a) está se movendo para baixo entre as árvores (Figura 102).

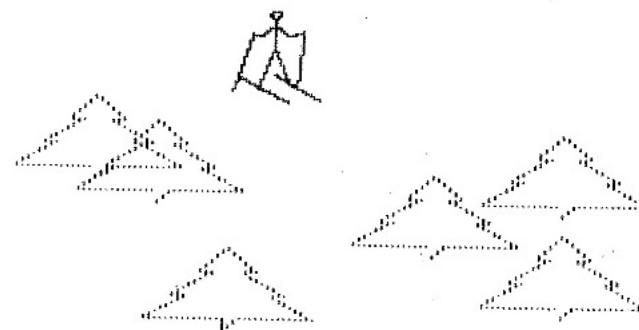


Figura 102

Veja se você consegue guiá-lo(la) sem acidentes no meio dessa floresta.

Com a tecla M você faz com que ele(ela) vá para a direita, com a tecla V para a esquerda e com a barra de espaço desloca o(a) mesmo(a) para baixo.

Caso o(a) esquiador(a) colida com uma árvore o programa para.

Boa sorte amigo(a).

Divirta-se com esse último programinha no seu TK-2000 COLOR.

```

10 DIM XA(6), YA(6)
15 TEXT : PRINT "VOCÊ IRÁ PARTICIPAR DE UMA PERIGOSA
DESCIDA - BOA SORTE!!!"
20 XI = 20 : YI = 35 : SX = 100 : P=9 : SY=20 : IC=25
30 HGR
40 HCOLOR = 6 : GOSUB 430
50 T=0
55 X=100 : Y=20 : HCOLOR=3 : GOSUB 330
60 PX=100 : PY=20
70 GOSUB 110
80 IF SY > 145 THEN 510
90 GOSUB 170
100 GOTO 70
110 REM É AQUI QUE VOCÊ COMANDA A DESCIDA
120 GET A$
125 T = T+1      ← Letra V
130 IF A$ = CHR$(86) THEN SX = SX-P
140 IF A$ = CHR$(77) THEN SX = SX+P
150 IF A$ = CHR$(32) THEN SY = SY+7
                    ← Barra de espaço em branco
160 RETURN
170 REM É AQUI QUE SE DESLOCA O(A) ESQUIADOR(A)
180 IF SX > 10 AND SX < 240 THEN 220
190 X=PX : Y=PY : HCOLOR=0 : GOSUB 330
200 IF SX < 10 THEN SX = 240
210 IF SX > 220 THEN SX = 10
220 X = PX : Y = PY : HCOLOR=0 : GOSUB 330
225 X = SX : Y = SY : HCOLOR=3 : GOSUB 330
230 PX = SX : PY = SY
240 FOR I=1 TO 6

```

```

250 IF YA(I) > (SY+IC) OR (YA(I) + YI) < SY THEN 280
260 IF (SX+IC) < XA(I) OR SX > (XA(I) + XI) THEN 280
270 GOTO 290
280 NEXT I
285 RETURN
290 REM AÍ ESTÁ A INDICAÇÃO QUE O(A) ESQUIADOR(A) COLIDIU
COM A ÁRVORE
300 HPLOT SX-13, SY+10 TO SX+8, SY+10
310 HPLOT SX-3, SY TO SX-3, SY+20
320 END
330 REM É AQUI QUE VAI A SUBROTINA PARA DESENHAR O(A)
ESQUIADOR(A)
340 HPLOT X,Y TO X+3, Y+1 TO X+2, Y+3 TO X, Y+3 TO X-1,
Y+1 TO X,Y
350 HPLOT X+1, Y+4 TO X+1, Y+15 TO X-5, Y+33
360 HPLOT X+1, Y+15 TO X+7, Y+32
370 HPLOT X-12, Y+28 TO X+6, Y+39
380 HPLOT X+1, Y+27 TO X+17, Y+38
390 HPLOT X+2, Y+7 TO X+5, Y+10 TO X+11, Y+8 TO X+9, Y+30
400 HPLOT X, Y+7 TO X-3, Y+10 TO X-7, Y+9 TO X-14, Y+34
410 HPLOT X+2, Y+7 TO X+5, Y+10 TO X+11, Y+8 TO X+9, Y+30
420 RETURN
430 REM AQUI COMEÇA A SUBROTINA PARA DESENHAR AS ÁRVORES
CUJA LOCALIZAÇÃO É DEFINIDA ALEATORIAMENTE
440 FOR J=1 TO 6
450 M = INT(210*RND(1))+3
460 N=INT(121*RND(1))+4
470 XA(J)=M : YA(J) = N
480 HPLOT M,N TO M-10, N+10 TO M-5, N+10 TO M-20, N+20
TO M-15, N+20 TO M-30, N+30 TO M-2, N+30 TO M-2, N+35
TO M+3, N+35 TO M+3, N+30 TO M+30, N+30 TO M+15, N+20
TO M+20, N+20 TO M+5, N+10 TO M+10, N+10 TO M,N
490 NEXT J
500 RETURN

```

```

51Ø TEXT: HOME: PRINT "PARABÊNS, VOCÊ CONSEGUIU EM:Ø";T;"Ø
MOVIMENTOS"
52Ø PRINT "VOCÊ QUER TENTAR OUTRA DESCIDA (S/N)?"
53Ø GET RØ
54Ø IF RØ = "S" THEN HOME ; GOTO 2Ø
55Ø GOTO 32Ø

```

0.1. ou comentários explicativos

- 1) Linha 4Ø - Chama a subrotina que desenha as 6 árvores cuja posição na tela é escolhida de forma aleatória (veja as linhas 45Ø - 46Ø).
- 2) Linha 55 - Chama a subrotina que desenha o(a) esquiador(a).
- 3) Linhas 6Ø-1ØØ - Esta parte do programa controla todo o jogo chamando as subrotinas que permitem através do teclado controlar o movimento e deslocar o(a) esquiador(a).

O programa dá voltas através dessas linhas até que o(a) esquiador(a) atinja a parte baixa da tela (o valor de SY deve ser maior que 145 como está indicado na linha 8Ø) quando se desvia para a linha 51Ø onde se dão os parabéns e o número de movimentos realizados e eventualmente você pode voltar para um novo jogo.

Cuidado que as vezes as árvores podem cair fora da tela e o jogo só começa quando temos o (a) esquiador(a) e as 6 árvores na tela.

Porém pode também ocorrer a colisão com alguma árvore antes de se atingir a parte de baixo da tela.

- 4) Linhas 11Ø-16Ø - É nesse trecho que ocorre a entrada via teclado do controle do movimento do(a) esquiador(a).

Atribui-se a variável AØ um valor que entra via teclado.

Se a tecla pressionada for V a linha 13Ø faz com que seja tirado P=9 de SX, se for pressionada a tecla M através da instrução da linha 14Ø faz-se com que seja somado a SX o valor P=9 e finalmente se a tecla pressionada for a barra para espaço adiciona-se a SY o valor 7.

Note que na linha 125 vão sendo somados o número de movimentos acumulados na variável T.

Aliás na linha 13Ø temos a função CHRØ(86) com a qual se retorna o caractere ASCII correspondendo ao valor do argumento que nesse caso é a letra V (letra que corresponde ao valor 86).

Idem para CHRØ(77) que faz retornar o M e CHRØ(32) que é para a barra de espaço.

- 5) Linhas 17Ø-285
Desloca-se com essas instruções o(a) esquiador(a) para uma nova posição.
Na linha 18Ø verifica-se a posição do(a) esquiador(a) não está fora da tela.
- 6) Linhas 25Ø-26Ø - Verifica-se aí se houve uma colisão com alguma das seis árvores.
- 7) Linhas 29Ø-31Ø - Com estas instruções desenha-se uma "cruz" (+) em cima do(a) esquiador(a) representando com isso que ele(ela) bateu.
- 8) Linhas 43Ø-5ØØ - É aí que está a subrotina com a qual se colocam 6 árvores na tela.

A descida só começa quando for possível ter as 6 árvores e como já dissemos nem sempre os números sorteados M e N permitem isto.

Trabalhinho final para você trepidante leitor(a)

Que tal você modificar o desenho das árvores tornando-as mais finas, porém aumentando o seu número?

Você ainda consegue descer são e salvo sem colisões e arranhões?

O.I. Se você se divertiu com o programa "ESQUIANDO" você realmente terá divertimento muito maior se adquirir qualquer um dos seguintes programas ou seja do software desenvolvido pela MICROSOFT.

Leia com atenção o que está escrito nas próximas quatro páginas.

Relação de software disponível para o TK-2000 COLOR...

JOGOS ANIMADOS

ATAQUE

Emocionante jogo onde você deve combater os invasores alienígenas, que tentam destruí-lo. Todo cuidado é pouco, pois, com graciosas evoluções elas praticam ataques suicidas para destruir sua nave. A esmagadora maioria numérica dos inimigos exige muita perícia para vencer.

As naves-mãe, quando atingidas, possibilitam um maior número de pontos para você. A cada fase completada o nível de dificuldade aumenta.

São fornecidas três naves para o combate. A cada três mil pontos completados, você receberá uma nave extra como bônus.

AUTO-ESTRADA

Você foi assistir um Grand Prix e conseguiu "tomar emprestado" um daqueles maravilhosos carros; para não atrapalhar a corrida, foi testar seus reflexos em uma auto-estrada.

Cuidado! Você será incomodado por "domingueiros", manchas de óleo, carros de bombeiro e outros perigos mais.

Aproximando-se do acostamento, a velocidade de seu carro diminuirá, ocasionando rodopios. O painel de instrumentos, com velocímetro e odômetro o auxiliará a pilotar.

Ao iniciar a partida, você possui três chances de colisão. Veja no painel a quantidade de chances disponíveis.

BOMBARDEIRO

Você é um piloto de caça, encarregado de missões de ataque em território inimigo. Como veterano de muitos combates, os mais perigosos objetivos são reservados para você.

Ao todo serão cinco missões para cumprir. O futuro de toda uma guerra, e de todas as nações envolvidas, depende de sua perícia e habilidade em contornar os inusitados perigos que surgirão.

A autonomia de voo de sua aeronave é limitada. Durante a missão, o centro de controle enviará aviões de reabastecimento, os quais lançarão cargas de com

bustível e bombas para você. Muita calma e nervos de aço são os principais requisitos para se reabastecer em vôo.

CORRIDA

Você é um piloto de corridas de demolição. Uma potente máquina está em suas mãos, para vencer um desafio nunca sequer imaginado: correr em uma pista fechada, na mão de direção contrária em relação a um carro teleguiado, sem colidir.

Para vencer cada bateria, você deverá percorrer todo o circuito, passando em cada local pelo menos uma vez. Para efeito de controle da comissão organizadora, a pista está toda marcada, e o seu carro, enquanto anda, irá apagando as marcas do piso.

Ao final de algumas baterias, um outro carro teleguiado será adicionado ao circuito, também em mão de direção contrária à sua.

ELIMINATOR

Suas qualidades de piloto espacial estão sendo postas à prova. Todos os perigos deverão ser enfrentados com a máxima habilidade, cumprindo o seu destino de eterno combatente, no espaço sideral.

Sua espaçonave está inteiramente equipada para longas viagens e possui um poder de fogo invejável em sua categoria.

Destrua os inimigos de sua civilização. Os destinos do universo estão em suas mãos.

FLIPERAMA

A máquina de jogos do futuro está desafiando você para uma deliciosa partida. Perícia, habilidade e nervos de aço são os principais requisitos para o jogador.

Um domínio completo das condições de jogo são oferecidos por este programa, que transforma o seu micro-computador em um sofisticado FLIPERAMA, com grande capacidade de entretenimento para você e seus amigos.

Venha jogar, prove a si mesmo a sua perícia, jogando contra a sorte nesta fabulosa máquina de jogos eletrônicos.

MULTI-INVADER

Seres desconhecidos, vindos de um longínquo planeta, atacam sua base estelar. Como único combatente de plantão, sua missão é fazer frente ao ataque. Em seu posto, está disponível um canhão laser.

Naves de transporte, que cruzarem a área de combate, também deverão ser destruídas. Toda a sua habilidade deve ser utilizada para não permitir que as naves pousem na base.

Lute até suas últimas forças. Os inimigos não desistirão, mesmo que várias esquadrilhas de combate sejam destruídas.

PÂNICO

Sua curiosidade levou-o a invadir um prédio, por todos temido, devido ao seu estranho habitante: um cientista que faz estudos sobre genética. Qual não foi a sua surpresa ao constatar que o cientista está morto e só restam estranhos seres, frutos de sua pesquisa.

Para destruí-los, você deve cavar buracos, sempre com o pé da frente no chão, e quando eles caírem, derrube-os andar abaixo, batendo com um martelo em suas cabeças. Você também poderá utilizar os buracos como passagem de um andar para outro.

O jogo terá início com apenas três seres, mas esse número aumentará gradativamente e surgirão outras espécies, cuja resistência chega a ser tanta que só morrerão se caírem de dois ou três andares.

PAPA TUDO

Um feiticeiro colocou-o em um labirinto, habitado por monstros-fantasmas. Nesse labirinto ocorrem coisas estranhas... Uma delas é que todo humano é transformado em um faminto PAPA TUDO. Agora você carregará a triste sina de ter que satisfazer sua imensa fome. Utilize toda sua habilidade e reflexos para comer as vitaminas que estão espalhadas pelo labirinto, sem ser pego pelos monstros-fantasmas.

As vitaminas maiores têm o poder de cegar temporariamente os monstros. Aproveite estes momentos para comê-los, mas cuidado, pois a qualquer instante eles voltarão ao normal. Periodicamente aparecerão frutas que valem um número maior de pontos. Você tem três vidas, mas este número poderá ser aumentado durante a partida, pois, a cada vinte mil pontos conseguidos, você ganhará uma vida extra como bônus.

PULO DO SAPO

Emocionante jogo, onde o objetivo é guiar o sapo, desde o ponto de partida, até uma casa vazia. Você deve cruzar a rodovia, com muito cuidado, pulando em seguida sobre as tartarugas e troncos, de forma a atravessar o perigoso rio e chegar em casa, são e salvo. Algumas das tartarugas, periodicamente, mergulham. Estando sobre uma delas quando isto ocorrer, pule fora rapidamente, para não perder a sua vida.

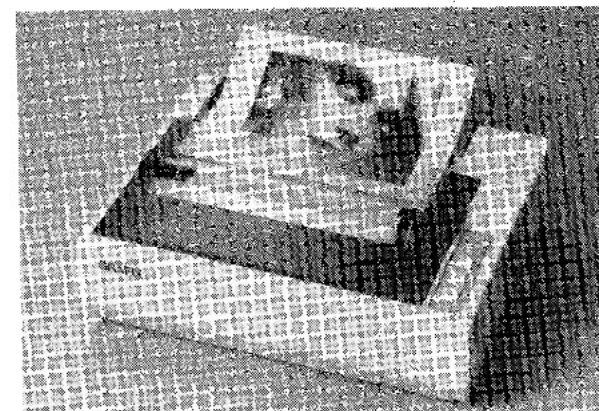
Você dispõe de um tempo limitado para efetuar a travessia. A aventura desenvolve-se em várias fases, cada uma delas mais perigosa que a anterior.

Impressora GRAFIX

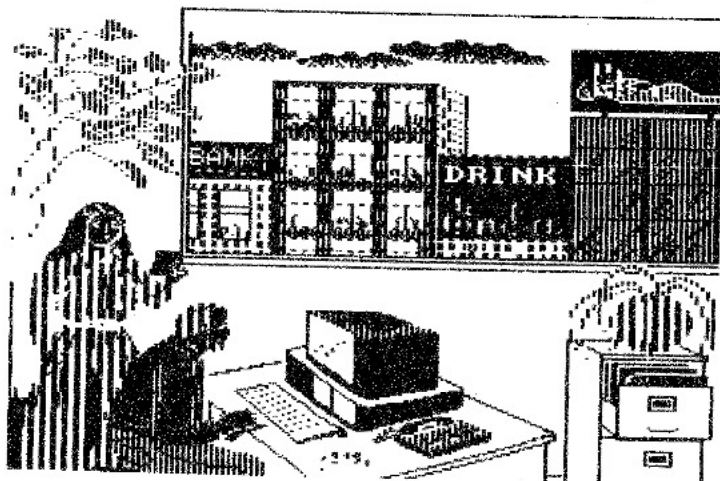
GRAFIX é a nova impressora da SCRITTA, que entra no mercado de informática com um produto de alta tecnologia e bom índice de nacionalização.

Imprime com uma matriz 9x9, a 100 cps, bidirecionalmente com procura lógica, isto é, não perde tempo voltando ao início da linha ou indo até o final. A GRAFIX imprime em quatro modos: normal, comprimido, expandido e comprimido-expandido. Além disso, ela imprime nas metades superior e inferior da linha, justamente o que se esperava no mercado para acabar com os problemas de índices, expoentes e uma série infindável de sinais. Para os grifos, a GRAFIX apresenta a grande vantagem de imprimir também em itálico, por exemplo, comprimidos expandidos, em cima ou em baixo, como quiser, até mesmo com as famosas linhas de sublinhar, que podem ser diretas, contínuas ou com outros truques. A impressora da SCRITTA permite ainda o controle de espaçamento entre as linhas, tabulação horizontal com 112 posições e vertical com 64, controle de posição de picote (pode pulá-lo automaticamente), usa papel de qualquer largura entre 4 e 10 polegadas, avisa o computador que o papel acabou, imprime acentos, cedilhas sem ter que voltar. Ela tem 207 caracteres, incluídos \$, 1º, & e outros.

O mais importante é que a GRAFIX opera com resolução de 120 pontos por polegadas, ou seja, 960 pontos em papel de 8". A GRAFIX é fabricada em dois modelos: GRAFIX-80 (80 colunas) e GRAFIX-100 (132 colunas). A GRAFIX possui caracteres acentuados da língua portuguesa.



Com a GRAFIX é muito fácil obter desenhos como os indicados abaixo:



E no momento o que podemos dizer é: "Parabéns engenheiros Ivan Bettito (Mackenzie), Edmundo Panzoldo Teixeira (Mackenzie), Caio S. Shimada (ITA) e Ivaldo Bettito (FAAP) pelas "joias" que são as impressoras que vocês desenvolveram com quase todos os componentes realmente feitos no Brasil".

Bibliografia

- ANSTIS, S.
Write Your Own APPLE Games
Creative Computing Press - Morris Plains
New Jersey - 1983
- BAILEY, H.J. - KERLIN, J.E.
APPLE Graphics - Activities Handbook
Robert J. Brady Co., Bowie - 1984
- BARNETT, M.P. - BARNETT, G.K.
Personal Graphics for Profit and Pleasure on
the APPLE II Plus Computer.
Little, Brown and Company - Boston - 1983
- BORATTO, F.
BASIC para Engenheiros e Cientistas
Livros Técnicos e Científicos - Editora S.A. -
Rio de Janeiro - 1984
- CASTRUCCI, B. - GIOVANNI, J.R.
A conquista da Matemática - 8ª série - 1º grau
Editora F.T.D. - São Paulo, 1983
- COAN, J.S.
BASIC APPLE BASIC
Hayden Book Company, Inc. - New Jersey - 1982
- CHANCE, D.
33-Challenging Games for TRS-80/APPLE/PET
Tab Books Inc. - Blue Ridge - 1981
- CUELLAR, G.
Fancy Programming in Applesoft
Reston Publishing Company - Virginia - 1983

- DEVON, R.F.
The First Few Bytes
Kendall/Hunt Publishing Company - Iowa - 1984
- ELGARTEN, G.H.-POSAMENTIER, A.S.-MORESH, S.E.
Using Computers in Mathematics
Addison-Wesley Publishing Company - Menlo Park
California - 1981
- GARDNER, D.A.-GARDNER, M.L.
APPLE BASIC Made Easy
Prentice-Hall, Inc - New Jersey - 1984
- GILDER, J.H.
BASIC Computer Programs in Science and
Engineering
Hayden Book Company, Inc. - New Jersey - 1980
- GOLDSTEIN, L.J.-GOLDSTEIN, M.
Robert J.Brady Co. - Bowie - Maryland - 1982
- GRILLO, J.P.-ROBERTSON, J.D.
Color Computer Applications
John Wiley and Sons, Inc. - New York - 1983
- HEARN, D.-BAKER, M.P.
Microcomputer Graphics
Prentice-Hall, Inc - Englewood Cliffs, N.J.-1983
- LAMOITIER, J.P.
BASIC Exercises for the APPLE
SYBEX Inc - 1982
- MANUAL TÉCNICO DO TK-2000 COLOR
Microdigital Eletrônica Ltda. - S.Paulo - 1984
- MASON, T.-PAYNE, S.-BLACK, B.
Picture Perfect Programming in APPLESOFT BASIC
A Reston Computer Group Book
A Prentice-Hall Company - Virginia - 1984
- MANUAL DE OPERAÇÃO DO TK-2000 COLOR
MICRODIGITAL Eletrônica Ltda - S.Paulo - 1984
- MANUAL DE INSTRUÇÕES - ELPPA II Plus
Victo. do Brasil Eletrônica Ltda. -S.P.-1984
- MIRSHAWKA, V.
BASIC sem segredos
Livraria Nobel S.A. - S.Paulo - 1984

- MIRSHAWKA, V.
Dê um APPLE à sua vida
Livraria Nobel S.A. - S.Paulo - 1984
- MIRSHAWKA, V.
Linguagem BASIC
Livraria Nobel S.A. - S.Paulo - 1984
- MIRSHAWKA, V.
Termologia - volume 1
Livraria Nobel S.A. - S.Paulo - 1966
- MIRSHAWKA, V.
TK-Calculando
Livraria Nobel S.A. - S.Paulo - 1984
- MIRSHAWKA, V.
TK-Divertindo
Livraria Nobel S.A. - S.Paulo - 1984
- MIRSHAWKA, V.
TK-Lembrando
Livraria Nobel S.A. - S.Paulo - 1984
- MIRSHAWKA, V.
TK-2000 na Matemática
Livraria Nobel S.A. - S.Paulo - 1984
- MOSHER, D.
Your Color Computer
SYBEX - Berkley - 1984
- MYERS, R.E.
Microcomputer Graphics
Addison-Wesley Publishing Company - Menlo Park,
California - 1982
- PECKHAM, H. - ELLIS, Jr.W.-LODI, E.
Manual de BASIC para sistemas compatíveis com o
Apple II
Mc Graw Hill do Brasil - S.Paulo - 1984
- PELCZARSKI, M.-TATE, J.
The Creative Apple
Creative Computing Press - Morris Plains,
New Jersey - 1982

REVISTA NOVA MATEMÁTICA MODERNA
Editora Lisa

TROST, S.R.

APPLE II - BASIC Programs in Minutes
SYBEX - Berkley - 1983

WADSWORTH, N.

Introduction to Computer Animation
Hayden Book Company, Inc. - New Jersey-1979

WATTENBERG, F.

Your APPLE II Need You: 30 Programming
Projects for the APPLE II
Prentice-Hall, Inc. - Englewood Cliffs,
New Jersey - 1984

WILLIAMS, K.-KERNAGHAN, B.-KERNAGHAN, L.

APPLE II Computer Graphics
Robert J. Brady Co. A Prentice-Hall Publishing
and Communications Company - Bowie - 1983

ZUANICH, M.A.-LIPSCOMB, S.D.

BASIC Fun with graphics
Avon Books - New York - 1983

Nº 0824