

UMA MATRIZ PARA ELABORAÇÃO DE TESTES EM MICROCOMPUTADORES PESSOAIS

† Winston de Castro Graça

O autor propõe uma matriz para testes, em Basic, onde as questões são introduzidas em linhas Data, no fim da matriz, facilitando a construção de testes objetivos para quaisquer assuntos. A matriz roda com apenas 4 kbytes, cedendo uma grande área livre da RAM para a alocação das questões.

The author proposes a computer matrix program in which questions are introduced through Data lines at the program, making objective tests building easier for any kind of subject. This matrix runs with only 4 kbytes, giving an extensive RAM area to allocate questions.

INTRODUÇÃO

A construção de testes em Basic não é uma tarefa difícil. Os programas resultantes porém se tornam cansativos em sua construção se não se obedecer os seguintes parâmetros: estabelecer subrotinas corretas e bem calculadas dentro de um fluxograma; alocar as questões em linhas Data, ao final

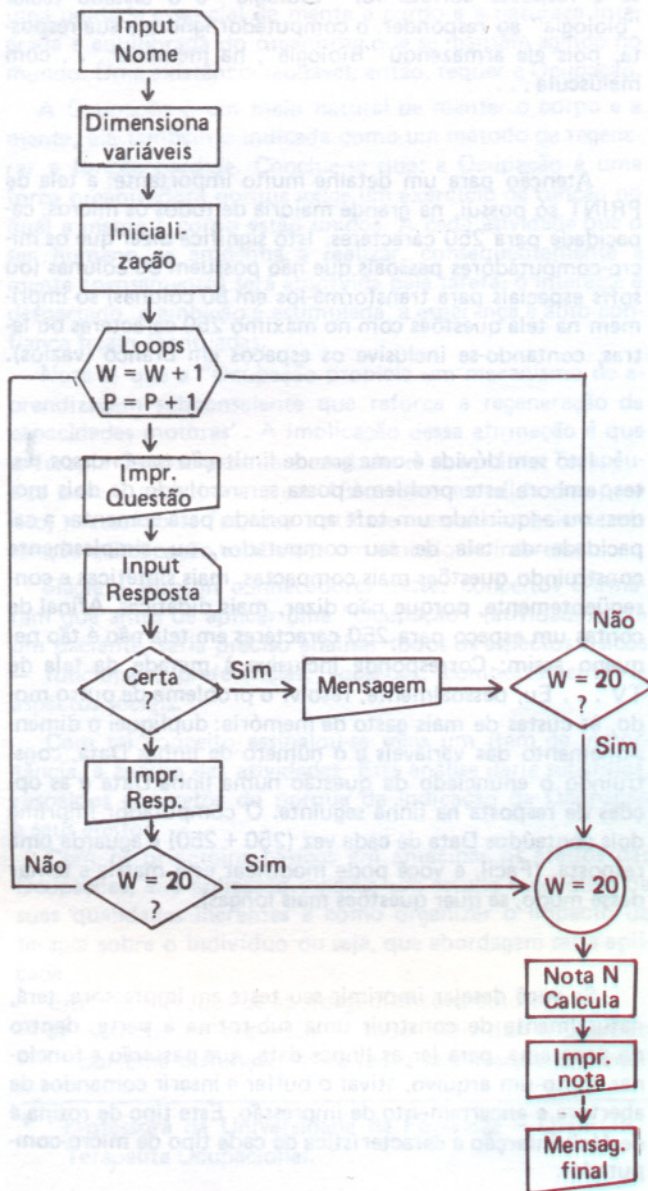
do programa, resultando em maior economia de bytes, maior facilidade de construção, orientação, remoção e correções; um fluxograma bem definido, de construção simples, para não acontecer um número excessivo de loops embutidos (ninhos de nexts) podendo ocorrer inclusive o risco de o programa (e o programador principalmente) se perder em espirais e o computador (e o programador também) entrar em "non sense".

* Professor de Cinesiologia da Universidade de Fortaleza - UNIFOR.

O programa aqui proposto tem um fluxograma simples e helicoidal, com um núcleo repetitivo (correspondendo ao centro da hélice), e foi construído para um Computador CP - 400, da Prológica. O CP - 400, apesar de "falar" o Basic tem pequenas variações de linguagem ("sotaques") que podem ser convertidas para qualquer outro micro, contanto que o programador saiba fazer a adaptação. O PRINT @, p. ex., é uma forma de tabulação de tela correspondente ao PRINT TAB de outros computadores. A tabulação @ (arroba) vai de 0 (zero) a 510 pontos de posição na tela da TV para o CP - 400. Sabendo-se isto, é fácil converter qualquer dos números arroba do programa para outros micros. Outra exclusividade do CP é a forma de expressão sonora, feita através de comandos Sound ou Play. Isto porém é de menos importância, pois vinhetas sonoras são subjetivas e a gosto do programador.

A String da linha 20 do programa também pode parecer "sui generis", mas trata-se apenas de um label de demarcação de tela, uma pequena fita grega cuja função é puramente estética.

FLUXOGRAMA DO PROGRAMA



LISTAGEM

```

1 Rem "testdata"
10 Clear 5000:Dim Q$(20):Dim Q(20)
12 Gosub 7000
15 For Q = 1 to 20: Read Q$(Q):Next Q
20 TI$ = "Titulo":S$ =String$(32,150):Clsrnd(8)
30 Input "Seu primeiro nome"; No$:Play "L15GFFDE-CC"
35 Print @ 230, "Boa Sorte,"; NO$; : For F = 1 to 1000: Next
50 Cls 0 : Print @ 64,S$; : Print @ 416,S$;
60 Print @ 230, "Teste - "; TI$; : Print @ 295, "20 QUESTÕES"; : Print @ 354, "Som"; : Print @ 375, "Enter";
70 Soudrnd(200), 3: If inkey$ = "" then 70
80 Cor = rnd(8) : cor = 1 then cor = 3
90 W = 1 : Ponto = 0 : Pause = 44539
100 Gosub 5000
101 RC$ = "A" : Gosub 2000 : Gosub 5000
102 RC$ = "B" : Gosub 2000 : Gosub 5000
.
.
.
112 C$ = "Palavra certa" : Gosub 1000 : Gosub 5000 : Rem Quest Subjet
.
.
.
120 RC$ = "D" : Gosub 2000
500 Cls 0 : Print @ 64,S$; : Print @ 240, "Fim"; : Print @ 321, "Vc. Respondeu 20 questões";
510 Print @ 360, "sobre"; TI$; : Print @ 385, "e acertou"; Ponto; "Questões";
515 N = Ponto * 10/20
520 Print @ 417, "Sua nota e' ";N;" - digite fim"; : Print @ 448,S$;
540 F$ = "fim" : Play "L20ABCDEF GFEDCC" : Input Z$
550 If Z$ = F$ goto 600 else end
600 Cls cor: If N = 5 then print NO$; ", estude mais" else goto 6300
610 Sound rnd(200), 1 : For t = 1 to 100 : Next t: goto 610
1000 Rem SubRot Quest Subjet
1050 Input N$ : N$ = C$ gosub 6000 else gosub 5500
1100 Return
2000 Rem SubRot Quest Objet
2050 Input R$ : R$ : If R$ = RC$ gosub 4000 else gosub 4500
2100 Return
4000 Print "CERTO"; : Sound 8,4: Sound 4,3: Print "Muito bem,"; NO$ : Ponto = Ponto + 1 : Print S$
4100 Gosub 6100
4200 Return
4500 Sound 128,8 : Print "Errado"; ", a resposta e' "; RC$ : print S$;
4600 Gosub 6100
4700 Return
5000 Cls cor: Sound rnd(200),4 : Print @ 64,S$; : Print Q$(W) : W = W + 1: Return
  
```

```

5500 Sound 128,8 : Print "Errado"; "a resposta e"; C$ :
      Gosub 6100
5600 Return
6000 Sound 8,4 : Sound 4,3 : Print "Certo, muito bem";
      NO$ : Ponto = Ponto + 1
6100 Print " - Enter"; : Exec Pause
6200 Return
6300 Print @ 230, "Mensagens"; NO$;
6400 Play "L3CCDCFE; P2" : End
7000 Rem Inserção de questões em linhas DATA
7001 Data "1. Questão . . ."
7002 Data "2. Questão . . ."
.
.
.
7100 Return
7200 End

```

DEFINIÇÃO DE VARIÁVEIS:

TI\$ = Título ou assunto do teste
Q\$ = Questão
R\$ = Resposta objetiva
C\$ = Resposta subjetiva
RC\$ = Resposta certa
NO\$ = Nome do testado
S\$ = String
Q = Número DATA
W = Flag para Loop contador
N = Nota
Ponto = Questão respondida corretamente

PRINCIPAIS SUB ROTINAS

- Linha 10 – dimensionamento de variáveis para 20 questões.
- Linha 12-15 : leitura das linhas Data (20 linhas).
- Linha 20-80 : construção da abertura.
- Linha 101-120 : colocação das respostas corretas de cada questão.
- Linha 500-610 : encerramento, cálculo de nota e mensagem final.
- Linha 1000-1100 : sub-rotina para resposta subjetiva (uma palavra ou frase). Se resposta correta, salta para linha 6000, se errada, para a 5500.
- Linha 2000-2100 : Sub-rotina para resposta objetiva (uma letra). Se correta salta para a sub-rotina 4000, se errada, para a 4500.
- Linha 7000-7100 : linhas Data contendo as questões.
- Linha 5000-5600 : núcleo rotativo da hélice do programa. Mostra cada questão na tela, de uma por uma, aguardando resposta, imprimindo a resposta correta se a entrada foi errada, imprimindo mensagem de estímulo se a resposta foi correta. Esta pode ser considerada a sub-rotina principal para fazer girar o programa. Contem ainda um flag (W) para um loop contador de questões.

INSTRUÇÕES PARA A CONSTRUÇÃO DE TESTES USANDO-SE A MATRIZ

O teste está dimensionado na matriz, na linha 10 da listagem do programa, para 20 questões. Pode entretanto ser aumentado para mais questões, alterando-se o dimensionamento das variáveis. O consumo de memória porém cresce em progressão geométrica para cada valor dimensionado a mais na memória do computador, o que não é desejável para micros de pequeno porte, sem expansores de memória, sobrando portanto menos memória Ram para alocar questões, sobretudo se estas forem extensas em sua maioria, correndo-se o risco de um indesejável OM (over memory) antes de se completar o teste.

As respostas corretas do teste devem ser colocadas entre as pas das questões 101 a 120. Se a resposta é uma palavra, como por exemplo na linha 112, coloque-a também entre aspas, mas cuidado com a sua grafia: P. ex., se a resposta correta for "Biologia" e o testado teclar "biologia" ao responder, o computador ignorará sua resposta, pois ele armazenou "Biologia", na memória . . . com maiúscula . . .

Atenção para um detalhe muito importante: a tela de PRINT só possui, na grande maioria de todos os micros, capacidade para 250 caracteres. Isto significa dizer que os micro-computadores pessoais que não possuem 80 colunas (ou softs especiais para transformá-los em 80 colunas) só imprimem na tela questões com no máximo 250 caracteres ou letras, contando-se inclusive os espaços em branco (vazios).

Isto sem dúvida é uma grande limitação para nossos testes, embora este problema possa ser resolvido de dois modos: ou adquirindo um soft apropriado para aumentar a capacidade de tela de seu computador, ou, simplesmente construindo questões mais compactas, mais sintéticas e conseqüentemente, porque não dizer, mais didáticas. Afinal de contas um espaço para 250 caracteres em tela não é tão pequeno assim: Corresponde inclusive à metade da tela de TV . . . Eu, pessoalmente, resolvi o problema de outro modo, às custas de mais gasto de memória: dupliquei o dimensionamento das variáveis e o número de linhas Data, construindo o enunciado da questão numa linha Data e as opções de resposta na linha seguinte. O computador imprime dois conteúdos Data de cada vez (250 + 250) e aguarda uma resposta. (Fácil, e você pode modificar esta matriz e tentar desse modo, se quer questões mais longas).

Se você desejar imprimir seu teste em impressora, terá, naturalmente de construir uma sub-rotina à parte, dentro do programa, para ler as linhas data, que passarão a funcionar como um arquivo, ativar o buffer e inserir comandos de abertura e encerramento de impressão. Este tipo de rotina é de fácil inserção e característica de cada tipo de micro-computador.