

# Desvendando a Torre de Hanói

Mário Leite

...

A Torre de Hanói é um clássico jogo de estratégia muito utilizado para testes de memória e pode ser empregado em desafios de lógica. O esquema é constituído por três postes verticais nos quais devem ser colocados discos de diâmetros diferentes com furos no centro, e podendo variar o número de discos. O jogo inicia com todos os discos no poste da esquerda e a finalidade é movimentar todos os discos para o poste mais à direita de acordo com as seguintes regras:

- Movimentar apenas um disco de cada vez.
- Movimentar apenas o disco de cima de cada poste.
- Nunca colocar um disco sobre outro disco de diâmetro menor.

O número mínimo de movimentos é calculado através da expressão:  $2^n - 1$ , onde  $n$  é o número de discos. A **figura 1a** mostra como os três discos devem ficar, inicialmente, no primeiro poste; e a **figura 1b** como devem ficar ao final do jogo.

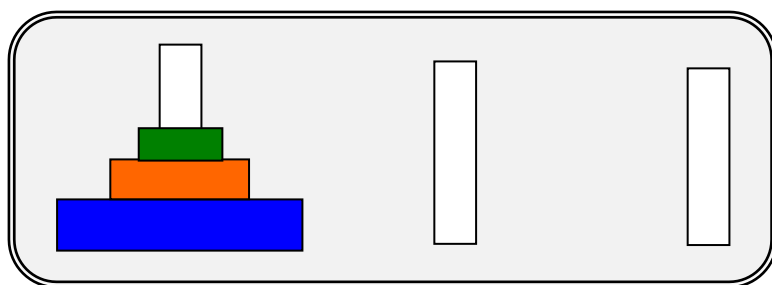


Figura 1a - Situação inicial do jogo “Torre de Hanói”

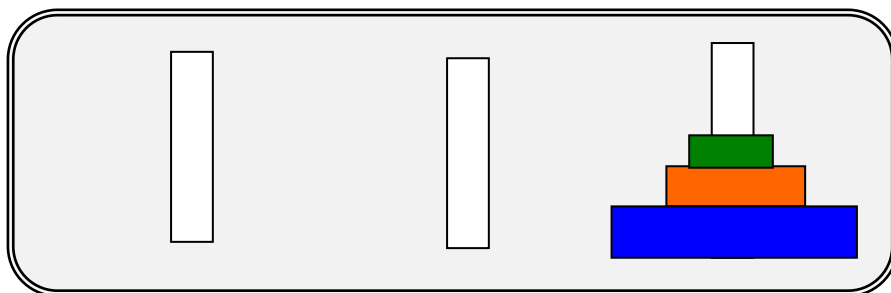


Figura 1b - Situação final do jogo “Torre de Hanói”

Como pode ser observado pela fórmula, para três discos o número **mínimo** de movimentos que deve ser executado para terminar o jogo com sucesso é  $2^3 - 1 = 7$ . Isto quer dizer que o jogador pode executar 7, 8, 9, ...  $n$  movimentos, mas sete é o mínimo que se consegue para otimizar a solução. A **figura 2a** mostra uma solução em nove movimentos, e a **figura 2b** com a solução otimizada em sete. A **figura 3** mostra a saída do programa “TorreHanoi”, codificado em Pascal. E a despeito dos que criticam esta linguagem, taxando-a de “velha e ultrapassada”, resolvi utilizá-la para mostrar que, embora com algumas limitações na parte gráfica, esta linguagem ainda pode ser muito eficiente no aprendizado em Programação e em programas práticos, além de acessar dispositivos da máquina usando recursos em baixo nível, como faz a linguagem C.

---

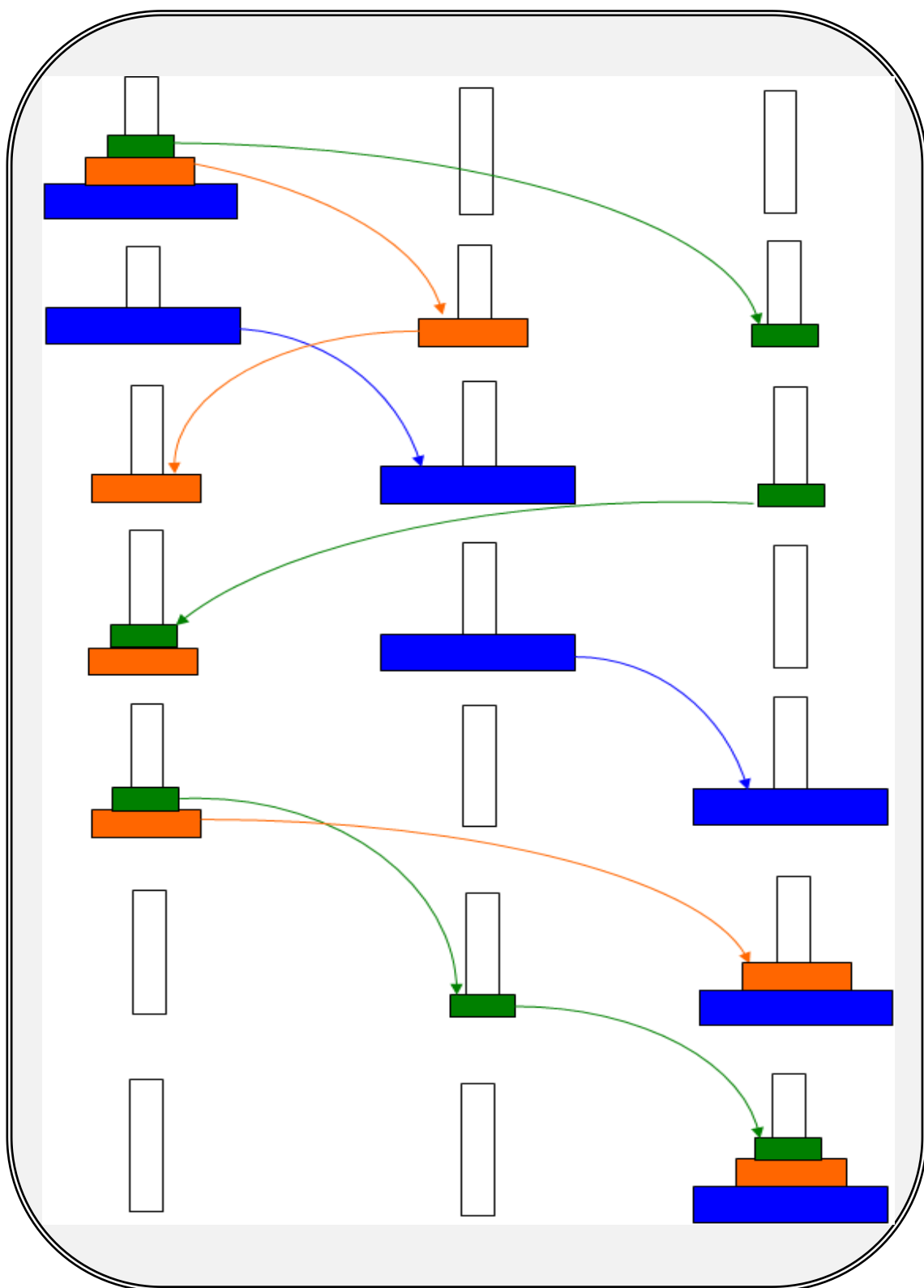


Figura 2a - Solução da “Torre de Hanói” em nove movimentos

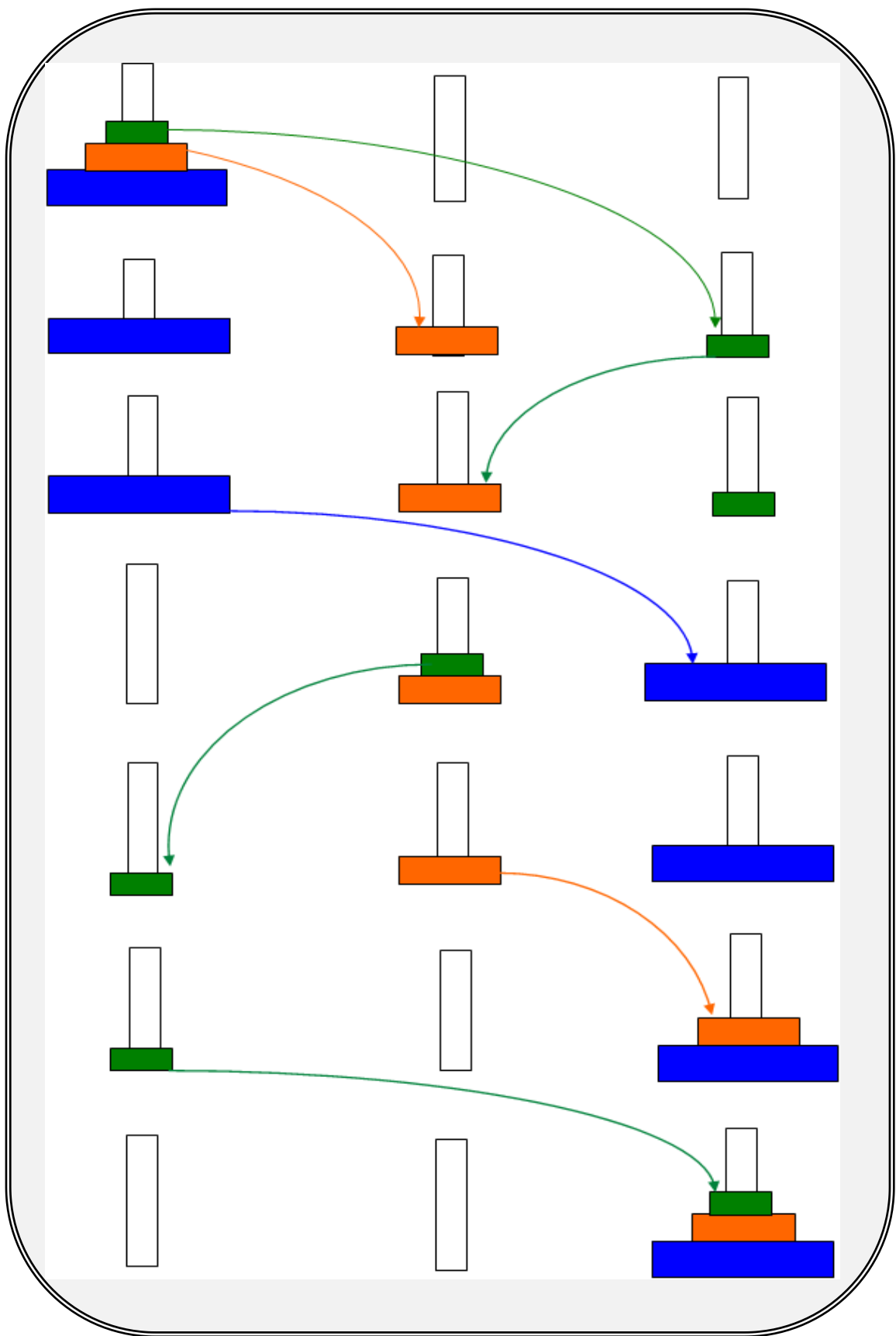


Figura 2b - Solução da “Torre de Hanói” em sete movimentos

```
Pascalzim Console (pressione ctrl+c para interromper o programa)
Situação inicial: Primeiro movimento
Poste1: DDDVDA
Poste2:
Poste3:

Segundo movimento
Poste1: DA
Poste2: DV
Poste3: DD

Terceiro movimento
Poste1: DV
Poste2: DA
Poste3: DD

Quarto movimento
Poste1: DDDV
Poste2: DA
Poste3:

Quinto movimento
Poste1: DDDV
Poste2:
Poste3: DA

Sexto movimento
Poste1:
Poste2: DD
Poste3: DA

Sétimo movimento
Poste1:
Poste2:
Poste3: DDDVDA

Terminou!!
```

Figura 3 - Saída do Programa “TorreHanoi”

```

Program TorreHanoi;
//Reproduz o clássico jogo "Torre de Hanoi" de maneira dinâmica, com movimentos
//dos blocos nos postes. Os três discos aqui foram substituídos por blocos.
//Autor: Mário Leite - marleite@gmail.com
//-----
Uses Crt;
Var Poste: array[1..3] of string;
    m: integer;
//-----
Procedure ProMovimentoInicial;
//Situação inicial - Executa o primeiro movimento
    Var Linha: integer;
Begin
    Linha := 1;
    WriteLn('Situação inicial: Primeiro movimento');
    WriteLn(' ');
    Poste[1] := 'DD' + 'DV' + 'DA';
    Poste[2] := ' ';
    Poste[3] := ' ';
    WriteLn('Poste1: ', Poste[1]);
    WriteLn('Poste2: ', Poste[2]);
    WriteLn('Poste3: ', Poste[3]);
    Gotoxy(24,Linha+3); //posiciona o curso na tela
    TextColor(Yellow); //disc maior na cor amarela
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176));
    Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    Delay(200);
    Gotoxy(25,Linha+2);
    TextColor(LightRed); //disc intermediário na cor vermelha
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    Delay(200); //interrompe o processamento por 200 milissegundos
    Gotoxy(26,Linha+1);
    TextColor(LightGreen); //disc menor na cor verde clara
    Write(Chr(176)); Write(Chr(176));
    ReadKey; //aguarda alguma tecla ser pressionada
End; //fim da procedure "ProMovimentoInicial"
//-----
Procedure ProSegundoMovimento;
//Executa o segundo movimento
    Var j, Linha: integer;
Begin
    Linha := 7;
    Gotoxy(1,Linha);
    WriteLn('Segundo movimento');
    Poste[3] := 'DD';
    Poste[2] := 'DV';
    Poste[1] := 'DA';
    WriteLn('Poste1: ', Poste[1]);
    WriteLn('Poste2: ', Poste[2]);
    WriteLn('Poste3: ', Poste[3]);
    for j:=Linha to (Linha+3) do begin //movimentação do disco verde
        Gotoxy(41,j);
        TextColor(LightGreen);
        Write(Chr(176)); Write(Chr(176));
        Delay(200);
        Gotoxy(41,j);
        if(j=Linha+3) then
            Break; //sai incondicionalmente do loop
        TextColor(Black);
        Write(Chr(176)); Write(Chr(176));
    end;
end;

```

```

for j:=Linha to (Linha+3) do begin //movimentação do disco vermelho
    Gotoxy(33,j);
    TextColor(LightRed);
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    Delay(200);
    Gotoxy(33,j);
    if(j=Linha+3) then
        Break; //sai incondicionalmente do loop
    TextColor(Black);
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
end;
for j:=Linha to (Linha+3) do begin //movimentação do disco amarelo
    gotoxy(24,j);
    TextColor(Yellow);
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176));
    Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    Delay(200);
    Gotoxy(24,j);
    if(j=Linha+3) then
        Break;
    TextColor(Black);
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176));
    Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
end;
ReadKey;
End; //fim da procedure "ProSegundoMovimento"
//-----
Procedure ProTerceiroMovimento;
//Executa o terceiro movimento
    Var j, Linha: integer;
Begin
    TextColor(LightGreen);
    Linha := 13;
    Gotoxy(1,Linha);
    WriteLn('Terceiro movimento');
    Poste[1] := 'DV';
    Poste[2] := 'DA';
    Poste[3] := 'DD';
    WriteLn('Poste1: ', Poste[1]);
    WriteLn('Poste2: ', Poste[2]);
    WriteLn('Poste3: ', Poste[3]);
    Gotoxy(41,Linha+3); //mantém o disco verde na posição anterior
    TextColor(LightGreen);
    Write(Chr(176)); WriteLn(Chr(176));
    Delay(200);
    for j:=Linha to (Linha+3) do begin //movimentação do disco vermelho
        Gotoxy(25,j);
        TextColor(LightRed);
        Write(Chr(176)); Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
        Delay(200);
        Gotoxy(25,j);
        if(j=Linha+3) then
            Break;
        TextColor(Black);
        Write(Chr(176)); Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    end;
    for j:=Linha to (Linha+3) do begin //movimentação do disco amarelo
        gotoxy(32,j);
        TextColor(Yellow);
        Write(Chr(176)); Write(Chr(176)); Write(Chr(176));
        Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    end;
end;

```

```

        Delay(200);
        Gotoxy(32,j);
        if(j=Linha+3) then
            Break; //sai incondicionalmente do loop
        TextColor(Black);
        Write(Chr(176)); Write(Chr(176)); Write(Chr(176));
        Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    end;
    ReadKey;
End; //fim da procedure "ProTerceiroMovimento"
//-----
Procedure ProQuartoMovimento;
//Executa o quarto movimento
    Var j, Linha: integer;
Begin
    TextColor(LightGreen);
    Linha := 19;
    Gotoxy(1,Linha);
    WriteLn('Quarto movimento');
    Poste[1] := 'DD' + 'DV';
    Poste[2] := 'DA';
    Poste[3] := '  ';
    WriteLn('Poste1: ', Poste[1]);
    WriteLn('Poste2: ', Poste[2]);
    WriteLn('Poste3: ', Poste[3]);
    Gotoxy(25,22); //mantém o disco vermelho na posição anterior
    TextColor(LightRed);
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    Delay(200);
    for j:=(Linha-1) to (Linha+2) do begin //movimentação do disco verde
        Gotoxy(26,j);
        TextColor(LightGreen);
        Write(Chr(176)); Write(Chr(176));
        Delay(200);
        Gotoxy(26,j);
        if(j=Linha+2) then
            Break;
        TextColor(Black);
        Write(Chr(176)); Write(Chr(176));
    end;
    Gotoxy(32,Linha+3); //mantém o disco amarelo na posição anterior
    TextColor(Yellow);
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176));
    Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    ReadKey;
End; //fim da procedure "ProQuartoMovimento"
//-----
Procedure ProQuintoMovimento;
//Executa o quinto movimento
    Var j, Linha: integer;
Begin
    TextColor(LightGreen);
    Linha := 25;
    Gotoxy(1,Linha);
    WriteLn('Quinto movimento');
    Poste[1] := Poste[1];
    Poste[2] := '  ';
    Poste[3] := 'DA';
    WriteLn('Poste1: ', Poste[1]);
    WriteLn('Poste2: ', Poste[2]);
    WriteLn('Poste3: ', Poste[3]);

```

```

Gotoxy(25,28); //mantém o disco vermelho na posição anterior
TextColor(LightRed);
Write(Chr(176)); Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
Gotoxy(26,27); //mantém o disco verde na posição anterior
TextColor(LightGreen);
Write(Chr(176)); Write(Chr(176));
Delay(200);
for j:=(Linha-1) to (Linha+3) do begin //movimentação do disco amarelo
    Gotoxy(40,j);
    TextColor(Yellow);
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176));
    Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    Delay(200);
    Gotoxy(40,j);
    if(j=Linha+3) then
        Break;
    TextColor(Black);
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176));
    Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
end;
ReadKey;
End; //fim da procedure "ProQuintoMovimento"
//-----
Procedure ProSextoMovimento;
//Executa o sexto movimento
Var j, Linha: integer;
Begin
    TextColor(LightGreen);
    Linha := 31;
    Gotoxy(1,Linha);
    WriteLn('Sexto movimento');
    Poste[1] := ' ';
    Poste[2] := 'DD';
    Poste[3] := Poste[3];
    WriteLn('Poste1: ', Poste[1]);
    WriteLn('Poste2: ', Poste[2]);
    WriteLn('Poste3: ', Poste[3]);
    Gotoxy(40,34); //mantém o disco amarelo na posição anterior
    TextColor(Yellow);
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176));
    Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    for j:=Linha to (Linha+3) do begin //movimentação do disco verde
        Gotoxy(34,j);
        TextColor(LightGreen);
        Write(Chr(176)); WriteLn(Chr(176));
        Delay(200);
        Gotoxy(34,j);
        if(j=Linha+2) then
            Break;
        TextColor(Black);
        Write(Chr(176)); WriteLn(Chr(176));
    end;
    for j:=Linha to (Linha+2) do begin //movimentação do disco vermelho
        Gotoxy(41,j);
        TextColor(LightRed);
        Write(Chr(176)); Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
        Delay(200);
        Gotoxy(41,j);
        if(j=Linha+2) then
            Break;
        TextColor(Black);

```



```

        Write(Chr(176)); Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    end;
    ReadKey;
End; //fim da procedure "ProSextoMovimento"
//-----
Procedure ProSetimoMovimento;
//Executa o sétimo movimento
    Var j, Linha: integer;
Begin
    TextColor(LightGreen);
    Linha := 37;
    Gotoxy(1,Linha);
    WriteLn('Sétimo movimento');
    Poste[1] := ' ';
    Poste[2] := ' ';
    Poste[3] := 'DD' + 'DV' + 'DA';
    WriteLn('Poste1: ', Poste[1]);
    WriteLn('Poste2: ', Poste[2]);
    WriteLn('Poste3: ', Poste[3]);
    Gotoxy(40,40); //mantém o disco vermelho na posição anterior
    TextColor(Yellow);
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176));
    Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    Gotoxy(41,39); //mantém o disco vermelho na posição anterior
    TextColor(LightRed);
    Write(Chr(176)); Write(Chr(176)); Write(Chr(176)); WriteLn(Chr(176));
    Delay(200);
    for j:=Linha to (Linha+1) do begin //movimentação do disco verde
        Gotoxy(42,j);
        TextColor(LightGreen);
        Write(Chr(176)); Write(Chr(176));
        Delay(200);
        Gotoxy(42,j);
        if(j=Linha+1) then
            Break;
        TextColor(Black);
        Write(Chr(176)); Write(Chr(176));
    end;
End; //fim da procedure "ProSetimoMovimento"
//=====
//Programa principal
BEGIN
    ClrScr; //limpa a tela
    TextColor(LightGreen);
    CursorOff; //oculta o cursor
    //Chama as procedures para executar os movimentos
    ProMovimentoInicial;
    ProSegundoMovimento;
    ProTerceiroMovimento;
    ProQuartoMovimento;
    ProQuintoMovimento;
    ProSextoMovimento;
    ProSetimoMovimento;
    CursorOn; //torna o cursor visível
    {Reposiciona o cursor para baixo}
    for m:=1 to 6 do begin
        WriteLn(' ');
    end;
    WriteLn('Terminou!!');
    ReadKey;
END. //fim do programa "TorreHanoi"

```