Construção de Compiladores
Período Especial
Aula 12: Chamadas de Procedimento
Parâmetros por Referência

Bruno Müller Junior

Departamento de Informática
UFPR

2020

Introdução

- Na passagem de parâmetro por referência, o parâmetro real **OBRIGATORIAMENTE** é uma variável;
- Em tempo de execução, o endereço desta variável é colocado no espaço reservado ao parâmetro formal correspondente;
- Identifica-se um parâmetro passado por referência pelo uso da palavra reservada VAR à frente do parâmetro formal correspondente.

Semântica

- a variável "y"é um parâmetro formal passado por referência;
- ela contém um endereço, neste caso o endereço da variável "b";
- a instrução y:=y+x pode ser entendida como ⓑ:=ⓑ+x

```
program ref;
var a, b:integer;
   procedure p(x:integer, var y:integer)
   begin
      y:=y+x
   end

begin
   ...
   p(a, b);
   ...
end
```

Funcionamento

- Na chamada do procedimento é necessário empilhar o endereço de "b"e não seu valor (CREN);
- Dentro do procedimento a variável é acessada indiretamente através de "y":
- CREN: Carrega Endereço
  `M[s]:=endereço de b`
- CRVI: Carrega Valor Indireto
  `M[s]:=M[M[y]]`
- ARMI: Armazena Valor Indireto
  `M[M[y]]:=M[s]`

```
program ref;
var a, b:integer;
   procedure p(x:integer, var y:integer)
   begin
      y:=y+x
   end

begin
   ...
   p(a, b);
   ...
end
```

MEPA

- As novas instruções tratam parâmetros formais por referência.
- Em tempo de execução, os endereços léxicos indicados contém endereços físicos das variáveis-alvo.

Instrução	Ação	Significado

CREN k,n	`s:=s+1`
	`M[s]:=D[k]+n`
	`i:=i+1`
		Carrega Endereço

CRVI k,n	`M[M[D[k]+n]]:=M[s];`
	`s:=s+1`
	`i:=i+1`
		Carrega Valor Indireto

ARMI k,n	`M[s]:=M[M[D[k]+n]]`
	`i:=M[s-2]`
	`i:=i+1`
		Armazena Valor Indireto

Exemplo

```
program ref;
var a, b:integer;
   procedure p(x:integer, var y:integer)
   begin
      y:=y+x
   end

begin
   ...
   p(a, b);
   ...
end
```

```
...
CRVI 1,-4
CRVL 1,-5
SOMA
ARMI 1,-4
...
CRVL 0,0
CREN 0,1
CHPR...
...
```

Tradução

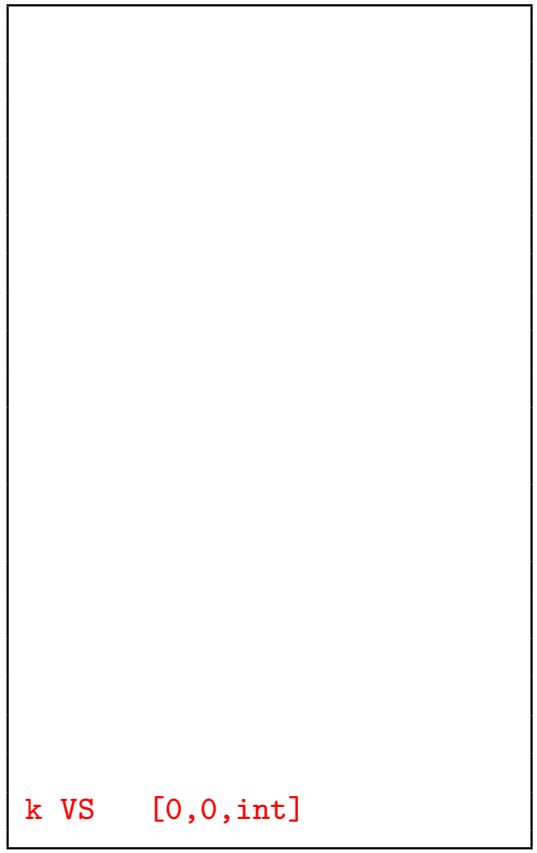
- Até o momento só trabalhamos com variáveis simples e parâmetros formais passados por valor;
- A tradução era [CRVL,ARMZ];
- As novas instruções são aplicadas somente quando envolvem parâmetros passados por referência:
  - CREN: Chamadas de procedimento quando o parâmetro real é VS ou PF-valor;
  - CRVI/ARMI: a variável envolvida é um PF-referência;

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
program passRef(input, output);          INPP
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
INPP
AMEM 1
```



Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
   INPP
   AMEM 1
   DSVS R00
R01:ENPR 1
```

```
g PF   [1,?,int,ref]
n PF   [1,?,int,vlr]
p PROC [R01,1,?{}
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
```

```
g PF   [1,-4,int,ref]
n PF   [1,-5,int,vlr]
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
```

```
h VS   [1,0,int]
g PF   [1,-4,int,ref]
n PF   [1,-5,int,vlr]
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
```

```
h VS   [1,0,int]
g PF   [1,-4,int,ref]
n PF   [1,-5,int,vlr]
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
```

```
h VS   [1,0,int]
g PF   [1,-4,int,ref]
n PF   [1,-5,int,vlr]
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
```

```
h VS   [1,0,int]
g PF   [1,-4,int,ref]
n PF   [1,-5,int,vlr]
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
      CRVL 1,-5
      CRCT 1
      SUBT
      CREN 1,0
      CHPR R01,1
```

```
h VS   [1,0,int]
g PF   [1,-4,int,ref]
n PF   [1,-5,int,vlr]
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
      CRVL 1,-5
      CRCT 1
      SUBT
      CREN 1,0
      CHPR R01,1
      CRVL 1,0
      ARMI 1,-4
```

```
h VS   [1,0,int]
g PF   [1,-4,int,ref]
n PF   [1,-5,int,vlr]
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
      CRVL 1,-5
      CRCT 1
      SUBT
      CREN 1,0
      CHPR R01,1
      CRVL 1,0
      ARMI 1,-4
```

```
      CRVL 1,-5
      CRCT 2
      SUBT
==>   CRVL 1,-4 <==
      CHPR R01,1
```

```
h VS   [1,0,int]
g PF   [1,-4,int,ref]
n PF   [1,-5,int,vlr]
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
      CRVL 1,-5
      CRCT 1
      SUBT
      CREN 1,0
      CHPR R01,1
      CRVL 1,0
      ARMI 1,-4
      CRVL 1,-5
      CRCT 2
      SUBT
      CRVL 1,-4
      CHPR R01,1
R03:NADA
```

```
h VS   [1,0,int]
g PF   [1,-4,int,ref]
n PF   [1,-5,int,vlr]
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
      CRVL 1,-5
      CRCT 1
      SUBT
      CREN 1,0
      CHPR R01,1
      CRVL 1,0
      ARMI 1,-4
      CRVL 1,-5
      CRCT 2
      SUBT
      CRVL 1,-4
      CHPR R01,1
R03:NADA
      CRVL 1,-5
      IMPR
      CRVI 1,-4
      IMPR
```

```
h VS   [1,0,int]
g PF   [1,-4,int,ref]
n PF   [1,-5,int,vlr]
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
      CRVL 1,-5
      CRCT 1
      SUBT
      CREN 1,0
      CHPR R01,1
      CRVL 1,0
      ARMI 1,-4
      CRVL 1,-5
      CRCT 2
      SUBT
      CRVL 1,-4
      CHPR R01,1
R03:NADA
      CRVL 1,-5
      IMPR
      CRVI 1,-4
      IMPR
      DMEM 1
```

~~h VS [1,0,int]~~
g PF [1,-4,int,ref]
n PF [1,-5,int,vlr]
p PROC [R01,1,2{[i,v][i,r]}]
k VS [0,0,int]

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
      CRVL 1,-5
      CRCT 1
      SUBT
      CREN 1,0
      CHPR R01,1
      CRVL 1,0
      ARMI 1,-4
      CRVL 1,-5
      CRCT 2
      SUBT
      CRVL 1,-4
      CHPR R01,1
R03:NADA
      CRVL 1,-5
      IMPR
      CRVI 1,-4
      IMPR
      DMEM 1
      RTPR 1,2
```

```
~~g PF   [1,-4,int,ref]~~
~~n PF   [1,-5,int,vlr]~~
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
      CRVL 1,-5
      CRCT 1
      SUBT
      CREN 1,0
      CHPR R01,1
      CRVL 1,0
      ARMI 1,-4
      CRVL 1,-5
      CRCT 2
      SUBT
      CRVL 1,-4
      CHPR R01,1
R03:NADA
      CRVL 1,-5
      IMPR
      CRVI 1,-4
      IMPR
      DMEM 1
      RTPR 1,2
R00:NADA
```

```
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
      CRVL 1,-5
      CRCT 1
      SUBT
      CREN 1,0
      CHPR R01,1
      CRVL 1,0
      ARMI 1,-4
      CRVL 1,-5
      CRCT 2
      SUBT
      CRVL 1,-4
      CHPR R01,1
R03:NADA
      CRVL 1,-5
      IMPR
      CRVI 1,-4
      IMPR
      DMEM 1
      RTPR 1,2
R00:NADA
      CRCT 0
      ARMZ 0,0
```

```
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
      CRVL 1,-5
      CRCT 1
      SUBT
      CREN 1,0
      CHPR R01,1
      CRVL 1,0
      ARMI 1,-4
      CRVL 1,-5
      CRCT 2
      SUBT
      CRVL 1,-4
      CHPR R01,1
R03:NADA
      CRVL 1,-5
      IMPR
      CRVI 1,-4
      IMPR
      DMEM 1
      RTPR 1,2
R00:NADA
      CRCT 0
      ARMZ 0,0
      CRCT 3
      CREN 0,0
      CHPR R01,0
```

```
p PROC [R01,1,2{[i,v][i,r]}]
k VS   [0,0,int]
```

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

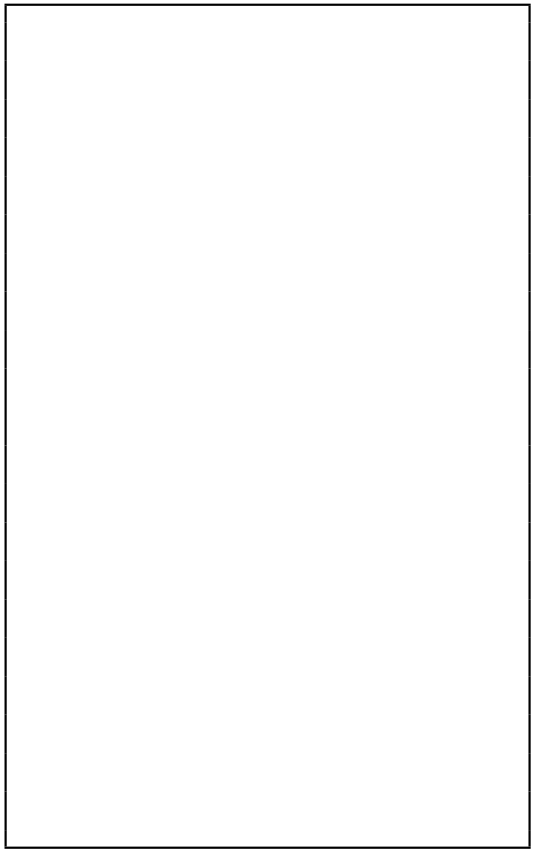
```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
      CRVL 1,-5
      CRCT 1
      SUBT
      CREN 1,0
      CHPR R01,1
      CRVL 1,0
      ARMI 1,-4
      CRVL 1,-5
      CRCT 2
      SUBT
      CRVL 1,-4
      CHPR R01,1
R03:NADA
      CRVL 1,-5
      IMPR
      CRVI 1,-4
      IMPR
      DMEM 1
      RTPR 1,2
R00:NADA
      CRCT 0
      ARMZ 0,0
      CRCT 3
      CREN 0,0
      CHPR R01,0
      DMEM 1
```

~~p PROC [R01,1,2{[i,v][i,r]}]~~
~~k VS [0,0,int]~~

Exemplo

```
program passRef(input, output);
var k: integer;
   procedure p(    n:integer;
                var g:integer);
   var h:integer;
   begin
      if (n<2)
         then g:=g+1
      else
      begin
         p(n-1,h);
         g:=h;
         p(n-2,g)
      end;
      write(n,g)
   end;
begin
   k:=0;
   p(3,k);
end.
```

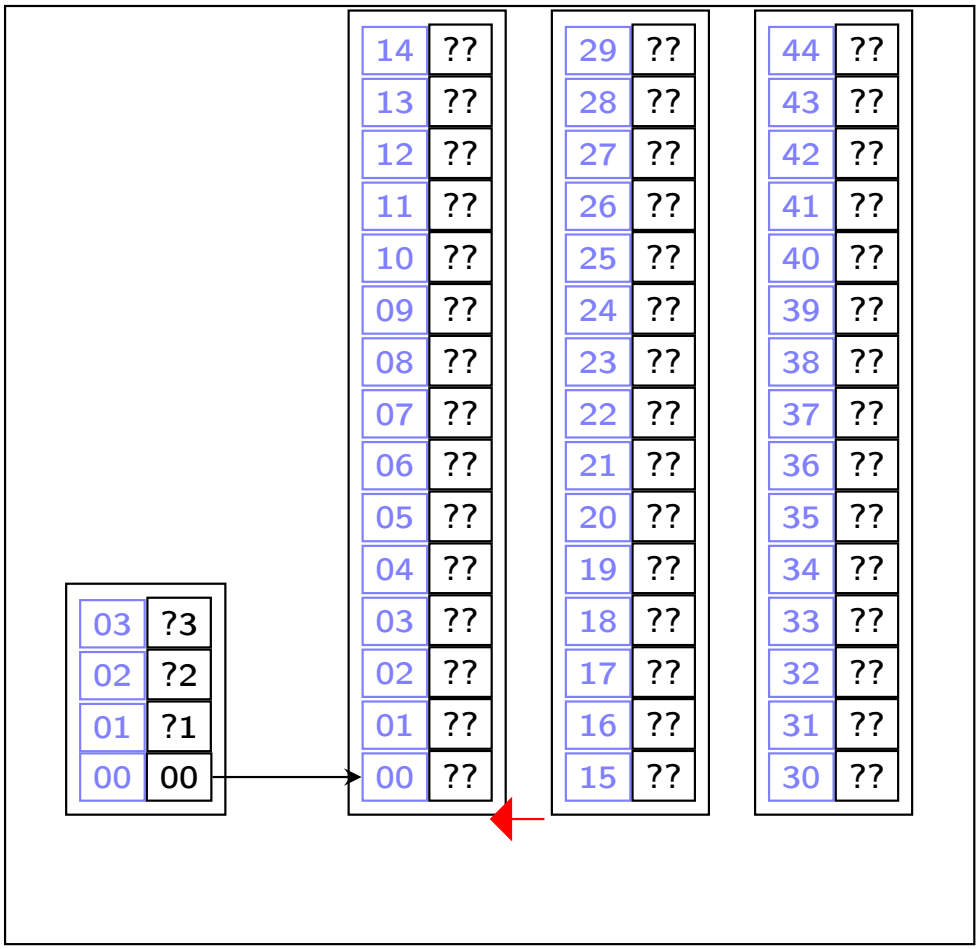
```
      INPP
      AMEM 1
      DSVS R00
R01:ENPR 1
      AMEM 1
      CRVL 1,-5
      CRCT 2
      CMME
      DSVF R02
      CRVI 1,-4
      CRCT 1
      SOMA
      ARMI 1,-4
      DSVS R03
R02:NADA
      CRVL 1,-5
      CRCT 1
      SUBT
      CREN 1,0
      CHPR R01,1
      CRVL 1,0
      ARMI 1,-4
      CRVL 1,-5
      CRCT 2
      SUBT
      CRVL 1,-4
      CHPR R01,1
R03:NADA
      CRVL 1,-5
      IMPR
      CRVI 1,-4
      IMPR
      DMEM 1
      RTPR 1,2
R00:NADA
      CRCT 0
      ARMZ 0,0
      CRCT 3
      CREN 0,0
      CHPR R01,0
      DMEM 1
      PARA
```



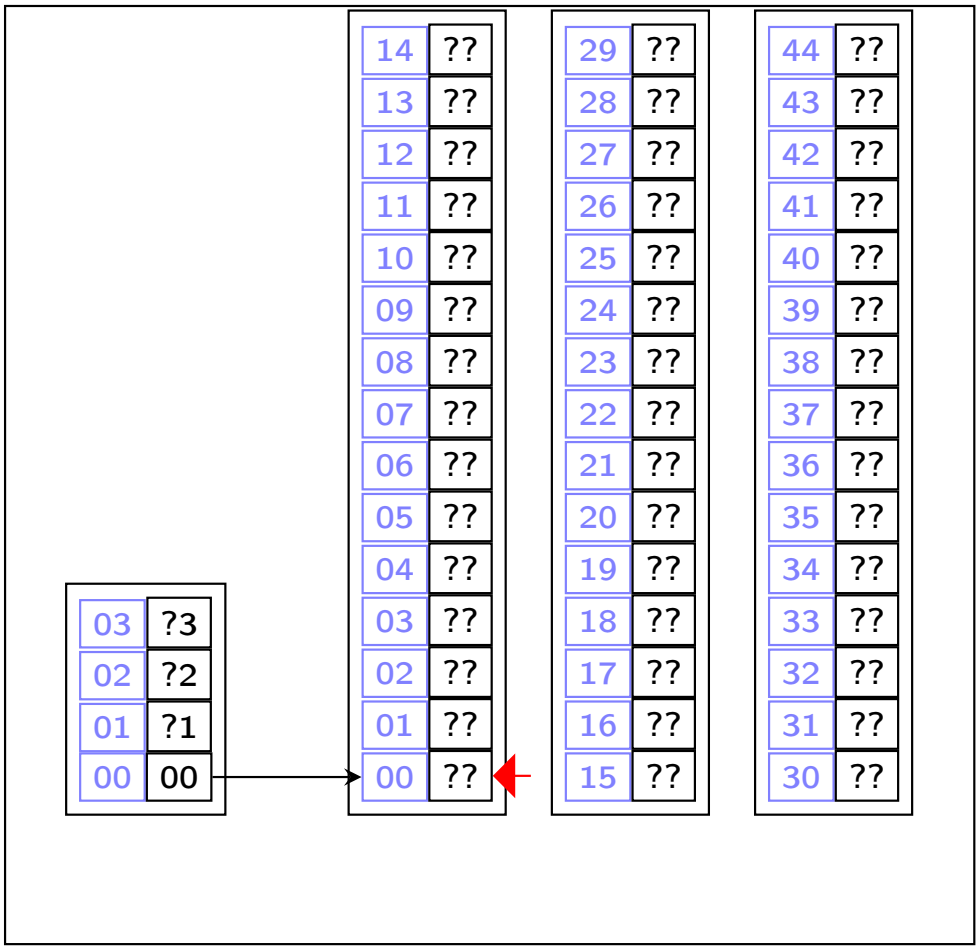
Execução

- Agora vamos examinar o que acontece em tempo de execução.
- O foco será a passagem de parâmetros por referência, os demais comandos serão vistos superficialmente.

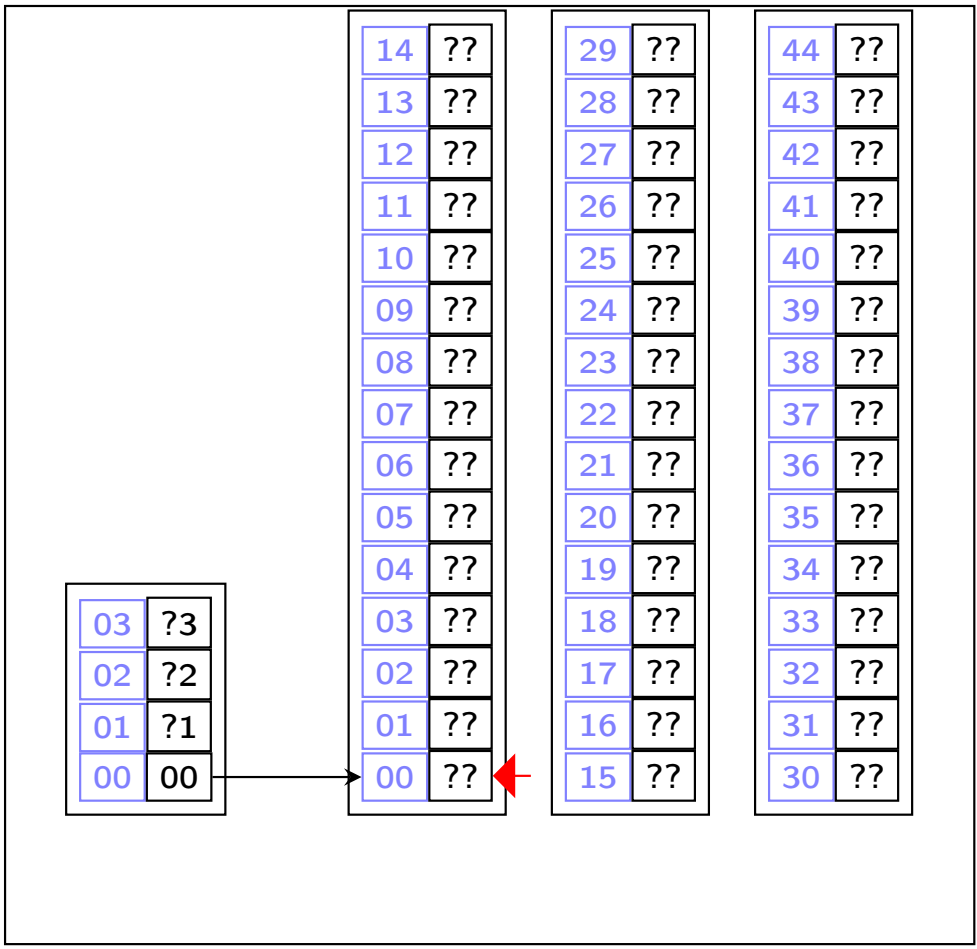
```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h);
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);
end.
```



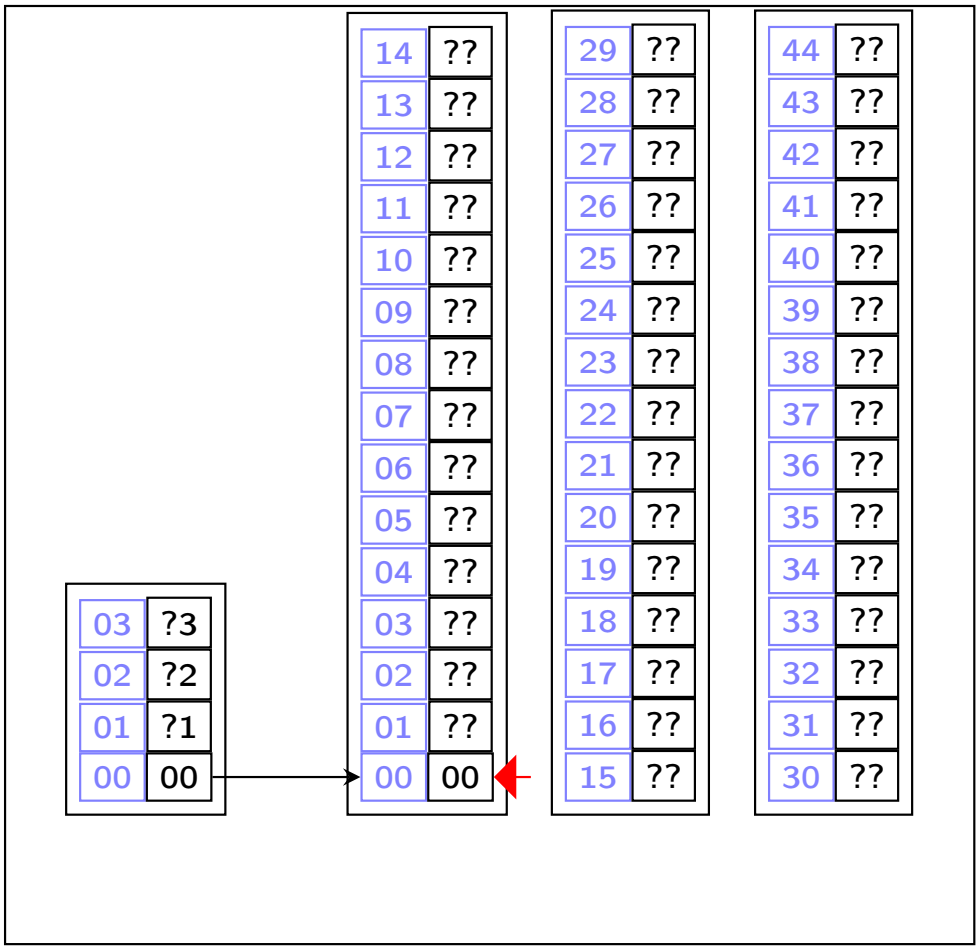
```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h);
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);
end.
```



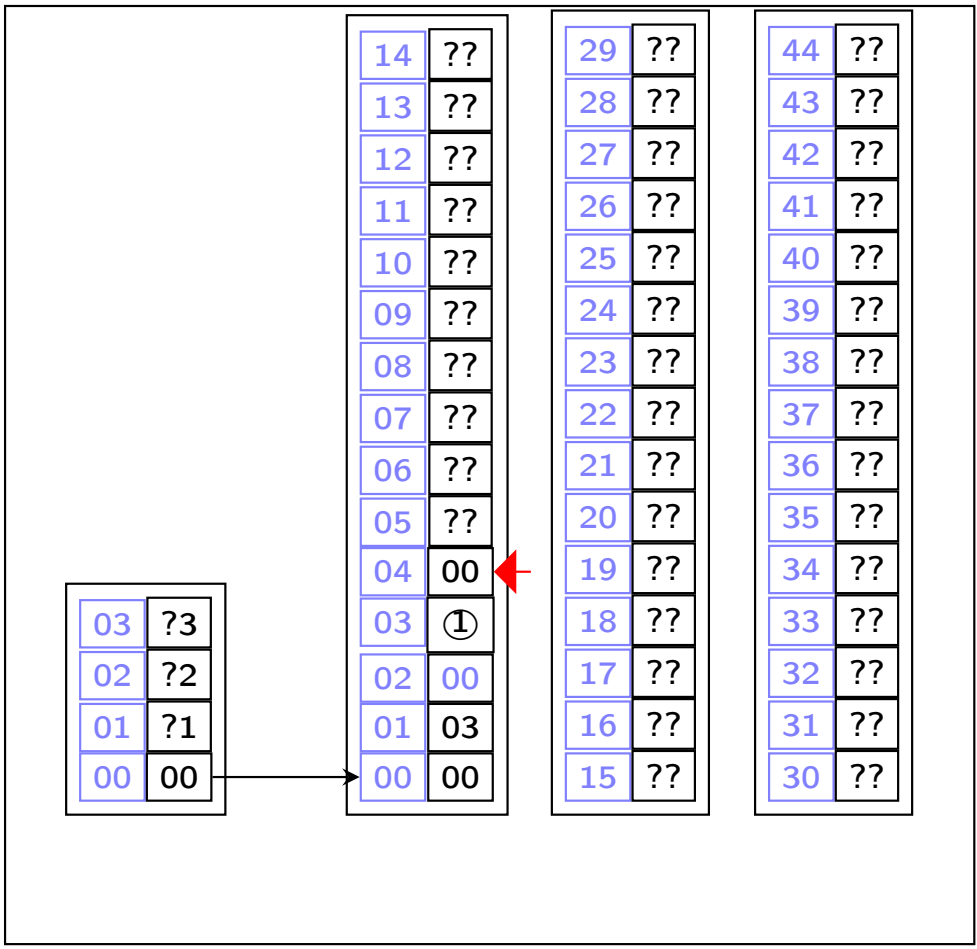
```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h);
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);
end.
```



```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h);
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);
end.
```

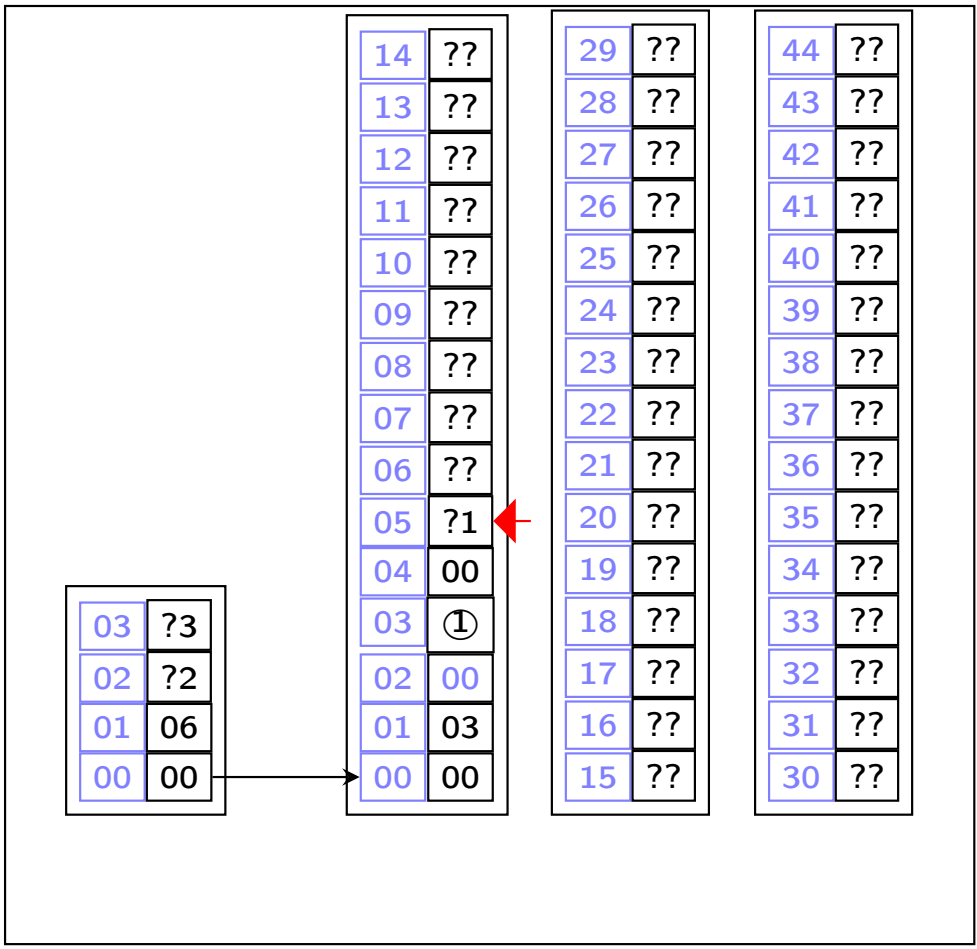


```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h);
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



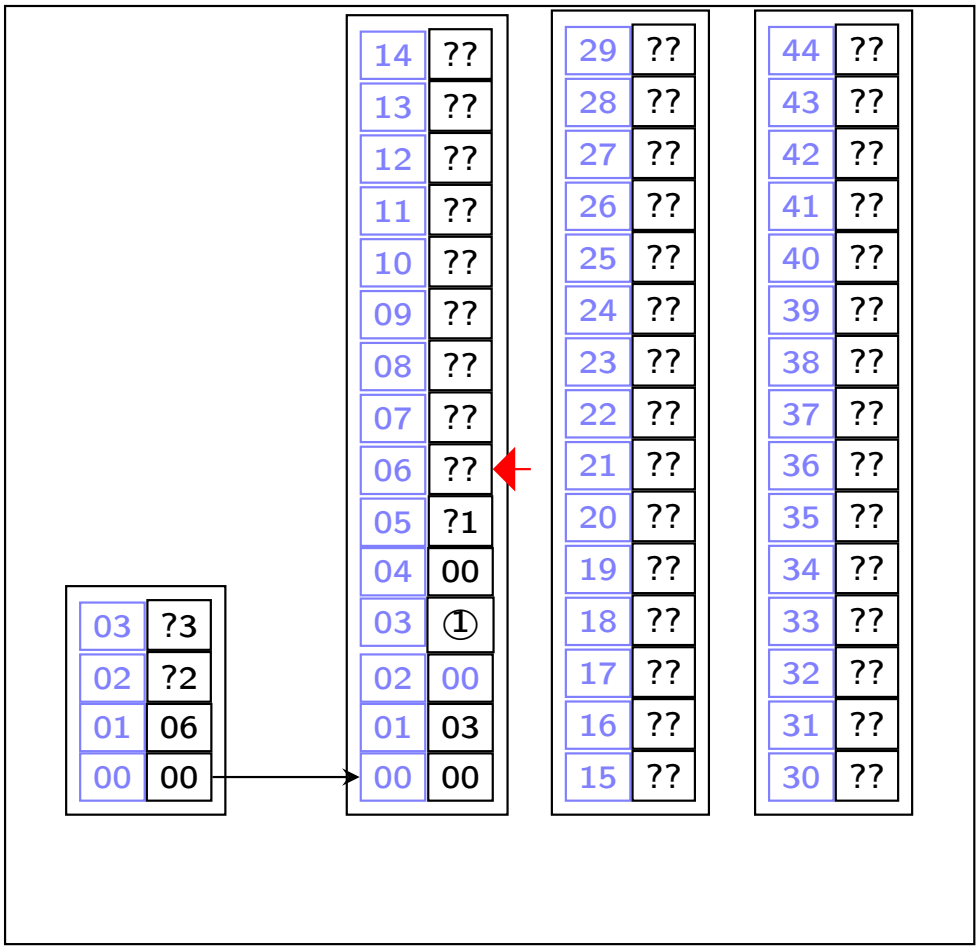
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h);
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



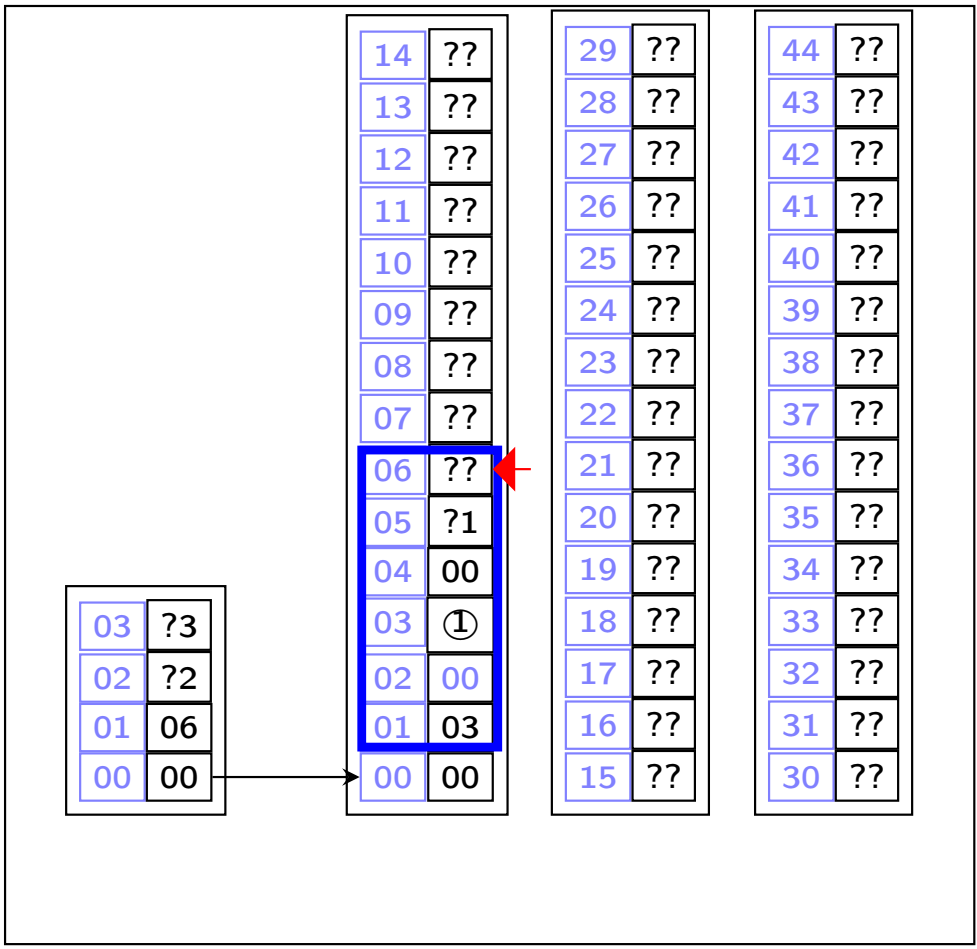
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h);
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



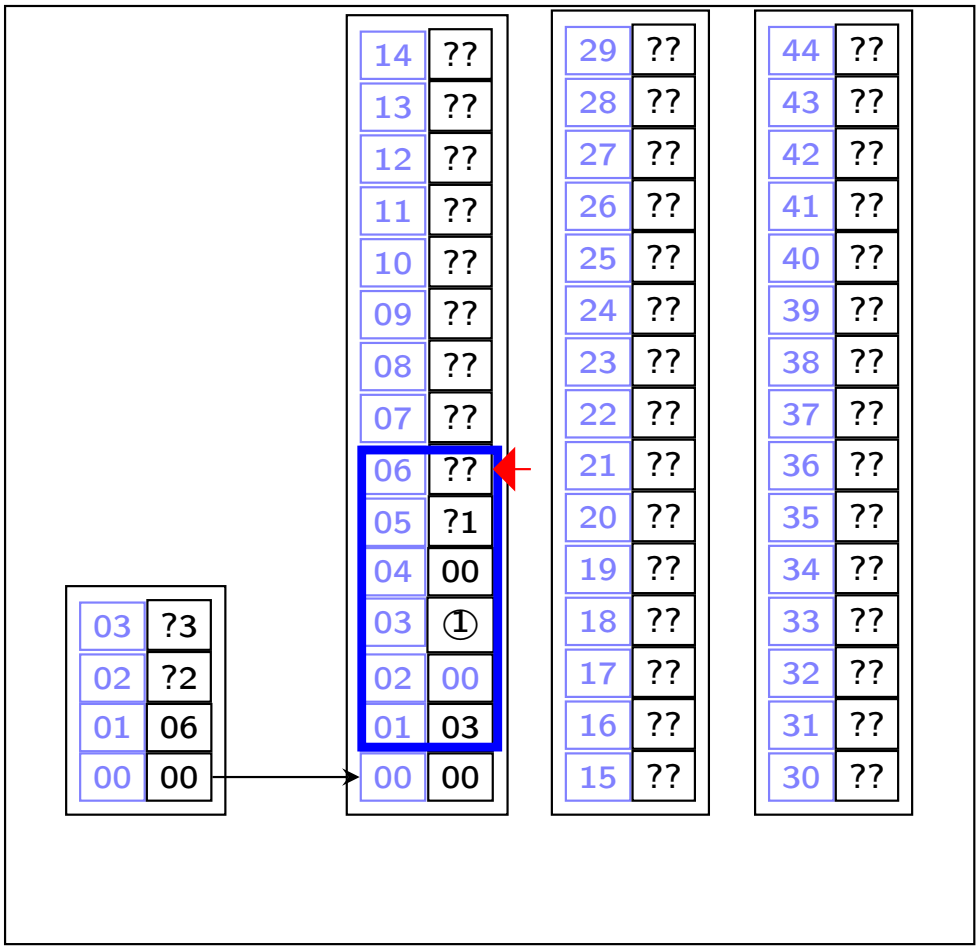
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h);
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



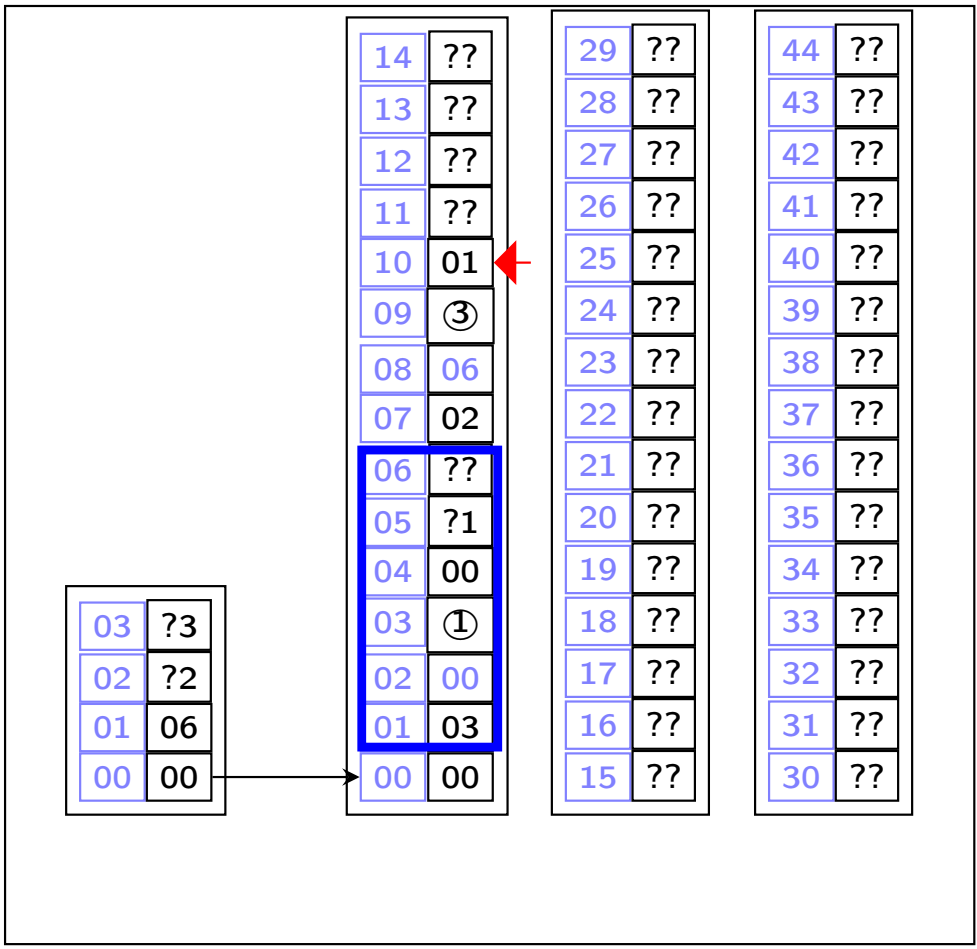
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h);
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



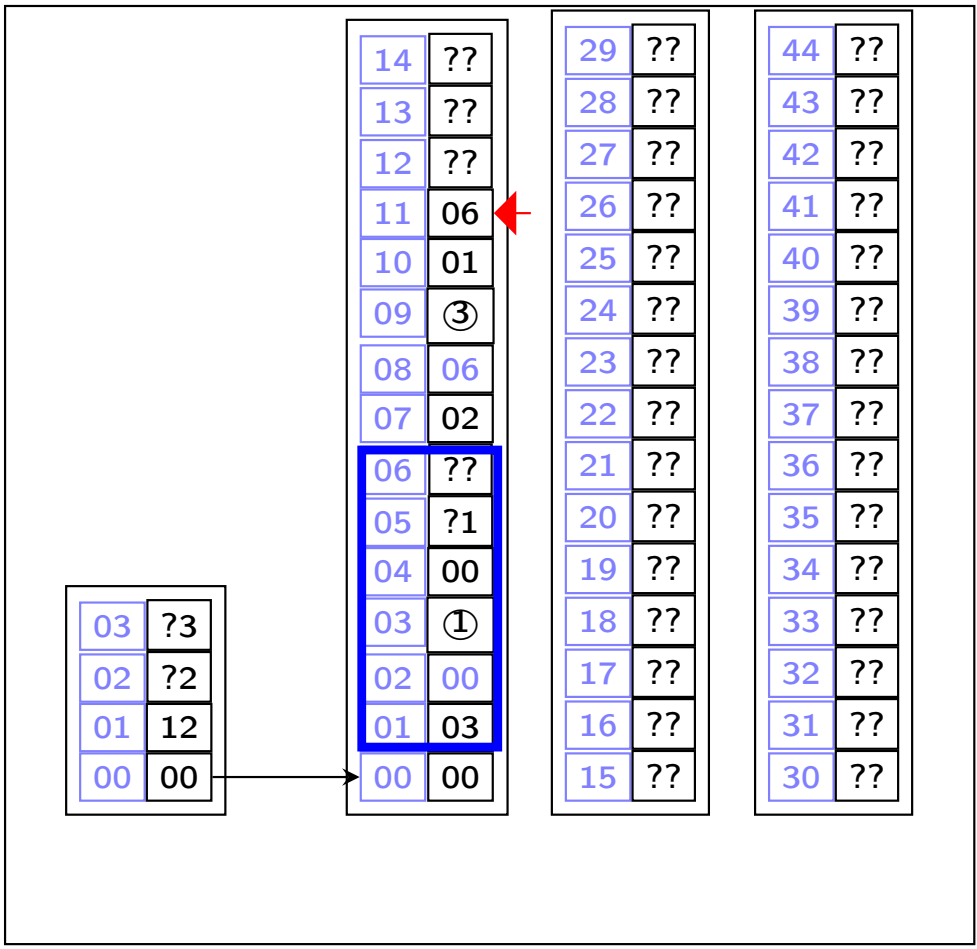
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



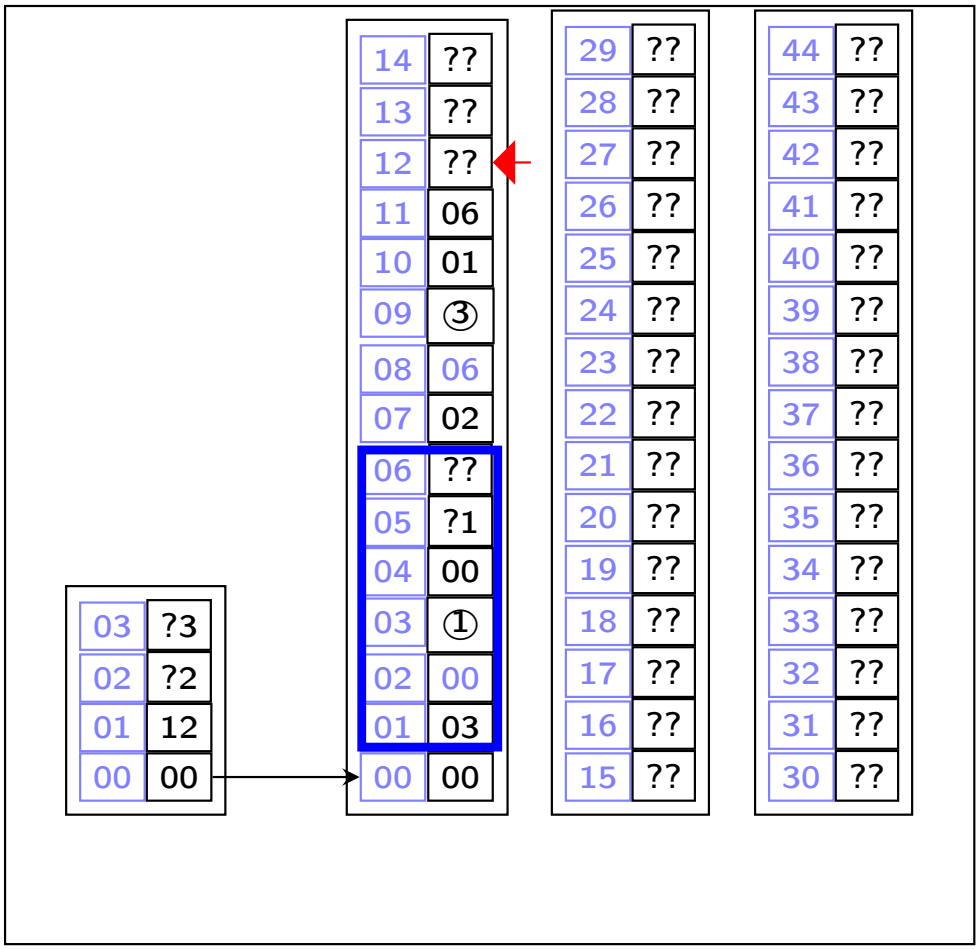
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



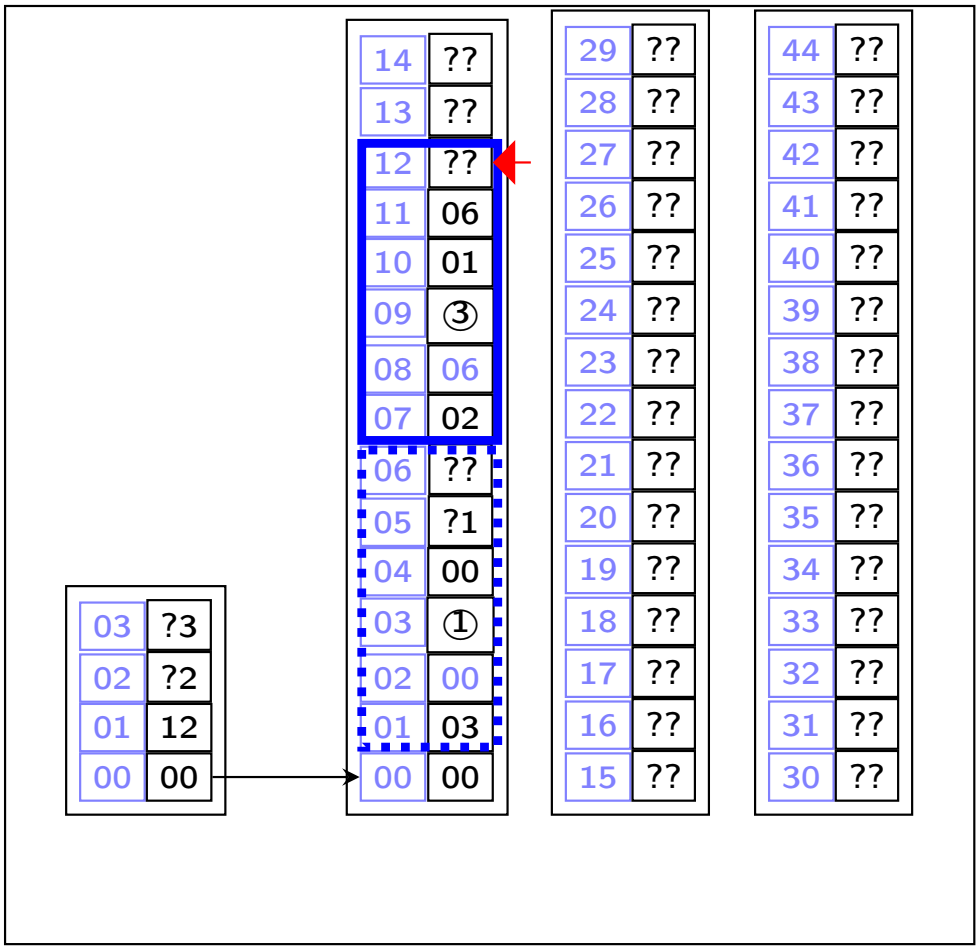
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



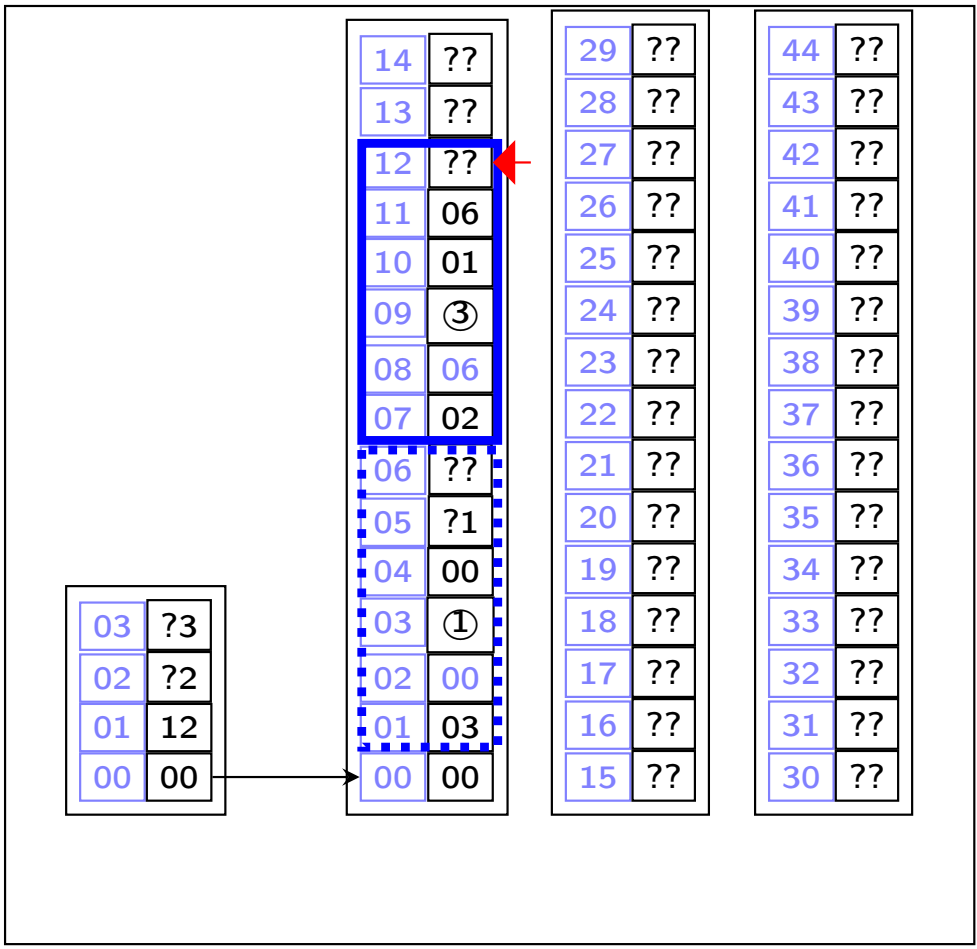
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



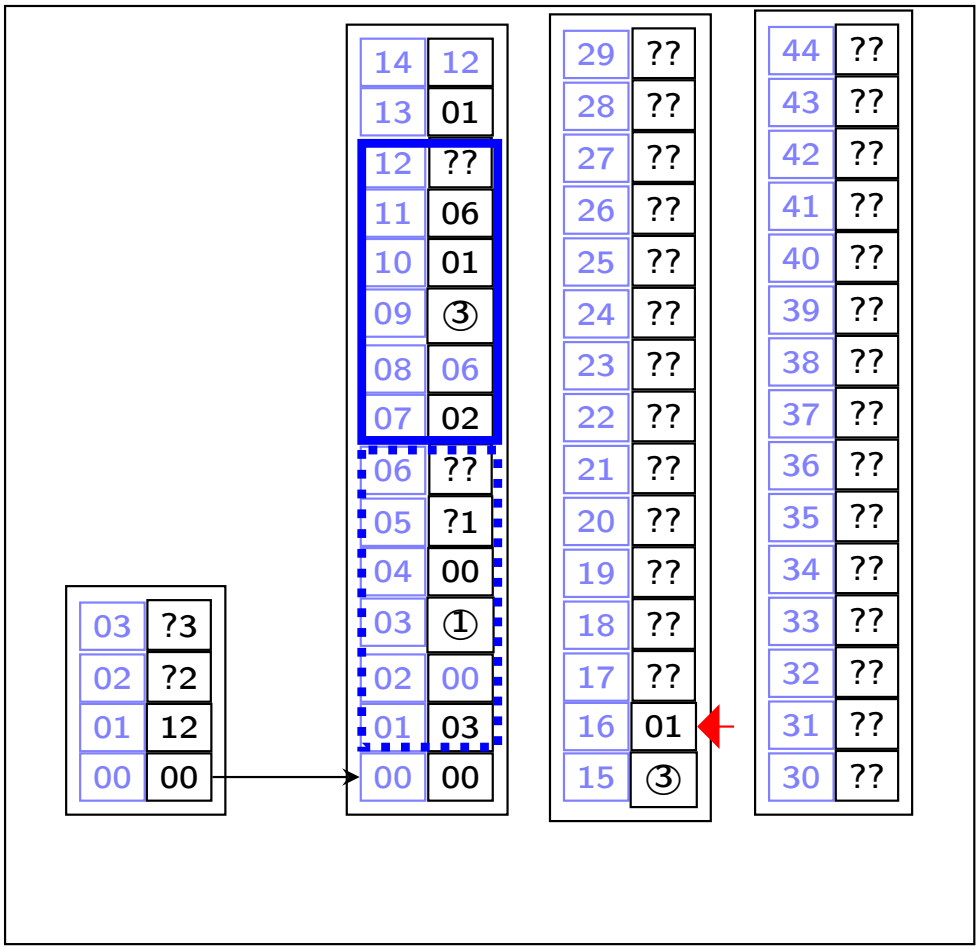
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



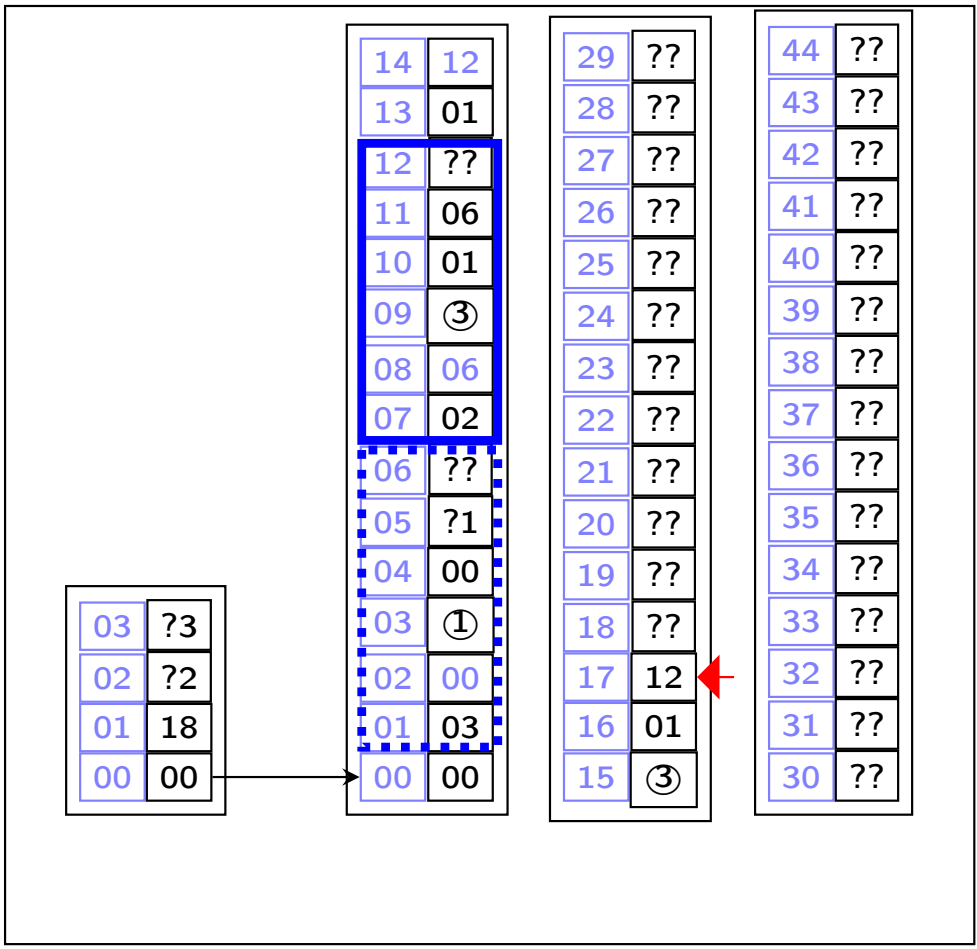
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



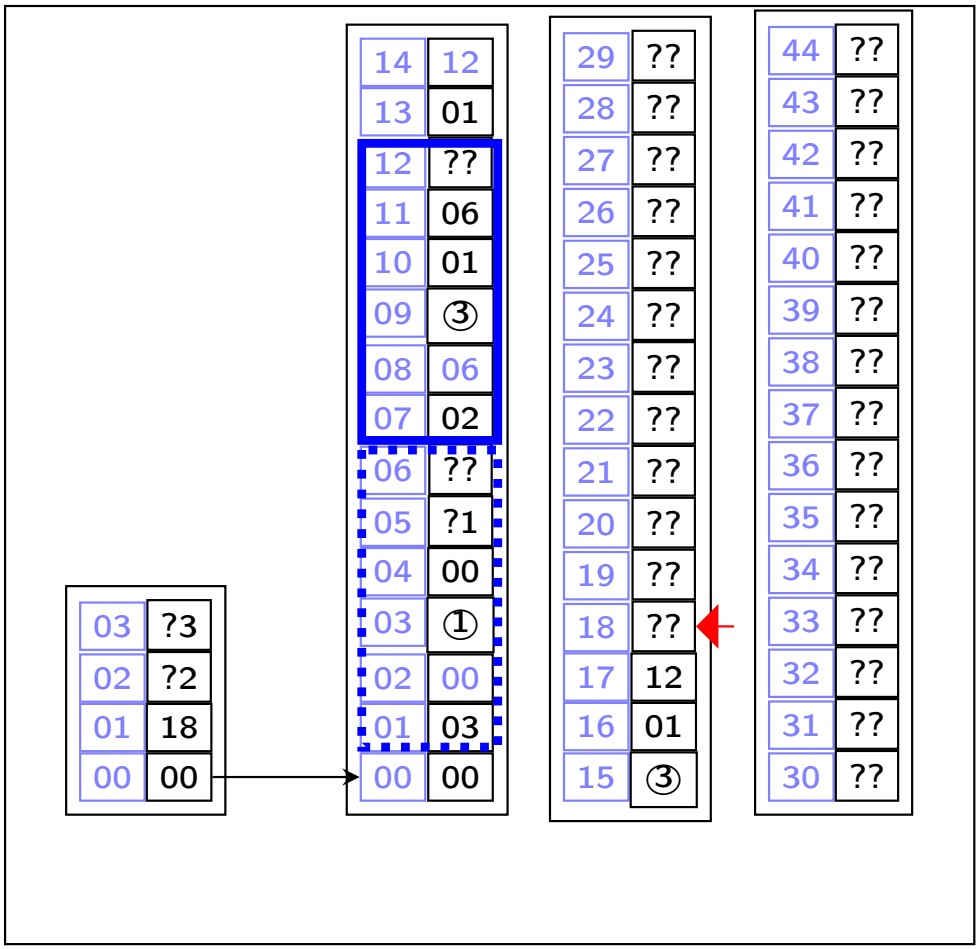
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



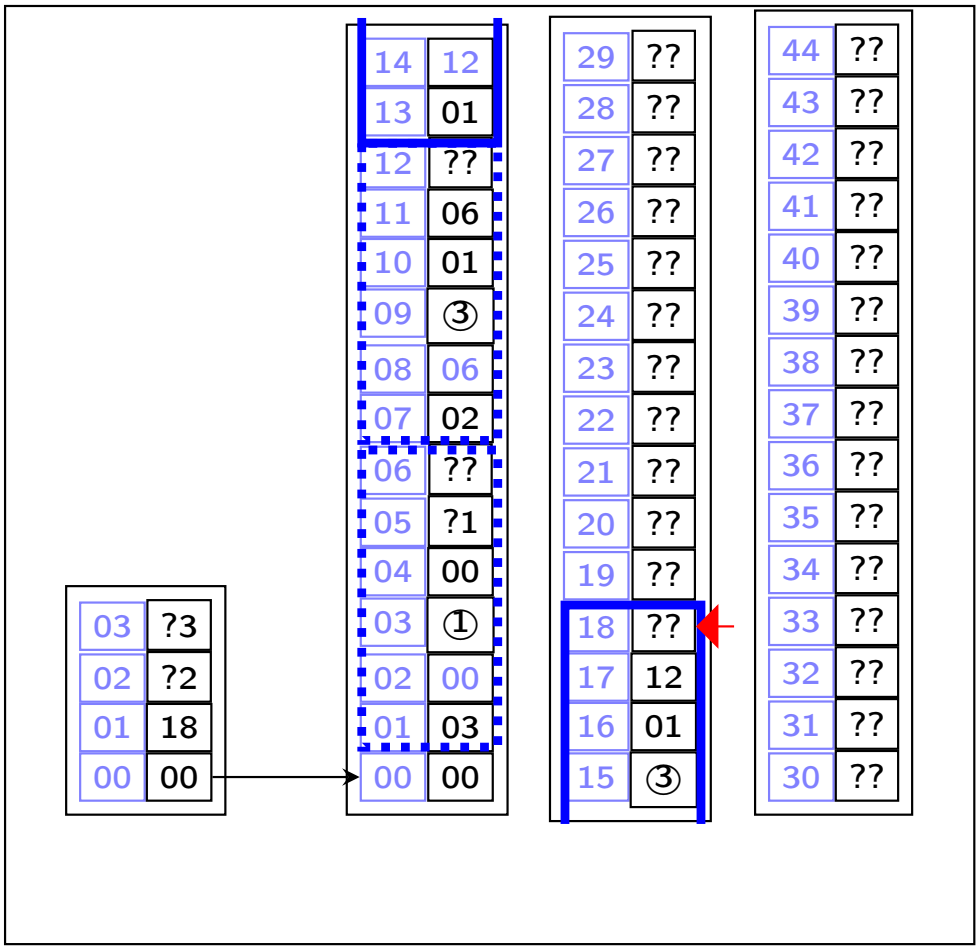
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



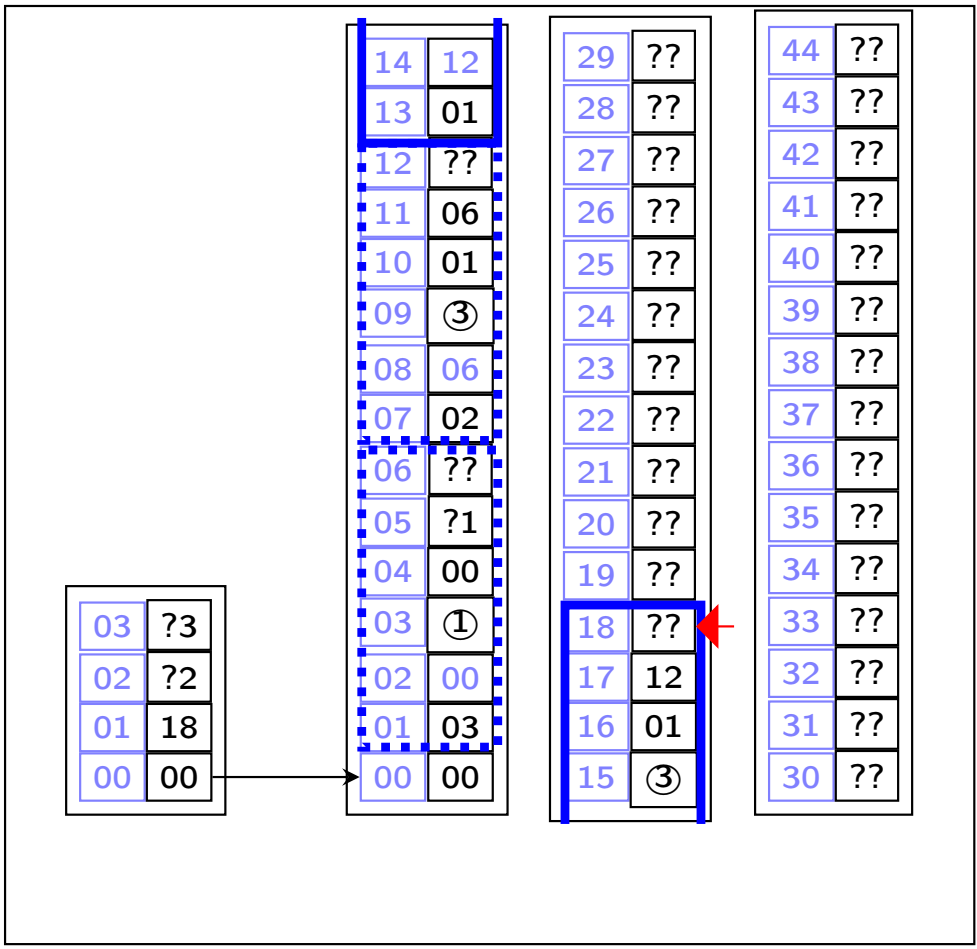
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



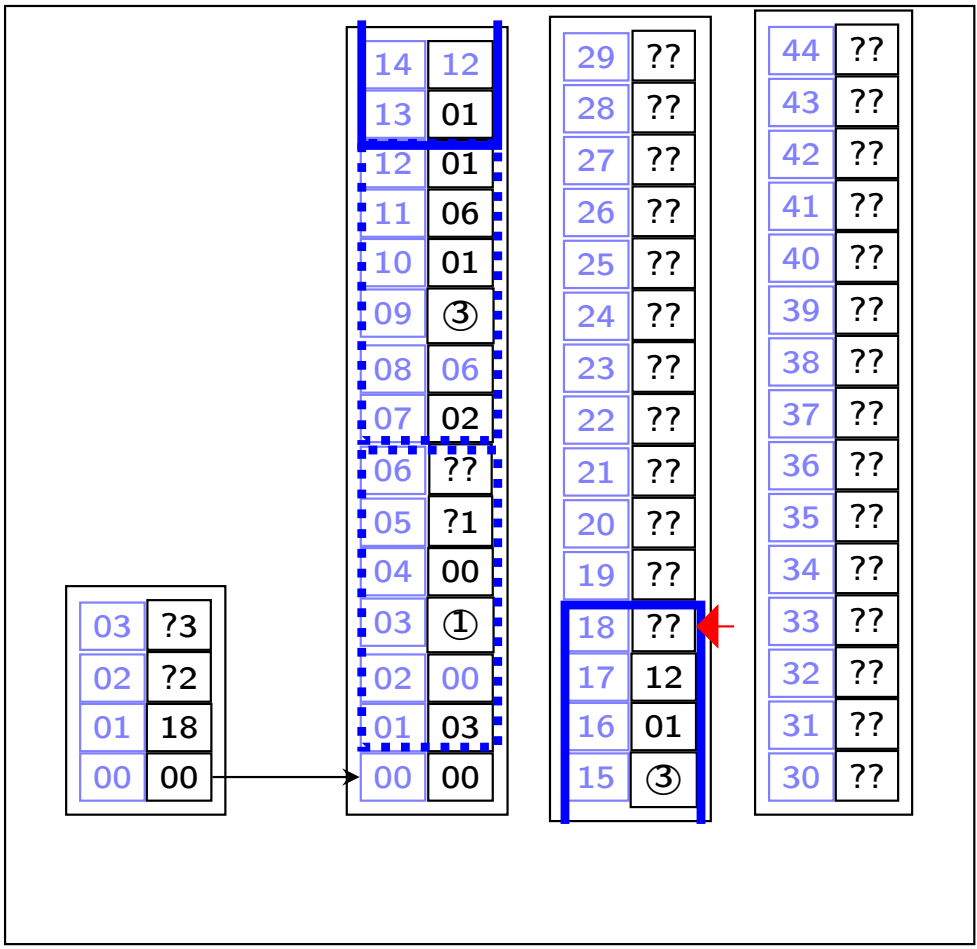
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



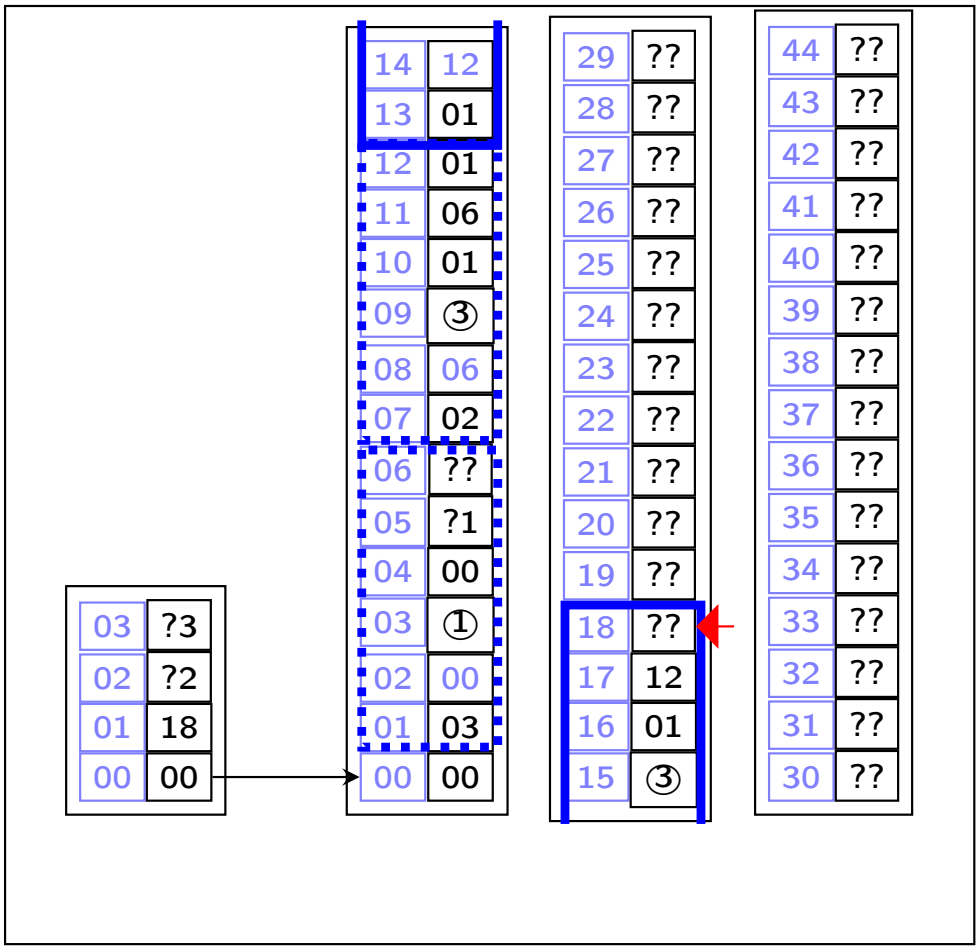
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



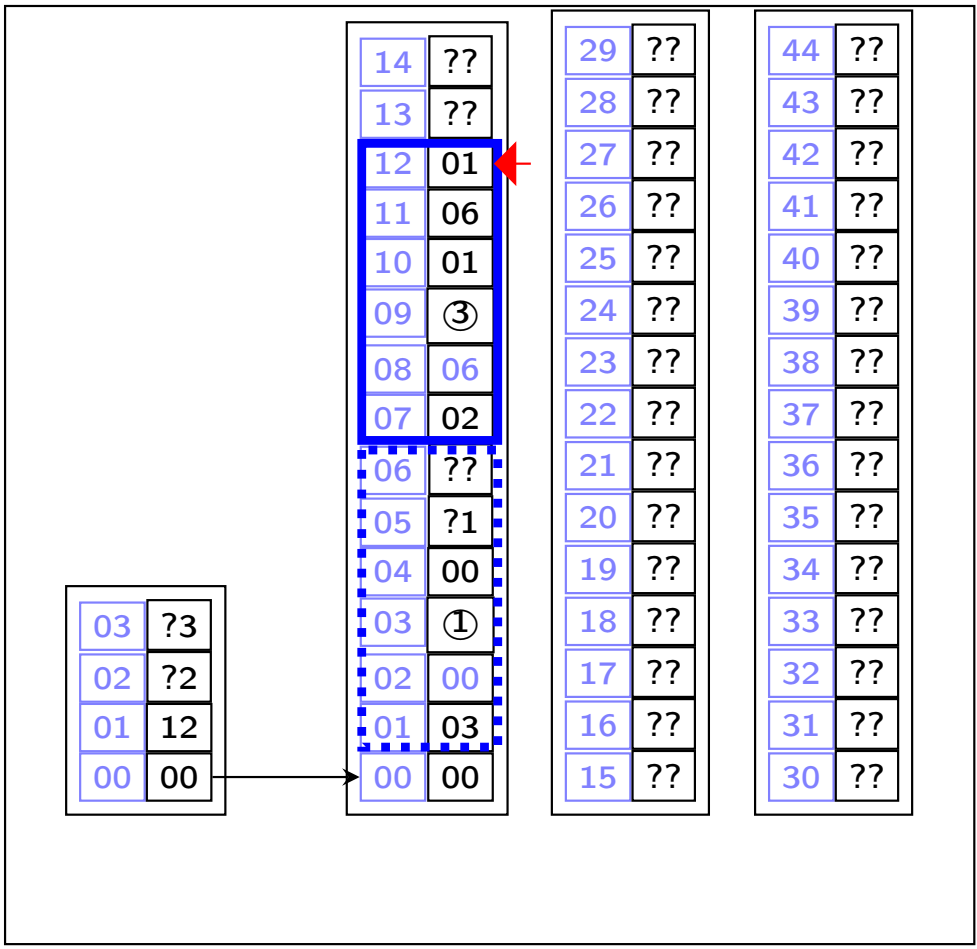
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



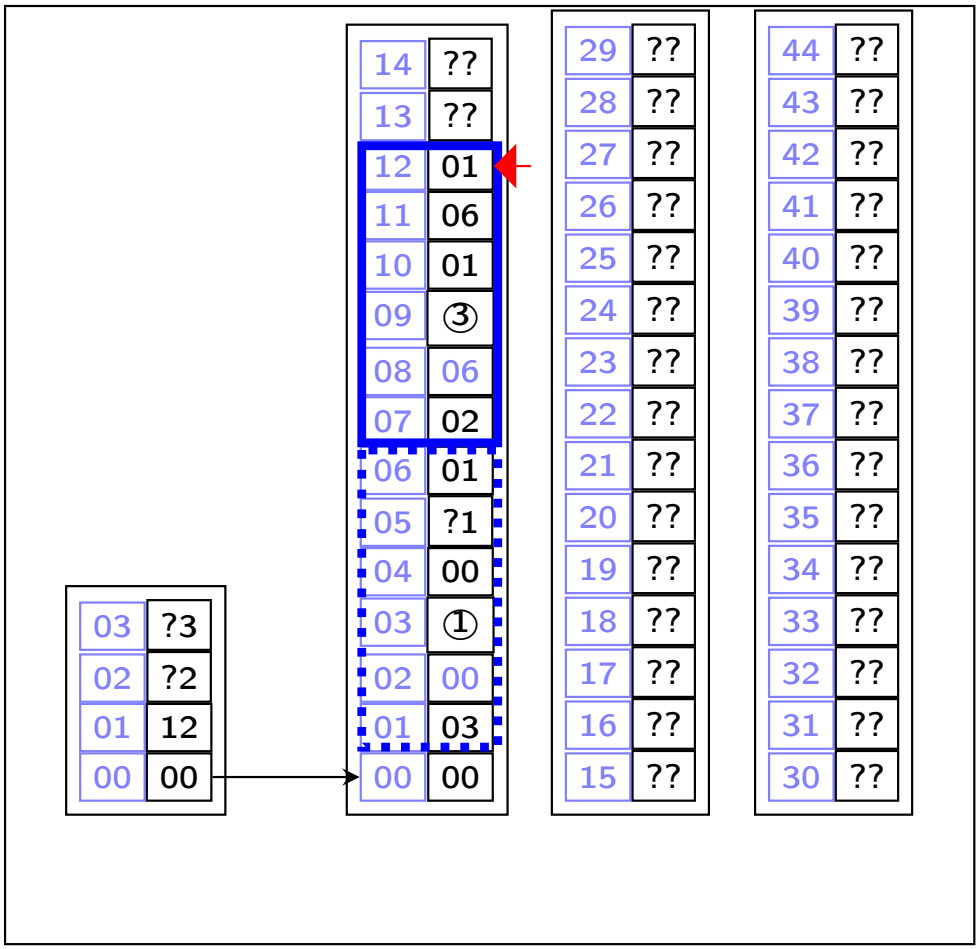
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



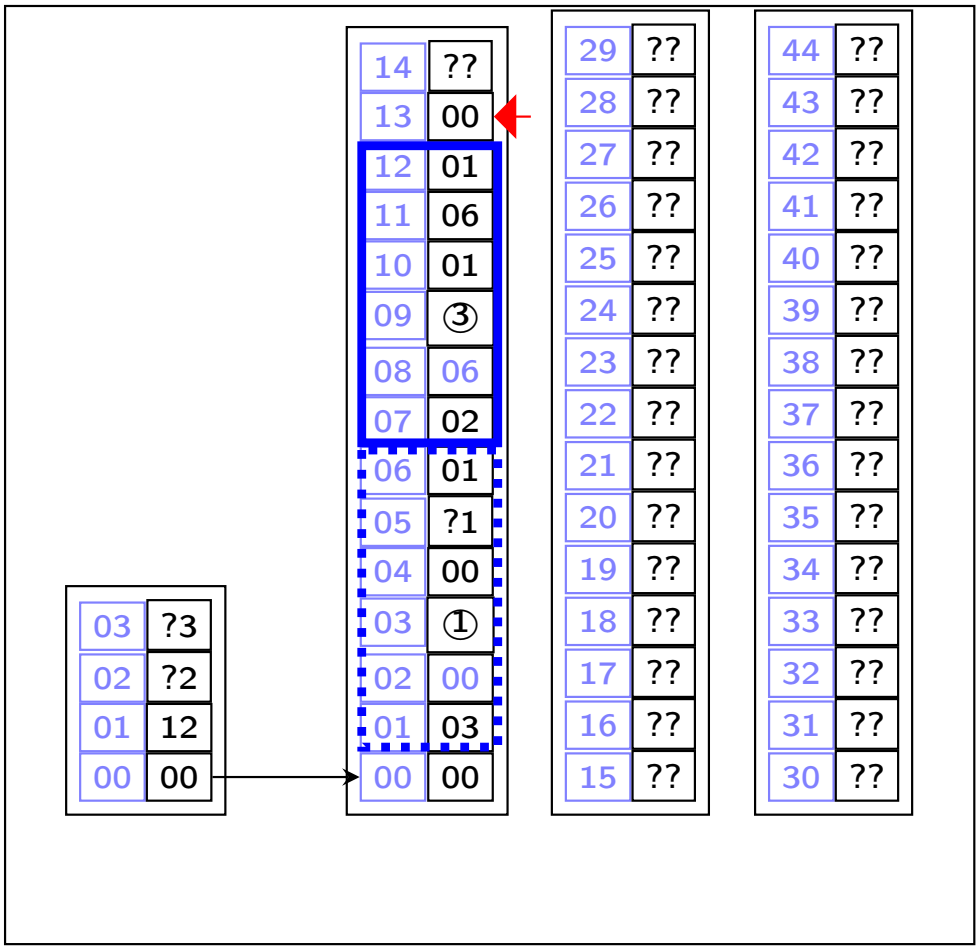
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



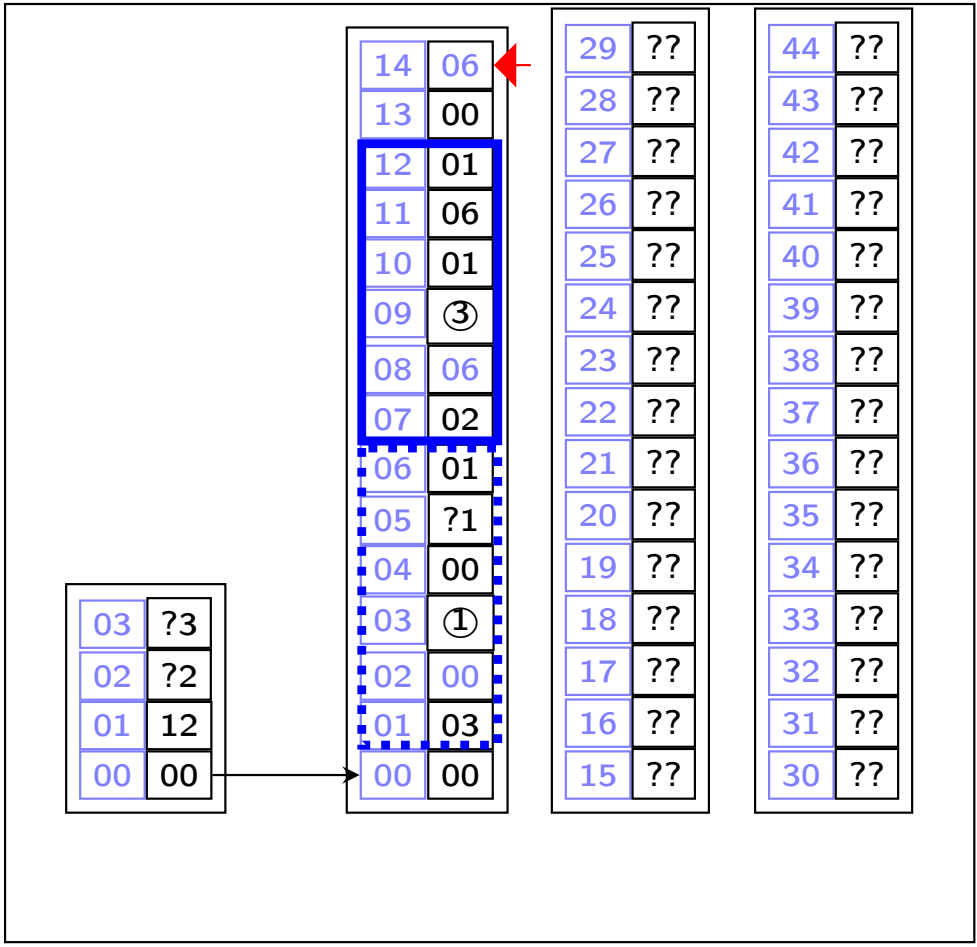
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
      p(n-2,g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



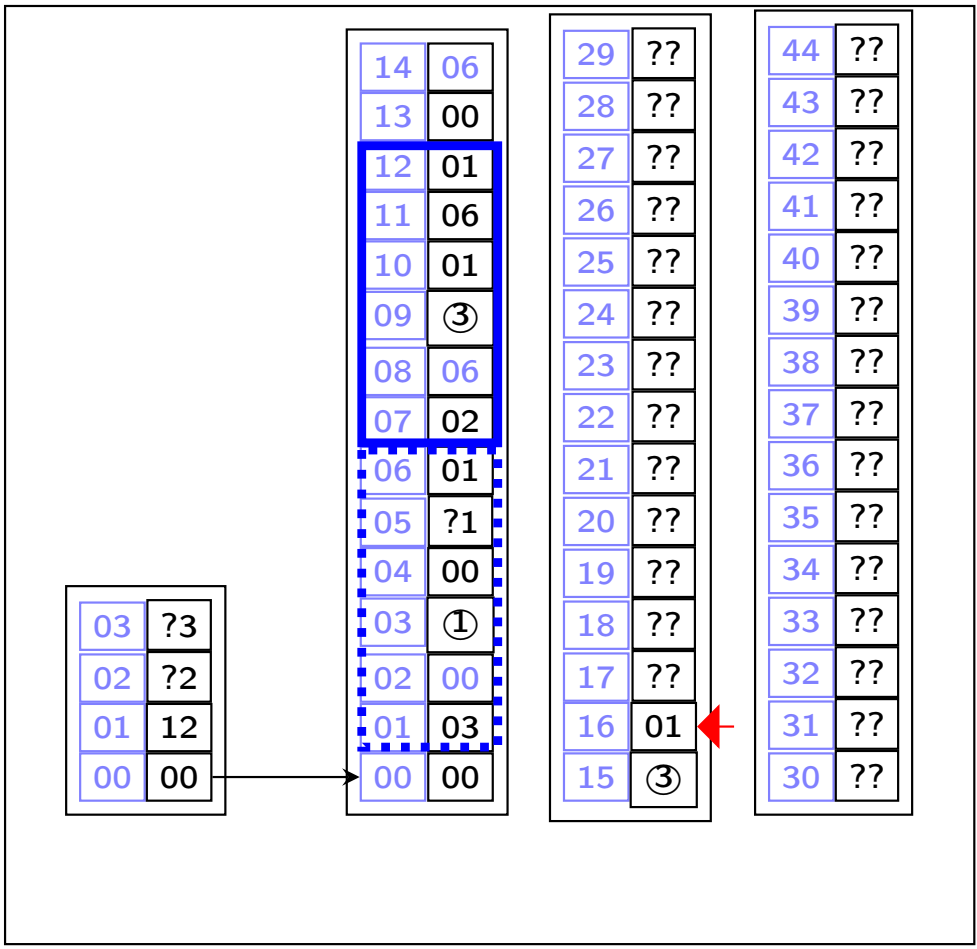
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
     p(n-2, g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



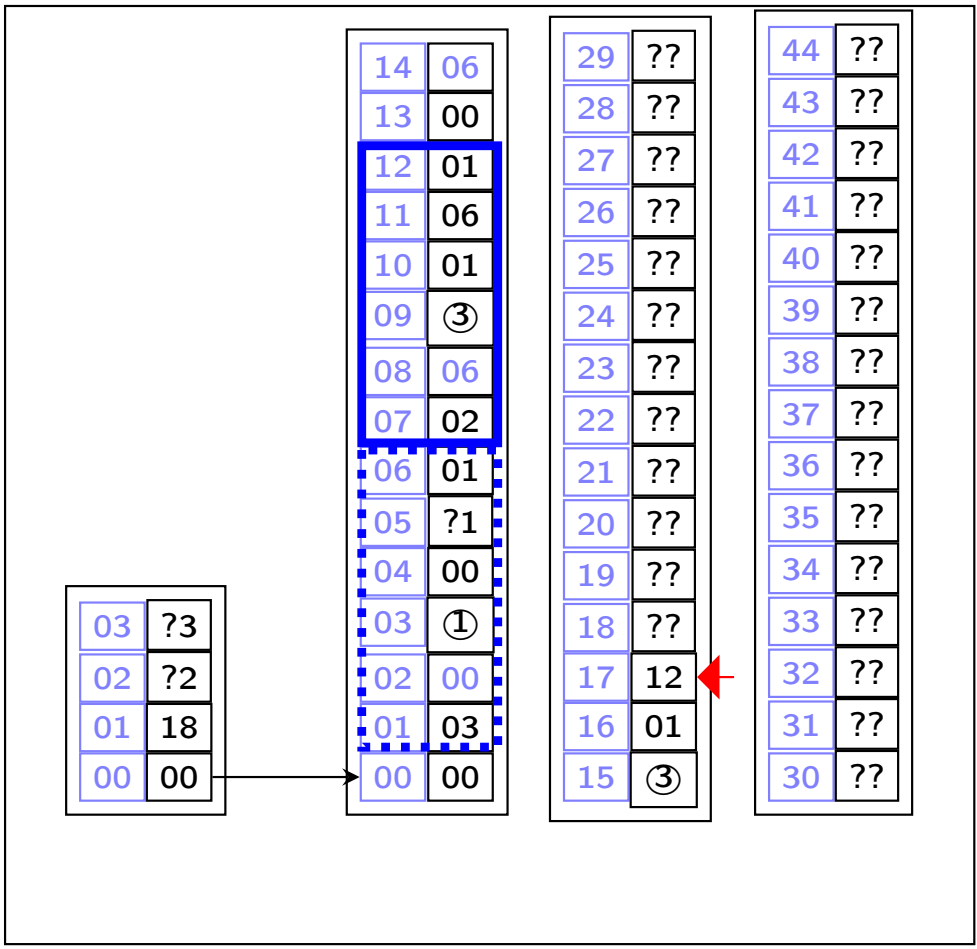
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
     p(n-2, g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



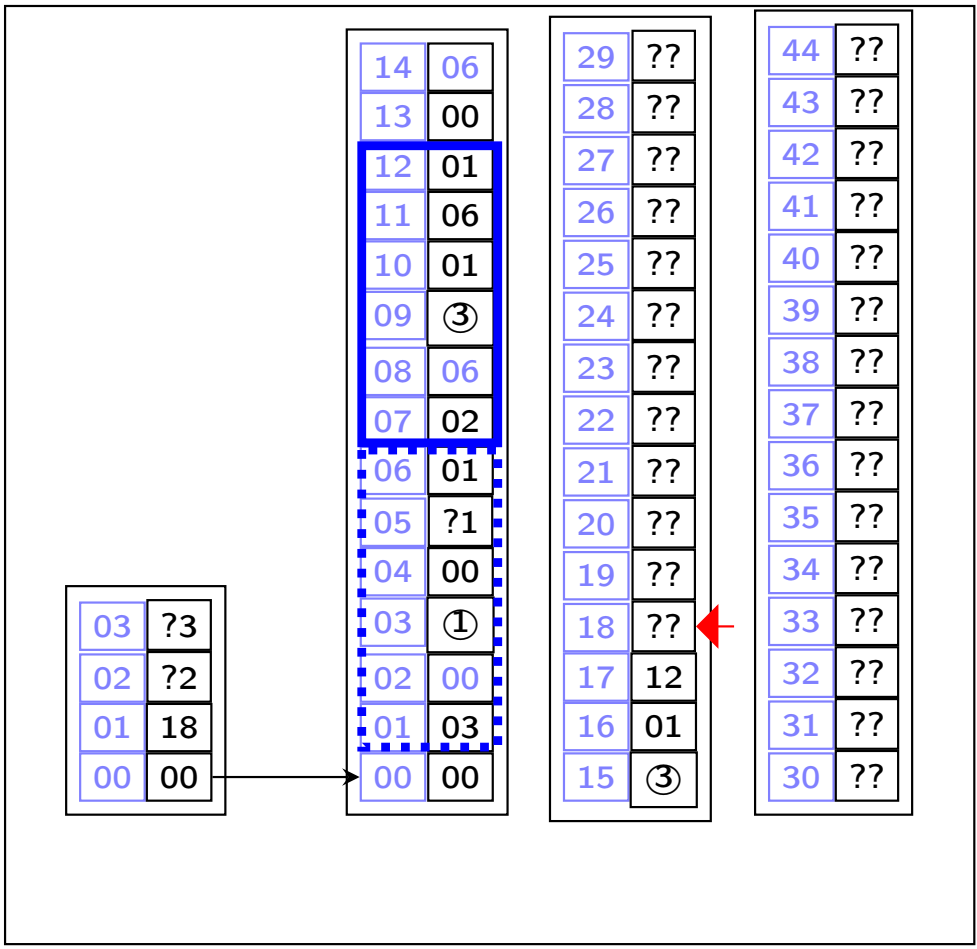
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
     p(n-2, g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



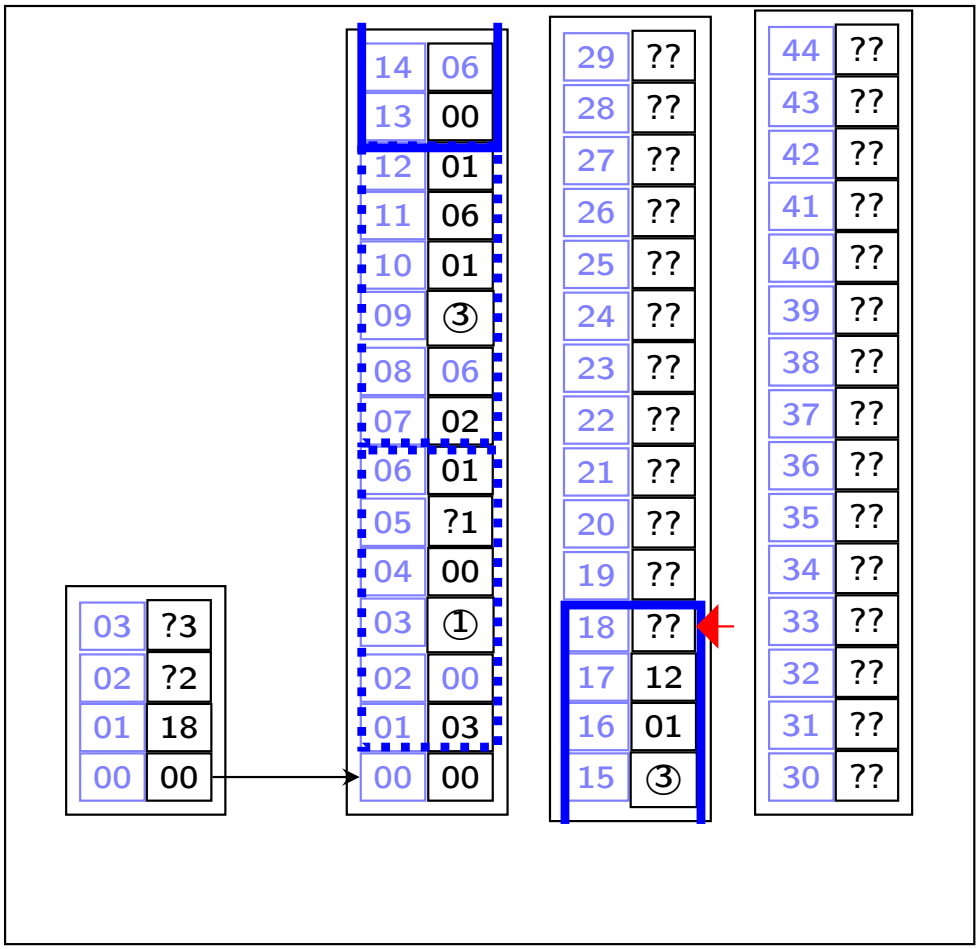
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
     p(n-2, g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



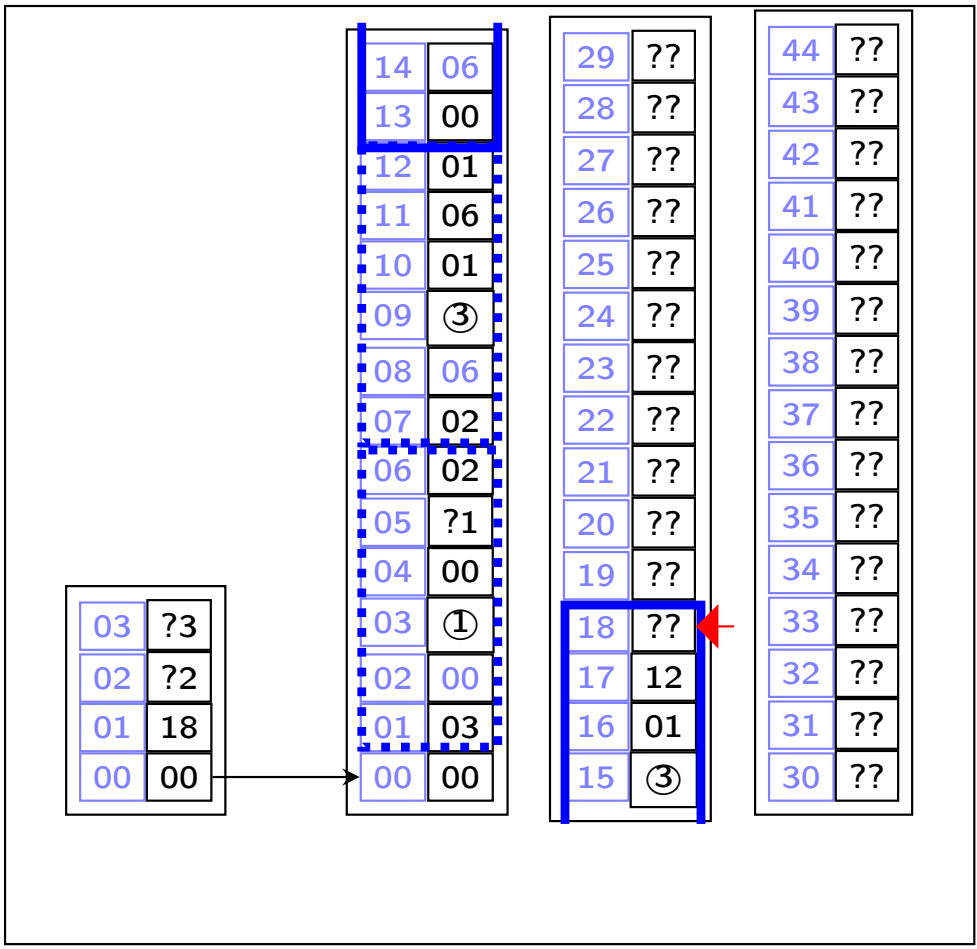
Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
     p(n-2, g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



Execução

```
program passRef (input, output);
var k: integer;
procedure p(n:integer; var g:integer);
var h:integer;
begin
   if (n<2)
      then g:=g+1
   else
   begin
      p(n-1,h)③;
      g:=h;
     p(n-2, g)
   end;
   write(n,g)
end;
begin
   k:=0;
   p(3,k);①
end.
```



Resumo

- A execução é muito longa, tornando inviável eu prosseguir aqui.
- Como exercício, prossiga esta execução até o fim.
- Concentre-se em questões abstratas como:
  - Encadeamento de registros de ativação através do D[k];
  - Conteúdo das informações gerenciais;
  - Quais variáveis estão alocadas, seu endereço léxico, sua posição no registro de ativação, e em caso de passagem por referência indique isso com uma seta (mantenha abstração).

Projeto

- Implemente passagem de parâmetro por referência no compilador.
- Problemas:
  - Nas regras que consomem identificadores, é necessário criar um mecanismo para saber se é para utilizar CRVL/ARMZ/CRVI/ARMI/CREN . Sugestão: faça uma análise exaustiva e estude como implementá-la.
  - A maior dificuldade está em saber qual instrução utilizar na hora de empilhar o parâmetro (na chamada do procedimento). A tabela a seguir apresenta um resumo.

Projeto

- Parâmetro Formal: `procedure p(pf1,...,pfn)`
- Parâmetro Real: `p(pr1,...,prn)`
- Dica: PF vlr: cópia / PF ref: endereço

	PF vlr	PF ref
VS		
PF vlr		
PF ref		

Projeto

- Parâmetro Formal: `procedure p(pf1,...,pfn)`
- Parâmetro Real: `p(pr1,...,prn)`
- Dica: PF vlr: cópia / PF ref: endereço

	PF vlr	PF ref
VS	CRVL	CREN
PF vlr	CRVL	CREN
PF ref	CRVI	CRVL

- Página para anotações

Licença

- Slides desenvolvidos somente com software livre:
  - LaTeX usando beamer;
  - Inkscape.
- Licença:
  - Creative Commons Atribuição-Uso Não-Comercial-Vedada a Criação de Obras Derivadas 2.5 Brasil License. `http://creativecommons.org/licenses/by-nc-nd/2.5/br/`