UNIVERSIDADE DE SÃO PAULO - ICMC
Departamento de Ciências de Computação
SCC 206 – Introdução a Compilação - 1° Sem /2010
PROFa: Sandra Aluisio
WIKI: http://wiki.icmc.usp.br/index.php/SCC-206

Trabalho Prático 4 -- Entrega: 6/7 na sala da professora

Implementar a geração de código para Pascal simplificado mais ALGUMAS extensões dos grupos (citadas abaixo), usando a versão mais atual das instruções da MEPA, dada em sala de aula.

Extensões a serem implementadas: **FOR, CASE, REPEAT, função, definição de constante, por seus respectivos grupos.**

Grupos que possuem as extensões: **definição de tipo, definição de array, real, string, char, Record e enumerado,** não precisam implementar a geração destas extensões. Usarão a gramática do PS básica. Isto porque só trabalharemos com variáveis SIMPLES e tipos inteiros e booleanos.

Todos devem testar a geração de código com a bateria de programas usados no livro texto do Kowaltowski: **Fig. 8.4, Fig 8.5, Fig 8.13, Fig 8.15, Fig 8.21** (listados abaixo).

- Os que tem funções devem também testar com o programa da **Fig 7.9**, listado abaixo.
- Os grupos que vão implementar suas extensões devem incluir programas com exemplos das extensões que são comandos FOR, REPEAT e CASE e definição de constante.
- Os que **não** vão implementar suas extensões, devem também testar com os programas das figuras **Fig 8.9, Fig 8.11**, listados abaixo.

OBS:
1) Gerar código dos programas acima usando a versão mais atual da MEPA, pois algumas soluções são de versões modificadas.
2) Não faremos a interpretação do código. Cuidado: a versão da MEPA a ser usada não inclui: Rótulos e Comandos de DESVIO; Passagem de procedimento e função como parâmetro; passagem por nome (corresponde a uma função sem parâmetro); Blocos com declarações locais. **Assim, RTPR é a que está nos slides do curso, pg 65 e CHPR é igual da pg 45:**

RTPR k, n **D[k] := M[s]** **i:= M[s-1]** **s := s – (n+2)**	**CHPR p** **s := s + 1** **M[s] := i + 1** **i := p**

Forma de Entrega:

- Programa fonte, arquivos de teste e arquivo de saída da geração: impressos. O relatório deve ser bem documentado e escrito.
- Código executável e instruções de como rodar: CD.

FIG 8.4 Programa sem procedimentos/funções

```
program ex1;
var n,k: integer;
    f1,f2,f3: integer;
begin
read(n);
f1:=0; f2:=1; k:=1;
while k <= n do
    begin
f3:= f1 + f2;
f1 :=f2; f2 := f3;
k := k + 1
end;
write(n,f1)
end.
```

Façam a tradução usando a última versão das instruções da MEPA, dada em sala de aula.

8.5 Programa com procedimento sem parâmetros

```
program ex2;
var x, y: integer;
procedure p;
    var z: integer;
    begin
            z := x; x := x -1;
            if z > 1 then p
            else y := 1;
            y := y * z
    end;
begin
    read(x);
    p;
    write(x,y)
end.
```

Façam a tradução usando a última versão das instruções da MEPA, dada em sala de aula.

FIG 8.9 Programa com 2 procedimentos recursivos sem parâmetros

```
program ex7;
var x,y: integer;
procedure p;
   var z: integer;
   begin
     z:= x; x:=x-1;
     if z > 1 then p else y:= 1;
     y := y * z;
   end;
   procedure q;
      var s,t: integer;
      begin
        s:= x; t := x – 1; x := t;
        if s = 0 then y := 1
        else if ((s div 2)*2) = s
               then q else p;
        y:= y * s
      end;
  begin
    read(x);
    q;
    write(y)
  end.
```

Façam a tradução usando a última versão das instruções da MEPA, dada em sala de aula.

FIG 8.11 Programa com 2 procedimentos recursivos sem parâmetros, encaixados

```
program ex8;
var x,y: integer;
procedure p;
  var z: integer;
  procedure q;
    var s: integer;
    begin
      s:=z-1; x:=x-1;
      if s>1 then p else y := 1;
      y := y * s
    end;
 begin
   z := x; x := x – 1;
   if z >  1 then q else y :=1;
   y:= y * z
 end;
begin
   read(x);
   p;
   write(y)
end.
```

Façam a tradução usando a última versão das instruções da MEPA, dada em sala de aula.

FIG 8.13 Programa com procedimento que possui parâmetro passado por valor

```
program exemplo9;
var x,y: integer;
procedure p (t:integer);
var z: integer;
begin
     if t > 1 then p(t-1)
     else y := 1
     z := y;
     y := z * t
end;
Begin
     read(x);
    p(x); write(x,y)
End.
```

Façam a tradução usando a última versão das instruções da MEPA, dada em sala de aula.

FIG 8.15 Programa com procedimento que possui parâmetro passado por referência

```
program exemplo;
var x, y: integer;
procedure p (var s: integer);
var z: integer;
Begin
    if s = 1 then y := 1
    else
    begin
      z := s – 1;
      p(z);
      y := y * s
    end
End;
Begin
    x := 4;
    p(x);
    write(x,y)
End.
```

Façam a tradução usando a última versão das instruções da MEPA, dada em sala de aula.

FIG 8.21 Programa com procedimento que possui parâmetros passados por valor e por referência

```
Program ex10;
Var k: integer;
Procedure p(n: integer; var g:
integer);
Var h: integer;
Begin
If n < 2 then g := g + n
else begin
    h:=g;
    p(n-1, h);
    g :=h;
    p(n-2,g);
      end;
Write(n,g)
End;
Begin
k:= 0; p(3,k)
end.
```

K: 0,0
N:1,-4
G:1,-3
H:1,0

```
INPP
AMEN 1
DSVS  L1
L2  ENPR      1
AMEN 1
CRVL  1,-4
CRCT  2
CMME
DSVF  L3
CRVI   1,-3
CRVL  1,-4
SOMA
ARMI   1,-3
DSVS  L4
L3       NADA
CRVI   1,-3
ARMZ 1,0
CRVL  1,-4
CRCT  1
SUBT
CREN 1,0
CHPR  L2
CRVL  1,0
ARMI   1,-3
CRVL  1,-4
CRCT  2
SUBT
CRVL  1,-3
CHPR  L2
L4       NADA
CRVL  1,-4
IMPR
CRVI   1,-3
IMPR
DMEN 1
RTPR  1,2  {remove os 2}
L1 NADA   { p.p.}
CRCT  0
ARMZ 0,0
CRCT  3
CREN 0,0
CHPR  L2
DMEN 1
PARA
```

FIG 7.9 Programa com função que possui parâmetros passados por valor e por referência

```
Programa ex5;
Var m: integer;
Function f(n:integer; var k:
integer):integer;
Var p,q: integer;
Begin
    if n <2 then
            begin
              f := n; k := 0
            end
    else begin
          f := f(n-1,p) + f(n-2,q);
          k := p + q + 1
         end;
    write(n,k)
End;
begin
write(f(3,m),m)
end.
```

```
m: 0,0
f: 1,-5
n:1,-4
k:1,-3
p: 1,0
q: 1,1
```

```
 INPP
AMEM 1
DSVS  L1
L2        ENPR        1
AMEM 2
CRVL  1,-4
CRCT  2
CMME
DSVF  L3
CRVL  1,-4
ARMZ 1,-5
CRCT  0
ARMI  1,-3
DSVS  L4
L3              NADA
AMEM 1
CRVL  1,-4
CRCT  1
SUBT
CREN 1,0
CHPR            L2
AMEM 1
CRVL  1,-4
CRCT  2
SUBT
CREN 1,1
CHPR            L2
SOMA
ARMZ 1,-5
CRVL  1,0
CRVL  1,1
SOMA
CRCT  1
SOMA
ARMI  1,-3
L4              NADA
CRVL  1,-4
IMPR
CRVI  1,-3
IMPR
DMEM 2
RTPR  1,2
L1              NADA
AMEM 1
CRCT  3
CREN 0,0
CHPR            L2
IMPR
DMEM 1
PARA
```