



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 e definição de tipos.**

Testar a geração de código com a bateria de programas usados no livro texto do Kowaltowski:

- Fig. 8.4, Fig 8.5, Fig 8.9, Fig 8.11, Fig 8.13, Fig 8.15, Fig 8.21 e, para os que têm funções a Fig 7.9. Alguns destes estão abaixo
- Incluam exemplos das extensões que são comandos FOR, REPEAT e CASE e a função, além de definição de tipo e de constante, para os respectivos grupos.
- Só trabalhem com variáveis SIMPLES, assim os grupos não precisam gerar código para variáveis indexadas nem campos de Record, nem incluir os tipos: real, char, string, Record, array, enumerado.
- Gerar código dos programas acima usando a versão mais atual da MEPA, pois algumas soluções são de versões modificadas.

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 62:

RTPR k,n (MODIFICADA)

D[k] := M[s]

i := M[s-1]

s := s – (n+2) remove os n par

- **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

```
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.
```

FIG 8.5

```
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.
```

FIG 8.13

```
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.
```

FIG 8.15

```

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.

```

FIG 8.21

```

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.

```

FIG 7.9

```

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.

```