

# Conceptos y Paradigmas de lenguajes de Programación 2016

## Práctica Nro. 5 Pilas de ejecución

**Objetivo:** Interpretar como se organiza la memoria de datos durante la ejecución de un programa con llamados a subrutinas.

**Ejercicio 1:** Explique claramente cual es la utilidad del registro de activación y que representan cada una de sus partes.(Basado en el modelo debajo detallado)

Modelo de registro de activación

Head (prog principal)
Pto retorno
EE (enlace estático)
ED (enlace dinámico)
Variable 1
Variable 2
... (mas variables)
Param1
Param2
... (mas parametros)
Proc 1
Proc 2
.....
Func 1
Func 2
.....
Valor de retorno

**Ejercicio 2:** Dado el siguiente programa escrito en Pascal-like, continuar la realización de las pilas de ejecución hasta finalizar las mismas.

a) Siguiendo la cadena estática b) Siguiendo la cadena dinámica

```
Program Main
  Var a: array[1..10] of integer;
      x,y,z:integer
  Procedure A ()
    var y,t: integer;
    begin
      a(1):= a(1)+1;z:=z+1;
      t:=1; y:=2;
      B(); a(y):=a(y)+3; y:=y+1;
      If z=11 Then Begin
        a(z-1):=a(z-2) + 3; z:=z-4;
        a(z-y):=a(z) - a(y) + 5;
      End;
    end;
  Function t():integer
  begin
    y:=y+1; z:=z-6;
    return(y+x);
```

```

    end;
Procedure B()
  var d:integer;
  Procedure I ()
  begin
    x:=0; x:=x+6;
  end;
  begin
    x:=x+t; d:=0;
    while x>d do begin
      I(); x:=x-1;
      d:=d + 2;
    end;
  end;
end;
begin
  For x:=1 To 10 do a(x):=x;
  x:=5; y:=1; z:=10;
  A();
  For x:=1 To 10 do write(a(x),x);
end.

```

Nota: La forma de evaluación de este lenguaje es de izquierda a derecha

Siguiendo la cadena estática

	*** Reg Activ Main
*1	Pto retorno
	A(1)= 4
	A(2)= 2
	A(3)= 3
	A(4)= 4
	A(5)= 5
	A(6)= 6
	A(7)= 7
	A(8)= 8
	A(9)= 9
	A(10)= 10
	X= 4..10-5
	Y= 4 - 2
	Z=10 - 44 - 5
	Procedure A
	Function T
	Procedure B
	VR . . . . .
*2	***Reg Activ A
	Pto Retorno
	EE (*1)
	ED (*1)
	Y = 2
	T = 1
	VR . . . . .
	*** Reg Activ B
	Pto Retorno
	EE
	ED
	D =
	Procedure I
	VR . . ¿? . .
	*** Reg Activ...(a partir de aca lo debe continuar...
	.....

Siguiendo la cadena dinámica

	*** Reg Activ Main
*1	Pto retorno
	A(1)= 1, 2, 5
	A(2)= 2
	A(3)= 3
	A(4)= 4
	A(5)= 5
	A(6)= 6
	A(7)= 7
	A(8)= 8
	A(9)= 9
	A(10)= 10
	X= 1..10-5
	Y= 4 - 2
	Z=10 - 11
	Procedure A
	Function T
	Procedure B
	VR . . . . .
*2	***Reg Activ A
	Pto Retorno
	EE (*1)
	ED (*1)
	Y = 2
	T = 1
	VR . . . . .
*3	*** Reg Activ B
	Pto Retorno
	EE (*1)
	ED (*2)
	D =
	Procedure I
	VR . . ¿? . .
*4	*** Reg Activ...(a partir de aca lo debe continuar...)
	.....

**Ejercicio 3:** Sea el siguiente programa escrito en Pascal-like. Realice la pila de ejecución

- Siguiendo la cadena estática
- Siguiendo la cadena dinámica

```

Procedure Main;
  var x, y: integer;
  vec: array[1..7] of integer;
  Function B:integer;
    var y:integer;
    begin
      y:=4; x:= y - 2;
      return (x);
    end;
  Procedure D;
    var i, x: integer;
    vec: array[1..7] of integer;
    Procedure A;
      var y:integer;
      begin
        y:=x + 5; vec(i + 2):= vec(i + 2) + y;
        x:= x +B; C;
      end;
    Function B:integer;

```

```

        begin
            vec(i):= y + 2; i:=i+2;
            vec(i):= vec(1) * i;
            return ( vec(i)-vec(1) );
        end;
    begin
        for x:= 1 to 7 do vec(x):= 1;
        x:=1; i:= 2;
        if y = 7 then A; else C;
        for x:= 1 to 7 do write(vec(x));
        end;
    Procedure C;
        var i, y: integer;
        begin
            i:= 1; y:= 6; x:= x + B;
            vec(2):= vec(2) * x;
            while (i < y) do begin
                vec(i):= vec(i) + B - 1;
                i:= i + 3;
            end;
            y:= y - 4;
        end;
    begin
        for x:= 1 to 7 do vec(x):= x;
        x:= 3; y:= B+5; D;
        if (x = 2) then begin
            vec(x):= vec(x) + 2;
            vec(x + 3):= vec(x) * 3;
        end;
        for x:= 1 to 7 do write(vec(x));
    end.

```

Nota: La forma de evaluación de este lenguaje es de izquierda a derecha

**Ejercicio 4:** Sea el siguiente programa escrito en Pascal-like. Realice la pila de ejecución

- Siguiendo la cadena estática
- Siguiendo la cadena dinámica
- La sentencia  $x := c + 5 + x$ , podría reemplazarse por  $x := x + c + 5$ ? Justifique la respuesta

```

Program Main;
  Var x, y, z:integer;
  a, b: array[1..6] of integer;
  Procedure B;
    var y,x: integer;
    Procedure C;
      var c:integer;
      begin
        y:= y + 2; c:=2;
        a(x):=a(x)*y;
        if (y >7) then
          b(y-6)=b(4)*2+b(y-6);
      D;
    end;
  begin
    x:=2; y:= x + 3;
    C; x:= x + 1; write (x,y);
  End;
  Procedure D;
  begin
    x:= c + 5 + x;
    y:= y + 2;
  end;

```

```

Function C: integer;
begin
    b(x):= b(x) + 1;
    x:= x + 1;
    a(y):=a(y)+b(x)+3;
    a(x+2)=a(x) + 2;
    return b(x);
end
begin
    x:= 1; Y:= 2;
    for z:=1 to 6 do begin
        a(z):= z;
        b(z):= z + 2;
    end;
    B;
    for z:= to 6 do write (a(z), b(z));
end.

```

Nota:La forma de evaluación de este lenguaje es de izquierda a derecha

**Ejercicio 5:** Sea el siguiente programa escrito en Pascal-like. Realice la pila de ejecución

- a) Siguiendo la cadena estática
- b) Siguiendo la cadena dinámica

```

Procedure Main;
var i, x, y: integer; b: array[1..5] of integer;
Procedure A
var x: integer;
Function B :integer;
Begin
    b[y]:= b[y] * i; return (i + x);
end;
begin
    x:=3; b[4]:= b[4] + B; D;
end;
Procedure C;
var i: integer;
Procedure D;
var i:integer;
b: array[1..5] of integer;
begin
    for i:=1 to 5 do b[ i ]:= i * 2;
    i:=5;
    if (x=7) then Begin x:=x+1; A; end;
    for i:=1 to 5 do write b[ i ];
end;
Function B:integer;
Begin
    b[i]:= i * b[5]; return(b(i));
end;
begin
    i:= x - 5; b[i]:= i * y;
    b[y + 1]:= b[i] +7 + B; D;
end;
Function B:integer;
begin
    b[3]:= 4 * b[3]; x:=x+3; return (b[3]);
end;
Procedure D;
begin

```

```

        b[2]:= b[2] + B;
    end;
begin
    for i:=1 to 5 do b[ i ]:= i; i:=2; y:=3; x:=7;
    C; write (x, y, i);
    for i:=1 to 5 do write b[ i ];
end.

```

**Ejercicio 6:** Sea el siguiente programa escrito en Pascal-like. Realice la pila de ejecución

- a) Siguiendo la cadena estática
- b) Siguiendo la cadena dinámica

<pre> Program principal Var i:integer; Var estado:boolean; var z: integer; Procedure proc1;   Var a: integer;   Var vec: array[1..4] of integer;   Var z: integer; Begin   a:=1;   z:=1;   for i:=1 to 4 do begin     z:= f+3;     vec[i]:=z;   end;   if (estado = true) then begin     a:=z;   end;   write(a, z);   For i=1 to 4 begin write(vec(i)); end; End; </pre>	<pre> Procedure proc2;   var f: integer; begin   estado := (i mod 2) =0;   f:=1+z;   if (z==0) proc1;   write(f);   z:=1; end;  Función f:integer; Begin   z:= i;   Return i*3; End;  Begin   z:=0;   i:=3;   estado:=true;   proc2();   Write(i,estado,z); END. </pre>
---	---