

Práctica 6

viernes, 24 de mayo de 2024 19:14

Ejercicio 4: Sea el siguiente programa escrito en Pascal-like

```

Procedure Main;
var j, m, i: integer;

Procedure Recibe (x:integer; y:integer);
begin
    m:= m + 1 + y;
    x:=i + x + j;
    y:=m - 1;
    write (x, y, i, j, m);
end;

Procedure Dos;
var m:integer;
begin
    m:= 5;
    Recibe(i, j);
    write (i, j, m);
end;
begin
    m:= 2;
    i:=1; j:=3;
    Dos;
    write (i, j, m);
end.
    
```

Main
Recibe Dos

a- Arme el árbol de anidamiento sintáctico y el registro de activación de cada una de las unidades.

b- Decir qué imprime el programa suponiendo que para todas las variables que se pasan el pasaje de parámetros es por. (Deberá hacer la pila estática y dinámica para cada caso)
I- Referencia II- Valor III-Valor Resultado IV- Nombre V-Resultado.

Referencia

Cadena estática

*1 Reg. Activación Main

Punto de retorno

Ld:-

Le:-

j:3,5

m:2,6

i:1,5

Recibe:

Dos:-

Valor de retorno:

Write 5,5,6

*2 Reg. Activación Dos

Punto de retorno

Ld:1

Le:1

m:5

Valor de retorno:

Write(5,5,5)

*3 Reg. Activación Recibe

Punto de retorno

Ld:2

Le:1

x:-

y:-

Valor de retorno:

m=m+1+y

m=2+1+3=6

x=i+x+j

x=1+1+3=5

y=m-1

y=6-1=5

Write(5,5,5,6)

Referencia

Cadena dinámica

*1 Reg. Activación Main

Punto de retorno

Ld:-

Le:-

j:3,8

m:2

i:1,5

Recibe:

Dos:

Valor de retorno:

Write 5,8,2

*2 Reg. Activación Dos

Punto de retorno

Ld:1

Le:1

m:5,9

Valor de retorno:

Write(5,8,9)

*3 Reg. Activación Recibe

Punto de retorno

Ld:2

Le:1

x:-

y:-

m=m+1+y

m=5+1+3=9

x=i+x+j

x=1+1+3=5

y=m-1

y=9-1=8

Write(5,8,5,8,9)

Valor de retorno:

Valor
Cadena estática

*1 Reg. Activación Main

Punto de retorno

Ld:-

Le:-

j:3,

Write(1,3,6)

m:2,6

i:1,

Recibe:

Dos:

Valor de retorno:

*2 Reg. Activación Dos

Punto de retorno

Ld:1

Write(1,3,5)

Le:1

m:5

Valor de retorno:

*3 Reg. Activación Recibe

Punto de retorno

Ld:2

m=m+1
m=2+1+3=6

x=x+1
x=1+1+3=5

y=m-1
y=6-1=5

Write(5,5,1,3,6)

Le:1

x:1,5

y:3,5

Valor de retorno:

Valor resultado
Cadena dinámica

*1 Reg. Activación Main

Punto de retorno

Ld:-

Le:-

j:3,8

Write(5,8,2)

m:2,

i:1,5

Recibe:

Dos:

Valor de retorno:

*2 Reg. Activación Dos

Punto de retorno

Ld:1

Write(5,8,9)

Le:1

m:5,9

Valor de retorno:

*3 Reg. Activación Recibe

Punto de retorno

Ld:2

m=m+1
m=5+1+3=9

x=x+1
x=1+1+3=5

y=m-1
y=9-1=8

Write(5,8,1,3,9)

Le:1

x:1,5

y:3,8

Valor de retorno:

Vombre(iden anterior)

Resultado
Explota pq se usan x e y sin antes inicializarlas

Ejercicio 4: Sea el siguiente programa escrito en Pascal-like

```
Procedure Main;  
var j, m, i: integer;  
  
Procedure Recibe (x:integer; y:integer);  
begin  
    m:= m + 1 + y;  
    x:=i + x + j;  
    y:=m - 1;  
    write (x, y, i, j, m);  
end;
```

```
Procedure Dos;  
var m:integer;  
begin  
    m:= 5;  
    Recibe(i, j);  
    write (i, j, m);  
end;  
  
begin  
    m:= 2;  
    i:=1; j:=3;  
    Dos;  
    write (i, j, m);  
end.
```

Ejercicio 5: Suponiendo que se está ejecutando un programa con el siguiente registro de activación en memoria y se llama al procedimiento rutina(iter,vec,a). Determine el tipo de parámetro que se deben utilizar en el llamado para que los resultados sean los siguientes:

- a) (4,6,7),(4,6,7), 2, 2 — DEF.
b) (3,5,6),(4,6,7), 2, 2 — VALOR
c) (3,5,6),(5,5,6), 0, -1 — VALOR TODAS

PR
LD
LE
Iter: true
Vec:[3,5,6]
a: -1
Rutina()
VR

.....
procedura rutina(tipoParam iteracion,tipoParam vector,tipoParam vit):

```
while iteracion begin  
    vit = a+1  
    vector[vit] = vector[vit]+1  
    iteracion = (vector[vit] mod 2)=0  
end  
print vec
```

Iter: true
Vec:[3,5,6]
a: -1
Rutina()
VR

```

.....
procedura rutina(tipoParam iteracion,tipoParam vector,tipoParam vit):
    while iteracion begin
        vit = a+1
        vector[vit] = vector[vit]+1
        iteracion = (vector[vit] mod 2)==0
    end
    print vec
    print vector
    print vit
    print a
.....

```

X referencia:
 Vit=0
 v[0]=3+1=4
 Iteracion=true
 Vit=1
 v[0]=6
 It=true
 Vit=2
 v[0]=7
 Iteracion=false

Ejercicio 7: Realice la pila de ejecución del siguiente programa: **a)** siguiendo la cadena estática **b)** siguiendo la cadena dinámica

```

Procedure Uno;
y, z: integer;
r1:array[1..6] of integer;
r2:array[1..5] of integer;
  Procedure Dos( nombre x, t:integer;
var
  io:integer; valor-resultado
y:integer);
    Procedure Dos( nombre
t1:integer);
      Procedure Tres;
      begin
        y:= y + 1;
        z:= z + 1;
      end;
      begin
        t1:= t1 + 1;
        t:= t + 1;
        Tres;
        t1:= t1 + 2;
        t:= t + 2;
      end;
    end;
  end;
begin
  x:= x + 1;
  t:= t + 1;
  io:= io + 1;
  x:= x + 2;
  if z =2 then Dos ( t );
end;
begin
  for y:= 1 to 6 do r1(y):= 2;
  for y:= 1 to 5 do r2(y):= 1;
  z:= 2;
  y:= 1;
  Dos( r1( y + r2( y )), r2( z ), y, z);
  for y:= 1 to 6 do write (r1(y));
  for y:= 1 to 5 do write (r2(y));
end.

```

*1 Reg. Activación uno
 Punto de retorno
 Ld:-
 Le:-
 y:1,2,3,4,5,6,1,2,3,4,5,1,2,
 z:2,3,3
 R1[1]:2
 R1[2]:2,3
 R1[3]:2
 R1[4]:2,4
 R1[5]:2
 R1[6]:2
 R2[1]:1
 R2[2]:1,2,3,4
 R2[3]:1,3,5
 R2[4]:1
 R2[5]:1
 Dos:
 Valor de retorno:

*2 Reg. Activación DOS
 Punto de retorno
 Ld:1
 Le:1
 x:=r1(y+r2(y))
 t:=r2(z)
 Io:
 y:2,3
 Procedure dos
 Valor de retorno:

*3 Reg. Activación DOS
 Punto de retorno
 Ld:2
 Le:2
 T1:tt
 Procedure Tres:
 Valor de retorno:

*4 Reg. Activación tres
 Punto de retorno

FOR i=1 to 6 FOR i=1 to 5
 $r1(y)=2$ $r2(z)=1$

Nombre Nombre 50 referencia
 Dos (r1(y + r2(y)), r2(z), y, z)

$x = x + 1$ $t = t + 1$ $y = y + 2$
 $x = r1(z)$ $t = r2(z)$ $x = r1(y)$
 $x = 2 + 1 = 3$ $t = 1 + 1 = 2$ $x = 2 + 2 = 4$
 $io = io + 1$ $io = 1 + 1 = 2$ $if z=2 then Dos(t)$

$t1 = t1 + 1$
 $t1 = t \rightarrow r2(z) = 2$

$t1 = 2 + 1 = 3$
 $t = t + 1$
 $t = 4$

$t1 = t1 + 2 \rightarrow r2(z) = r2(3) = 1$
 $t1 = t1 + z = 3$
 $t = t + 2 \rightarrow r2(z) = 1$

*4 Reg. Activación tres
Punto de retorno
Ld:3
Le:3
Valor de retorno:

$$r = r + 1$$

$$r = 1$$

$$r = r + 1$$

$$r = 3$$

$$r = 3$$

$$r = r + 1$$

$$r = 1$$

$$r = r + 1$$

$$r = 2$$

$$r = 3$$

b) Realice la pila estática y dinámica tanto con el pasaje de parámetros deep y shallow para el siguiente código.

```
Program A
  Var x:integer;
  Var y: char;
  Procedure B;
    Var h:integer;
    Begin
      h:=1+x;
      Write (y);
      C(D);
      Write (y);
    End;
  Procedure C (Subrutina S);
    Var x:integer;
    Var y: char;
    Begin
      x:=3;
      y:= "b";
      x:=S(x,y);
      y:= "j";
      Write (x,y);
    End;
```

```
Function D (j:integer, k:char);
  Begin
    j:=j+k;
    k:=y;
    Write (k);
    Return j;
  End;
BEGIN
  x:=0;
  y:= "a";
  B();
  Write (x,y);
END.
```

Pasaje de parámetros deep

*1 Reg. Activación A
Punto de retorno
Ld:-
Le:-
x:0
y:"a"
Procedure b
Procedure c
Procedure d
Valor de retorno:

$$C(D)$$

$$C(D)$$

*2 Reg. Activación B
Punto de retorno
Ld:1
Le:1
h:1
Valor de retorno:

$$h = 1 + x$$

$$h = 1$$

$$h = 1$$

$$C(D)$$

$$C(D)$$

*3 Reg. Activación C
Punto de retorno
Ld:2
Le:1
S:subrut D
x:3,3
y:b,j
Valor de retorno:3

$$x = S(x,y)$$

$$x = 3$$

$$y = b,j$$

$$C(D)$$

$$C(D)$$

*4 Reg. Activación D
Punto de retorno
Ld:3
Le:1
j:3,3
k:b,a
Valor de retorno:

$$j = j + x$$

$$j = 3 + 0 = 3$$

$$k = b$$

$$C(D)$$

$$C(D)$$

SHALLOW

*1 Reg. Activación A
Punto de retorno
Ld:-
Le:-
x:0
y:"a"
Procedure b
Procedure c
Procedure d

$$C(D)$$

$$C(D)$$

Valor de retorno:

*2 Reg. Activación B

Punto de retorno
Ld:1
Le:1
h:1
Valor de retorno:

$k = 1 + x$
 $k = 1 + 0 = 1$
 $write(x, y) \rightarrow 1, 1$
 $q(0)$
 $write(x, y) = 1, 1$

*3 Reg. Activación C

Punto de retorno
Ld:2
Le:1
S:subrut D
x:3,6
y:0,j
Valor de retorno:6

$z = s(x, y)$

$x = 6$
 $write(x, y) \rightarrow 6, j$

*4 Reg. Activación D

Punto de retorno
Ld:3
Le:1
j:3,6
k:b,b
Valor de retorno:

$j = j + x$

$j = 3 + 3 = 6$

$k = 0$
 $write(x, y) = 6$

```

Program A
Var x:integer;
Var y:char;
Procedure B;
  Var h:integer;
  Begin
    h:=1+x;
    Write (y);
    C(D);
    Write (y);
  End;
Procedure C (Subrutina S);
  Var x:integer;
  Var y:char;
  Begin
    x:=3;
    y:="b";
    x:=S(x,y)
    y:="j";
    Write (x,y);
  End;

```

```

Function D (j:integer, k:char);
  Begin
    j:=j+x;
    k:=y;
    Write (k);
    Return j;
  End;
BEGIN
  x:=0;
  y:="a";
  B();
  Write (x,y);
END.

```

nombre

*1 Reg. Activación main

Punto de retorno
Ld:-
Le:-
a(1):1,3
a(2):1,2
a(3):1,0
a(4):1
a(5):1
x:3,0,1,2
i:1,2,3,4,5
Procedure uno:
Valor de retorno:

$write(3, 2, 0, 1, 1)$

for i:=1 to 5
 $a(i) = 1$
 $uno(a(i))$

*2 Reg. Activación uno

Punto de retorno
Ld:1
Le:1
m:=a(x)
Procedure uno:
Valor de retorno:

$x = 0$
 $x := x + 1 = 1$
 $m = m + x + a(3)$
 $a(1) = a(1) + 1 + 1$
 1
 $x = x * 2$
 $x = 1 * 2 = 2$

$a(3) = a(3) - 1$
 $a(3) = 1 - 1$

$m = m + 1$
 $a(2) = 1 + 1 = 2$

REFERENCIA

*1 Reg. Activación main

Punto de retorno
Ld:-
Le:-
a(1):1

```

a(2):1
a(3):1,3,2,3
a(4):1
a(5):1
x:3,0,1,2
i:1,2,3,4,5
Procedure uno:
Valor de retorno:

```

```

*2 Reg. Activación uno
Punto de retorno
Ld:1
Le:1
m:
Procedure uno:
Valor de retorno:

```

valor = 3

```

*1 Reg. Activación main
Punto de retorno
Ld:-
Le:-
a(1):1
a(2):1
a(3):1,0,4
a(4):1
a(5):1
x:3,0,1,2
i:1,2,3,4,5
Procedure uno:
Valor de retorno:

```

```

*2 Reg. Activación uno
Punto de retorno
Ld:1
Le:1
m:1,3,4
Procedure uno:
Valor de retorno:

```

1, 1, 3, 1

x = 0

x = x + 1 = 1

$m = m + x + a(3)$
 $m = 1 + 1 + 1 = 3$
 $x = x * 2 = 2$

$a(3) = 3 - 1 = 2$
 $m = m + 1 = 3$

una(3)

```

Procedure main
a: array(1..5) of integer;
x: integer;
i: integer;
Procedure Uno (tipo_pasaje
m: integer)
Begin
x:=0;
x:=x+1;
m:=m+x + a(3);
x:=x*2;
a(3):=a(3) - 1;
m:=m+1;
End;

```

```

Begin
For i:=1 to 5 a(i):=1;
x:=3;
Uno(a(x));
For i:=1 to 5 write (a(i));
End.

```

Plantee diferencias, relacionada con la forma de implementación de cada uno y los **resultados** sobre este ejemplo considerando los siguientes tipos de pasajes parámetros: nombre, referencia y valor **resultado**.

m = m + x + 1

m = 3

m = 4

x = 2

a(3) = a(3) - 1