

**Práctica Nro. 6
Parámetros**

Objetivo: Descubrir las técnicas existentes para pasaje de parámetros entre unidades y sus diferencias esenciales de acuerdo al lenguaje que lo implementa

Ejercicio 1:

a- Explique brevemente los siguientes conceptos

- Parámetro
- Parámetro real
- Parámetro formal
- Ligadura posicional
- Ligadura por palabra clave o nombre

Ejercicio 2: Unir los siguientes puntos según corresponda y de una definición y un ejemplo de cada par.

| | |
|---------------|------------------------|
| | Resultado |
| | Valor |
| Modo IN | Valor / Resultado |
| Modo OUT | Nombre |
| modo IN / OUT | Resultado de funciones |
| | Valor Constante |
| | Referencia |

Ejercicio 3:

a- Complete el siguiente cuadro según lo correspondiente a cada lenguaje:

| Tipo de pasaje de parámetros | Lenguaje |
|------------------------------|----------|
| | ADA |
| | C |
| | Rubi |
| | JAVA |
| | Python |

b- Ada es más seguro que Pascal, respecto al pasaje de parámetros en las funciones. Explique por qué.

c- Explique cómo maneja Ada los tipos de parámetros in-out de acuerdo al tipo de dato

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

| | |
|---|---|
| <pre> 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; </pre> | <pre> 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. </pre> |
|---|---|

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.

c- ¿Existió algún caso que no pudo realizarlo porque saltó algún tipo de error? Diga cuál y por qué.

d- ¿Dará el mismo resultado si se trata de un lenguaje que sigue la cadena dinámica? Justifique la respuesta realizando las pilas de activación

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
- b) (3,5,6),(4,6,7), 2, 2
- c) (3,5,6),(5,5,6), 0, -1

| |
|-------------|
| 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
print vector
print vit
print a

```

.....

rutina(iter,vec,a)

Ejercicio 6: Indique con un ejemplo el comportamiento del parámetro por nombre (en el parámetro formal) para los siguientes casos de parámetros reales:

- Un valor entero.
- Una constante.
- Un elemento de un arreglo.
- una expresión.

Que sucede en cada caso?

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

| | |
|---|--|
| <pre> 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; </pre> | <pre> 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. </pre> |
|---|--|

Ejercicio 8:

- a)** Indique las diferencias entre los pasaje de subprogramas como parámetros deep y shallow.

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

| | |
|--|---|
| <pre>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;</pre> | <pre>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.</pre> |
|--|---|

Ejercicio 9: Sea el siguiente código escrito en Pascal like

| | |
|---|---|
| <pre>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;</pre> | <pre>Begin For i:=1 to 5 a(i):=1; x:=3; Uno(a(x)); For i:=1 to 5 write (a(i)); End.</pre> |
|---|---|

a- 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.

b- ¿Qué sucede si en Uno se agrega la siguiente declaración: x: integer? Indique el resultado para cada uno de los tipos de pasajes de parámetros (nombre, referencia y valor resultado)

Ejercicio 10: Sea el siguiente un programa escrito en Pascal:

| | |
|--|--|
| <pre>Program Uno; var x:integer; Function Dos:integer; begin x:= x + 1; return (x); end;</pre> | <pre>Procedure Tres (pasaje x:integer); begin x:= x + 5; x:= Dos + 10; end; begin x:= 8; Tres(x); write (x); end.</pre> |
|--|--|

a- Explique cómo simularía en Pascal el pasaje por valor-resultado y hágalo sobre este ejemplo.

Nota: No se pueden agregar más variables, ni cambiar el nombre de las que están.

b- Transcriba este ejemplo en Ada de manera tal que el resultado de la ejecución sea diferente si el pasaje de parámetros es por referencia y luego por valor – resultado