



Ejercicios para resolver

Ejercicios Práctica Taller 1 y Examen 1

Estos 2 Algoritmos ¿Intercambiar el Mayor con el Menor ? Es cierto Verdadero o Falso hacer la Trazas

Opción 1

```

A ← 1; C ← 3;
Escribir ( A ) ; Escribir ( B )
Si ( A ≤ C ) entonces
    Aux ← A;
    A ← C; C ← Aux;
FSi

Si ( A > C ) entonces
    Aux ← C;
    C ← A; A ← Aux;
FSi
Escribir (A);
Escribir(C);
    
```

Variable	Valor
A	
C	
Aux	

Inicio

Escribir A=
Escribir C=

Fin

Escribir A=
Escribir C=

Opción 2

```

A ← 1; C ← 3;
Escribir ( A ) ; Escribir ( C )
Si ( A ≤ C ) entonces
    Aux ← A;
    A ← C; C ← Aux;
SINO
    Aux ← C;
    C ← A; A ← Aux;
FSi

Escribir (A);
Escribir(C);
    
```

Variable	Valor
A	
C	
Aux	

Inicio

Escribir A=
Escribir C=

Fin

Escribir A=
Escribir C=

Realizar la Taza del siguiente Algoritmo

Algoritmo

x, y, z, w: entero;

Procedimiento **tibio**(**w**:entero; **y**:entero; **z**:entero);

w:= **y**+**z**;

x:= **w** + **y**;

z:= **x** + 2;

Escribir(**x**, **y**, **z**, **w**);

FProcedimiento;

Funcion **caliente**(**w**:entero; **z**:entero; **x**:entero) : **entero**;

Var

a:entero;

x:=**x**+1;

y:=**z**+1;

a:=**x**+2;

w:=**x**+**y**;

x:=**z**+**w**;

Escribir(**x**, **y**, **z**, **w**);

caliente:= **w** + **x**;

FFuncion

Inicio

x:=1; **y**:=2; **z**:=3; **w**:=0;

tibio(**caliente**(**x**,**w**,**z**),**w**,**y**);

Escribir(**x**, **y**, **z**, **w**);

Fin

Taza

Variable	Valor
x	
y	
z	
w	
w	
z	
x	
a	
w	
y	
z	

- Valor Inicial
- Variables Globales
- Variables Locales
- Variables Locales

Escribir =

caliente =

Escribir =

Escribir =

Determinar la Complejidad Temporal y Espacial del siguiente Algoritmo

Estos son EJEMPLOS no los valores reales



Complejidad Temporal

Nº Línea	Complejidad Big O
22	$O(1)$
25	$O(N)$
26	$O(1)$
Total	$2 O(1) + O(N)$
Simplificada	$O(1) + O(N)$
Final	$O(N)$

Complejidad Espacial

Cantidad	Tipo	Total
2	entero	8 bytes
1	caract	1 byte

Estructuras

Nom	Tam
a	50 B

```

1  Algoritmo
2  tipo
3  |      h = Arreglo [ 0..99] de entero;
4  var
5  |      n, pp, sp, tp, cp, cd, n1, may, par : entero (Variables GLOBALES )
6  |      a:h;
7  Función Validar ( x:entero) : booleano;
8  |      Validar ←( ( n > 9.999.999) y (n < 99.999.999) ) ;
9  ffuncion
10 |      Procedimiento Separar (y:entero, z:entero);
11 |      |      par ← ( y div z);
12 |      fprocedimiento
13 |      Función Mayor (w: entero, q: entero) : entero ;
14 |      |      Si (w > q) entonces
15 |      |      |      Mayor ← w ;
16 |      |      |      sino
17 |      |      |      Mayor ← q ;
18 |      Fsi
19 |      ffuncion
20 Inicio
21 |      Para i hasta 99 hacer
22 |      |      n1 ← a [ i ]
23 |      |      Si ( Validar ( a [ i ] ) = Verdad ) entonces
24 |      |      |      cd:←1.000.000; Separar (n1,cd); pp ← par; n1← (n1 mod cd) ;
25 |      |      |      cd:←10.000; Separar (n1, cd), sp ← par; n1← (n1 mod cd) ;
26 |      |      |      cd ← 100; Separar (n1, cd), tp ← par; cp ← (n1 mod cd) ;
27 |      |      may ← Mayor (Mayor (pp,sp), Mayor (tp,cp)) ;
28 |      |      Escribir( ' El mayor par de dígitos del numero ',a { i } , ' es ' , may);
29 |      |      sino
30 |      |      |      Escribir ( ' ERROR el Número ',a [ i ] , ' NO tiene 8 dígitos ');
31 |      Fsi
32 |      fpara
33 Fin
  
```

Problema Arreglos

Dado un Arreglo de 3 posiciones llamado N el cual contiene números enteros (solo números de un dígito del 1 al 9) suministrados por el usuario, y dado otro Arreglo llamado N1 de 10 posiciones con los números del 0 al 9, de uno en uno, en orden ascendente generados automáticamente. Se desea Elaborar un Algoritmo y un Programa en Python que:

- 1.- Solicite los números al usuario y los almacene en el Arreglo N y al finalizar lo muestre.
- 2.- Almacene de forma Automática los números del 0 al 9 en el Arreglo N1 y al finalizar lo muestre.
- 3.- Almacene en otro Arreglo llamado N2 de 30 posiciones los números de 2 dígitos conformados por cada número del arreglo N con cada uno de los números del Arreglo N1 y al finalizar lo muestre.
- 4.- Almacene en otro Arreglo llamado N3 de 30 posiciones el RESULTADO de RESTAR el primer número de N2 menos el mayor número de N1 y el resultado de esta operación se almacenara en la primera posición del Arreglo N3, y así sucesivamente hasta usar todos los números de N1, para luego repetir el proceso con el resto de los números de N2 y almacenando de forma consecutiva los resultados obtenidos en el Arreglo N3 y al finalizar mostrarlo.

Arreglos Ejemplo

1	2	3
---	---	---

N[I]

0	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

N1[J]

10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

N2[K]

1	3	5	7	9	11	13	15	17	19	11	13	15	17	19	21	23	25	27	29	21	23	25	27	29	31	33	35	38	39
---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

N3[L]

Elabora un programa RECURSIVO para determinar la raíz cuadrada aproximada de un número entero N utilizando el método de restas impares sucesivas. Este Método consiste en tomar el número al que se desea sacar la raíz cuadrada y restarle el primer número Impar (1), verificando luego si este resultado es un número negativo, en caso de NO serlo se le resta a este resultado el próximo número Impar (3) y nuevamente se verifica si el resultado obtenido es un numero positivo para seguir aplicando el proceso hasta que el resultado de la operación de resta sea un valor Negativo. La cantidad de veces que se pudo repetir el proceso antes de alcanzar esta condición (número negativo) corresponde al aproximado de la raíz cuadrada del número.

Ejemplos:

Para N=12

$$\left. \begin{array}{l} 12 - 1 = 11 \\ 11 - 3 = 8 \\ 8 - 5 = 3 \\ 3 - 7 = -4 \end{array} \right\} \textcircled{3}$$

Cantidad Operaciones = 3

Para N = 25

$$\left. \begin{array}{l} 25 - 1 = 24 \\ 24 - 3 = 21 \\ 21 - 5 = 16 \\ 16 - 7 = 9 \\ 9 - 9 = 0 \end{array} \right\} \textcircled{5}$$

$$0 - 11 = -11$$

Cantidad de Operaciones = 5

La cantidad de restas cuyo resultado sea mayor o igual a 0 indica el valor aproximado de la raíz cuadrada del número. Entonces la raíz cuadrada del número 12 es 3 y del número 25 es 5.

Elabora un Algoritmo y un Programa ITERATIVO y un Algoritmo y un Programa RECURSIVO para dado un número N entero positivo mayor que cero muestre esa cantidad de números de la secuencia de Fibonacci.

Ejemplos:

Para N=8

La Secuencia de los 8 primeros números de la secuencia de Fibonacci es:

0, 1, 1, 2, 3, 5, 8, 13.

