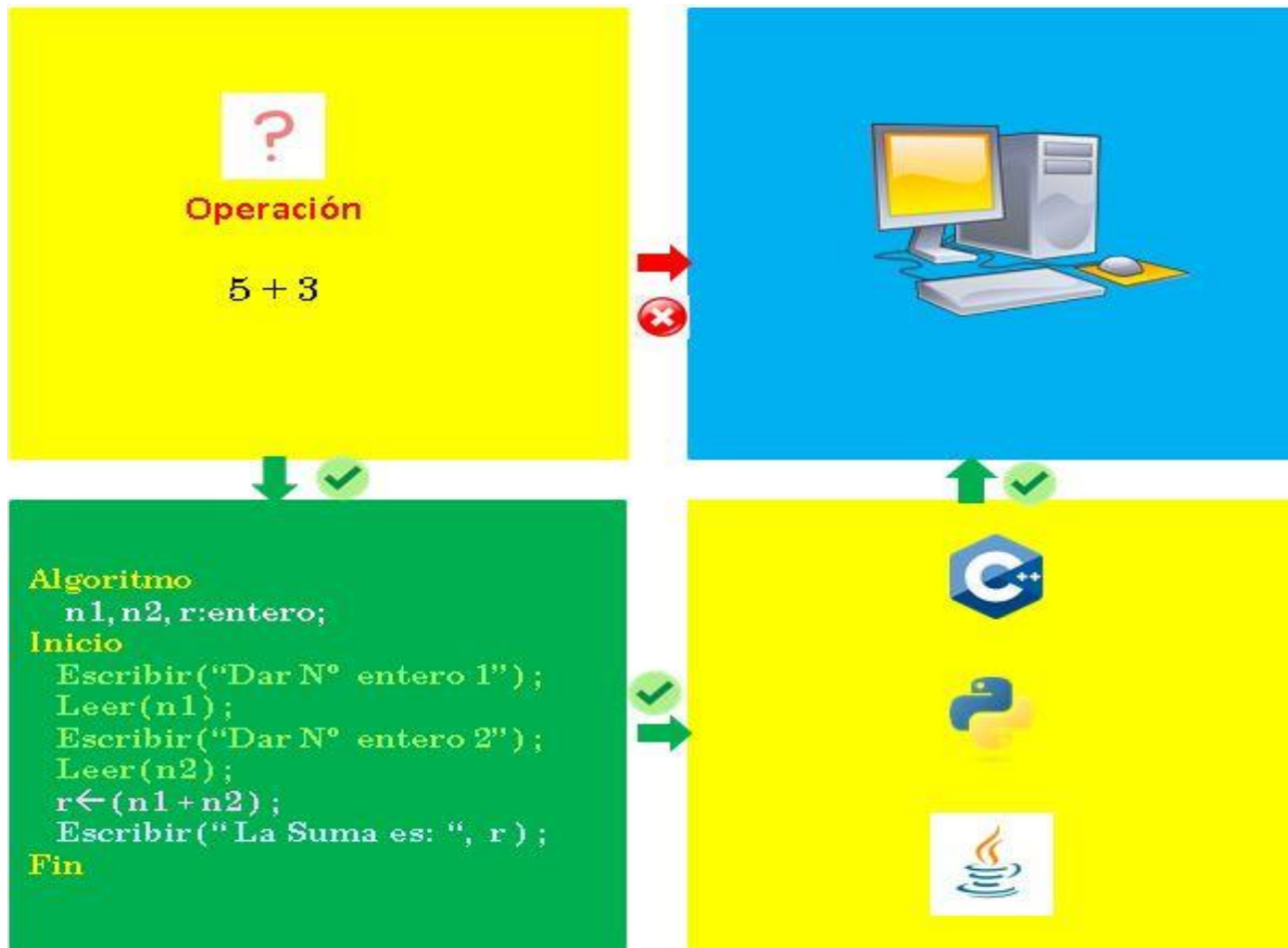


Conceptos de Algoritmos y Programación



Algoritmos y Programación

Prof. Rodolfo Holguín
holguinvaldez@gmail.com

Contenido Programático

Unidades Temáticas	
Unidad 1: Análisis de Complejidad de Algoritmos	<ul style="list-style-type: none"> 1.-Eficiencia en los algoritmos y su importancia. 2.- Tipos de complejidad: Espacio y tiempo. 3.- Notación Big O. 4.- Reglas de la notación O. 5.- Ordenes de complejidad. 6.- Complejidad de Algoritmos de ordenamiento y búsqueda.
Unidad 2: Recursividad	<ul style="list-style-type: none"> 1.- Definición de Recursividad. 2.- Tipos de recursividad: directa, indirecta, anidada. 3.- Algoritmo Iterativo versus Algoritmo Recursivo.
Unidad 3: Tipos de Datos Estructurados	<ul style="list-style-type: none"> 1.-Sub-rangos. 2.-Enumerados. 3.- Manejo avanzado de cadenas. 4.- Expresiones regulares.

Contenido Programático

Unidades Temáticas	
Unidad 4: Matrices	1.- Definición. 2.- Representación de matrices. 3.- Manejo de matrices. 4.- Operaciones con matrices.
Unidad 5: Archivos	1.- Archivos Binarios. 2.- Operaciones con archivos binarios. 3.- Acceso aleatorio.
Unidad 6: Algoritmos de búsqueda y ordenamiento	1.- Algoritmos de búsqueda: Secuencial. Binaria. 2.- Algoritmos de ordenamiento: Inserción directa (Baraja). selección directa. Intercambio directo (Burbuja). 3.- Algoritmos de ordenamiento recursivo: QuickSort, MergeSort, HeapSort

Plan de Evaluación

Tipo de Evaluación	Porcentaje	Día
Taller Práctico 1	10%	5-9/05/25
Examen Parcial 1	30%	12-16/05/25
Taller Práctico 2	10%	16-20/ 06 /25
Examen Parcial 2	30%	30-4/6 y 7/25
Intervenciones	5%	Todo el Semestre
Proyecto	15%	7-11/07/25

Nota: TODAS las Evaluaciones señaladas con anterioridad, están sujetas a **Interrogatorios Orales**, cuando el Profesor lo considere necesario y el resultado del mismo, puede modificar la Nota obtenida.

Conceptos de Algoritmos y Programación

Notación Algorítmica (pseudocódigo)

Algoritmo

- # Declaración de constantes
- # Declaración de variables
- # Declaración de Tipos

Procedimientos y Funciones

Inicio

Entrada de datos

Procesamiento de datos

Salida de datos

Fin

Lenguaje de Programación Python

Importar Bibliotecas

def main() :

Procedimientos y Funciones

Entrada de datos

Procesamiento de datos

Salida de datos

main ()

Conceptos de Algoritmos y Programación

Herramienta Algorítmica	Notación Algorítmica	Python
Operador de Asignación	\leftarrow $n1 \leftarrow 3$; $c1 \leftarrow$ “ Pedro Pérez “	= $n1 = 3$ $c1 =$ “ Pedro Pérez “
Operador de Escritura	Escribir (); Escribir (“ Su Nota es : “ , nota);	print () print (“ Su nota es : “ , nota) print (f “ Su nota es : { nota } “) COLORAMA import colorama from colorama import * init(autoreset=True) o init() print(Fore.BLUE+”Hola”)
Operador de Lectura	Leer () Leer (nota1); Leer (nom)	input () nom = input () nota1 = int (input ()) edad = int (input (“ Indique su edad “))
Parte entera de la división	DIV $n1 \leftarrow (7 \text{ DIV } 3)$; $n1 = 2$	// $n1 = (7 // 3)$ $n1 = 2$
Resto de la división	MOD $n2 \leftarrow (7 \text{ MOD } 3)$; $n2 = 1$	% $n2 = (7 \% 3)$ $n2 = 1$

Conceptos de Algoritmos y Programación

```
import colorama
from colorama import *
init(autoreset=True)      # o init()

def main():
    pi=3.14
    # Opcion 1
    print(Fore.BLUE+"Hola")
    # Opcion 2
    print(Fore.RED+f"El valor de PI es: { pi } ")
    # Opcion 3
    print(Fore.BLUE+"El valor de PI es: ",end = " ")
    print(Fore.RED+f"{pi}")
    print()
main()
```

Para Instalar COLORAMA

**Desde el Terminal de Visual Studio Code
escribir :**

> pip install colorama

Conceptos de Algoritmos y Programación

Herramienta Algorítmica	Notación Algorítmica	Python
Si Condicional Anidado	<p>Si (condición) entonces Acciones FSi</p> <p>Para n1 y n2 enteros distintos</p> <p>Si (n1 > n2) entonces may ← n1 ; men ← n2 ; FSi</p> <p>Si (n1 < n2) entonces may ← n2 ; men ← n1 ; FSi</p>	<p>if (condición) : Acciones</p> <p>Para n1 y n2 enteros distintos</p> <p>if (n1 > n2) : may = n1 men = n2</p> <p>if (n1 < n2) : may = n2 men = n1</p>
Si condicional con SINO	<p>Si (condición) entonces Acciones 1 SINO Acciones 2 FSi</p> <p>Si (nota >= 10) entonces Escribir (“ Aprobado “) ; SINO Escribir (“ Reprobado “) FSi</p>	<p>if (condición) : Acciones 1 else : Acciones 2</p> <p>if (nota == 10) : print(“ Aprobado “) else : print(“ Reprobado “)</p>

Conceptos de Algoritmos y Programación

Herramienta Algorítmica	Notación Algorítmica	Python
Si condicional con SINO	<p>El N° Mayor de 3 N° distintos</p> <p>Si ($n1 > n2$) y ($n1 > n3$) entonces Escribir(“ El Mayor es: “ , $n1$) SINO Si ($n2 > n1$) y ($n2 > n3$) entonces Escribir(“ El Mayor es: “ , $n2$); SINO Escribir(“ El medio es: “ , $n3$); FSi</p> <p>FSi</p>	<p>El N° del medio de 3 N° distintos</p> <pre>if ($n1 > n2$) and ($n1 > n3$) : print(“ El Mayor es: “ , $n1$) else : if ($n2 > n1$) and ($n2 > n3$) : print(“ El Mayor es: “ , $n2$) else : print(“ El medio es: “ , $n3$)</pre>
No Aplica	No Aplica	<p>Sola existente en Python (elif)</p> <pre>if ($n1 > n2$) and ($n1 > n3$) : print(“ El Mayor es: “ , $n1$) elif ($n2 > n1$) and ($n2 > n3$) : print(“ El Mayor es: “ , $n2$) else : print(“ El medio es: “ , $n3$)</pre>

Conceptos de Algoritmos y Programación

Problemas Sentencias Selectivas

- 1.- Elaborar un Algoritmo y un Programa en Python que simule un juego de PARES o NONES entre usted y el computador indicando quien es el Ganador de la partida.

- 2.- Elaborar un Algoritmo y un Programa en Python que simule un juego de PIEDRA, PAPEL y TIJERA entre usted y el computador indicando quien es el Ganador de la partida o si la partida quedo empate.

Conceptos de Algoritmos y Programación

Herramienta Algorítmica	Notación Algorítmica	Python
Ciclo Iterativo MIENTRAS	<p>Mostrar los N primeros N° IMPARES</p> <p>Escribir (“Indique valor entero positivo”); Leer (n); cont \leftarrow 0 ; ni \leftarrow 1; Mientras (cont \leq n) hacer Escribir (ni); cont \leftarrow (cont+1); ni \leftarrow (n1 + 2); FMientras</p>	<p>Mostrar los N primeros N° IMPARES</p> <pre>print (“Indique valor entero positivo”); n = int(input()) cont = 0 ni = 1 while (cont <= n) : print (ni) cont += 1 ni += 2</pre>
Ciclo Iterativo PARA	<p>Mostrar los N primeros N° IMPARES</p> <p>Escribir (“Indique valor entero positivo”); Leer (n); ni \leftarrow 1; Para i \leftarrow 1 hasta n hacer Escribir (ni); ni \leftarrow (n1 + 2); FPara</p>	<p>Mostrar los N primeros N° IMPARES</p> <pre>print (“Indique valor entero positivo”); n = int(input()) ni = 1 for i in range (0, n, 1) : print (ni); ni += 2</pre>

Conceptos de Algoritmos y Programación

Problemas Sentencias Iterativas (Mientras o While)

1.- Elaborar un Algoritmo y un Programa en Python que solicite un numero N entero positivo entre 8 y 15 y que a partir de este genere y muestre una secuencia de los N últimos números de 4 dígitos cada uno. Para cada número de la secuencia generada conformar a partir de este, dos pares de números P1 con el primer y último dígito y P2 con el segundo y tercer dígito, luego sumar los pares P1 y P2 y mostrar los pares y el resultado de la suma.

2.- Elaborar un Algoritmo y un Programa en Python que modifique los problemas resueltos de condicionales (juegos) implementando un menú para elegir alguno de los juegos o salir del programa, y además cada juego consistirá en un torneo a 3 partidas a lo sumo para el primero que gane 2, indicándolo como campeón del torneo y regresando nuevamente al menú y así hasta que el usuario desee salir del programa.

Conceptos de Algoritmos y Programación

Herramienta Algorítmica	Notación Algorítmica
Funciones ()	<p>Separar un N^a entero positivo de 4 dígitos en los 2 primeros y los 2 últimos.</p> <p>Algoritmo</p> <pre> var n, p1, p2 . entero ; función Separar (x: entero, y:enetro, z:entero) :entero ; var r:entero ; Si (z = 1) entonces r ← (x DIV y) ; Separar ← r ; SINO r ← (x mod y) ; Separar ← r , FSi Ffunción Inicio Escribir (“Indique N° entero positivo”); Escribir (“de 4 dígitos) ; Leer (n) ; p1 ← Separar(n, 100, 1) ; p2 ← Separar (n, 100, 2) ; Escribir (“ El Par 1 es: “ , p1); Escribir (“ El Par 2 es: “ , p2); Fin </pre>

Conceptos de Algoritmos y Programación

Herramienta Algorítmica	Python
Funciones ()	<pre> Separar un N^a entero positivo de 4 dígitos en los 2 primeros y los 2 últimos. def main () : # Declaración Variables Globales # Funciones y Procedimientos def Separar (x, y, z) : r:int if (z == 1) : r = (x // y) return r else : r = (x % y) return r # Programa Principal print ("Indique N° entero positivo") print ("de 4 dígitos) n = int(input ()) p1 = Separar (n, 100, 1) p2 = Separar (n, 100, 2) print (" El Par 1 es: " , p1) print (" El Par 2 es: " , p2) main () </pre>

Conceptos de Algoritmos y Programación

Herramienta Algorítmica	Notación Algorítmica
Procedimientos ()	<p>Separar un N^a entero positivo de 4 dígitos en los 2 primeros y los 2 últimos.</p> <p>Algoritmo</p> <p>var</p> <p> n, p1, p2 . entero ;</p> <p>procedimiento Separar (x: entero, y:enetro)</p> <p> p1 = (x // y)</p> <p> p2 = (x % y)</p> <p>Fprocedimiento</p> <p>Inicio</p> <p> Escribir (“Indique N^o entero positivo”);</p> <p> Escribir (“de 4 dígitos) ;</p> <p> Leer (n) ;</p> <p> Separar(n, 100) ;</p> <p> Escribir (“ El Par 1 es: “ , p1);</p> <p> Escribir (“ El Par 2 es: “ , p2);</p> <p>Fin</p>

Conceptos de Algoritmos y Programación

Herramienta Algorítmica	Python
Procedimientos ()	<pre> Separar un N^a entero positivo de 4 dígitos en los 2 primeros y los 2 últimos. def main () : # Declaración Variables Globales p1:int p2:int # Funciones y Procedimientos def Separar (x, y) : # Declaración Variables Globales nonlocal p1,p2 p1 = (x // y) p2 = (x % y) # Programa Principal print ("Indique N° entero positivo") print ("de 4 dígitos ") n = int(input ()) Separar(n, 100) ; print (" El Par 1 es: " , p1); print (" El Par 2 es: " , p2); main () </pre>

Conceptos de Algoritmos y Programación

Problemas

- 1.- Elaborar un Algoritmo y un Programa en Python que solicite un numero N entero positivo de 5 dígitos y determine si el número de 3 dígitos conformado por el segundo, tercero y cuarto dígito es una CAPICUA (se lee el mismo número de izquierda a derecha, que de derecha a izquierda), resolver el problema utilizando Funciones y Procedimientos.

- 2.- Elaborar un Algoritmo y un Programa en Python que modifique el problema resuelto en el ejercicio anterior N° 2 (Menú de los Juegos) pero utilizando Funciones y Procedimientos.

Conceptos de Algoritmos y Programación

Herramienta Algorítmica	Notación Algorítmica	Python
Arreglos	<p>Algoritmo</p> <p>Tipo</p> <p> x = Arreglo [0..4] de enteros;</p> <p> y = Arreglo [0..4] de conj de carac;</p> <p> z = Arreglo [0..4] de carcter;</p> <p>Var</p> <p> c:x;</p> <p> n:y;</p> <p> s:z;</p> <p> i, j, k, cal: entero</p> <p>Inicio</p> <p> Para i ← 0 hasta 4 hacer</p> <p> Escribir("Indique N° entero");</p> <p> Leer(cal);</p> <p> c [i] ← cal</p> <p> FPara</p> <p>Fin</p>	<pre>import numpy as np def main() : c = np . Array ([0] * 5) for i in range (0, len(c), 1) : print (" Indique N° entero ") cal = int (input ()) c [i] = cal main()</pre>