

USACA- EFPEM



CARLOS ORTIZ - 20091723
LIC. EN INFORMATICA

CARLOS ORTIZ SERRANO – USAC-EFPEM- LICENCIATURA EN INFORMATICA – DIDACTICA DE LA COMPUTACION – EXAMEN PARCILA No.1

1. Se comprende el problema:

Después de Heber leído el problema se lo logrado definir las estrategias de apoyo la cual podrá ser “Ensayo y error”, realizando un patrón e integrando una lista de elementos para el manejo de los datos.

1.1. Entradas

1.1.1. Se solicita el ingreso de cuatro elementos de tipo texto, dentro de una lista vacía.

```
parcial():  
    lista=[]  
    print ("Solicitar 4 Textos")  
    for inicio in range(4):  
        texto=input("Ingrese un texto:")  
        lista.append(texto)  
    print("\nLos elementos ingresados son:", lista)  
  
    print("---INDICAR CUAL DE LOS TEXTOS POSEE MENOR CARACTERES---")  
  
    texto_1=(lista[0])  
    texto_2=(lista[1])  
    texto_3=(lista[2])  
    texto_4=(lista[3])
```

1.2. Procesos

1.2.1. Se genera un encabezado, para la identificación del programa

1.2.2. Se encapsula dentro de una función los códigos para una mejor comprensión.

1.2.3. Se genera un código individual por cada enunciado de la lista.

```
print (" Carlos Ortiz Serrano")  
print (" Carnet 200917231")  
print (" Lic. en Informatica")  
print (" -----" )  
print ("P A R C I A L No. 1")  
  
def parcial():  
    lista=[]
```

1.3. Salidas

1.3.1. Serán las impresiones a pantalla durante la compilación del programa.

```
print ((li  
e:  
print ((li
```

2. Elabore un plan

Primero debemos generar variables, una lista, funciones, posteriormente seccionar cada cuestionamiento, y conllevar un patrón de orden.

3. Aplique un plan

Implementar las estrategias para la ejecución ir comprobando con cada ciclo y función específicamente.

```
def parcial():  
    lista=[]
```

4. Revise y verifique

Revisar todos los códigos leyendo cada línea y revisar si se puede resumir el código, se revisa si la solución es correcta y si se alcanza el fin adecuado de la incógnita.

4.1. Prueba 1:

4.1.1. Se realiza utilizando valor de tipo texto, donde el mayor será el segundo elemento y el cada texto será distinto del otro.

```
P A R C I A L No. 1  
Solicitar 4 Textos  
Ingrese un texto:España  
Ingrese un texto:Guatemala  
Ingrese un texto:Luis  
Ingrese un texto:Mar  
  
Los elementos ingresados son: ['España', 'Guatemala', 'Luis', 'Mar']  
--INDICAR CUAL DE LOS TEXTOS POSEE MENOR CARACTERES--  
El texto con menos caracteres es: Mar  
  
--TEXTO MAYOR Ó MENOR, EN RELACION DEL ELEMENTO 2 Y 3--  
Guatemala es el texto mayor  
  
--MOSTRAR LA SUMA DE LONGITUDES--  
No.de letras del textoo: España es : 6  
No.de letras del textoo: Guatemala es : 9  
No.de letras del textoo: Luis es : 4  
No.de letras del textoo: Mar es : 3  
Suma de todas las longitudes es: 22  
>>> |
```

4.2. Prueba 2:

- 4.2.1. Se utilizará valor numérico donde la suma de elementos será 12 y texto mayor será el último texto ingresado y donde textos estén iguales en longitud de caracteres e impida un ciclo de comparación.

```
P A R C I A L No. 1
Solicitar 4 Textos
Ingrese un texto:12
Ingrese un texto:345
Ingrese un texto:456
Ingrese un texto:9875

Los elementos ingresados son: ['12', '345', '456', '9875']
--INDICAR CUAL DE LOS TEXTOS POSEE MENOR CARACTERES--
Aplicacion no contenmpla IGUALDAD

--TEXTO MAYOR Ó MENOR, EN RELACION DEL ELEMENTO 2 Y 3--
456 es el texto mayor

--MOSTRAR LA SUMA DE LONGITUDES--
No.de letras del textoo: 12 es : 2
No.de letras del textoo: 345 es : 3
No.de letras del textoo: 456 es : 3
No.de letras del textoo: 9875 es : 4
Suma de todas las longitudes es: 12
>>> |
```