

GUÍA DE RETOS PARA REALIZAR FUERA DEL AULA DE CLASE

ASIGNATURA: Programación 1

RETO: RDA2 – CRITERIO 1 – TAREA 1

INTEGRANTES DEL GRUPO:

TEMA DEL RETO: Ejecutar programas en Python donde se maneja y asocia, cadenas de caracteres.

OBJETIVOS DEL RETO: Comprender cuál es el proceso de ejecución de programas evitando cometer errores y utilizando algunos de los componentes del tema del reto.

MARCO TEÓRICO:

- **Interpretar:** Es el proceso mediante el cual un programa se ejecuta línea por línea en tiempo real, sin necesidad de compilarlo previamente.
- **Intérprete:** Es el software capaz de leer y ejecutar el código fuente línea por línea.
- **Depurar:** Es el proceso por el cual se corrige un programa a fin de dejarlo libre de errores.

RECURSOS, MATERIAL Y EQUIPO: Computador con Python instalado, un editor de código como Notepad++ o Visual Studio Code, y el símbolo de sistema o terminal para desarrolladores.

ENUNCIADO, INSTRUCCIONES, ACTIVIDADES POR DESARROLLAR Y/O REGISTRO DE DATOS: Todos los programas que se realicen deben venir con su código fuente comentado.

- **EJERCICIOS:**

- a. Realizar un programa que ingrese una cadena de caracteres por teclado que representa una frase (palabras separadas por espacios), el programa debe contar cuantas palabras tiene la frase.

Ejemplo:

Introduce una cadena de caracteres... Me gusta la programación

Impresión:

La frase tiene **4** palabras

- b. Realizar un programa que se ingrese una cadena de caracteres por teclado que represente una frase (palabras separadas por espacios), el programa debe presentar la inicial de la letra en mayúscula de cada palabra.

Ejemplo:

Introduce una cadena de caracteres... me gusta la programación

Impresión:

Iniciales: **MGLP**

- **CUESTIONARIO**

Responda a las siguientes preguntas

1. **¿Cuál es la función principal de las cadenas de caracteres o strings en el contexto de la programación?**

“Una cadena de caracteres (string) puede ser representada por medio de un arreglo unidimensional de caracteres, y cada caracter de la cadena será almacenado en un elemento del arreglo”(Cátedra, n.d., p.1.).

Una cadena de caracteres de tipo string puede ser representada por números, letras, símbolos, entre otros. Cada uno de estos puede ser de diferente tipo como flotante, decimal, entre otros. Las funciones de estas son variadas como:

- **char* strcpy(char destino[], char origen[]);**

“Esta función permite copiar cadenas de caracteres, desde la cadena origen a la cadena destino. Notar que la cadena destino debe ser de un tamaño mayor o igual que la cadena origen”(Cátedra, n.d., p.1.).

Esta nos permite poder copiar caracteres de la cadena principal o de la subcadena.

- **int strlen(char s1[]);**

“ Esta función permite averiguar la longitud de una cadena de caracteres”(Cátedra, n.d., p.2.).

La función cad.len() es una que nos permite conocer la longitud de la cadena que ubico el programador o a su vez colocó por teclado el usuario.

2. ¿Cómo se definen las cadenas de caracteres y cuál es su utilidad en la representación de texto en lenguajes de programación?

“Es un conjunto de caracteres que se almacenan en localidades contiguas de memoria. Podemos verlo como un vector de caracteres donde cada elemento representa un elemento del vector”(Diaz & Ve, n.d., p. 2).

Se define de esta manera :

char cadena [6];

Es un grupo de caracteres dentro de una cadena adentro de una memoria, este se puede comprender como una continuidad de caracteres donde cada componente es representado. Estas son utilizadas para realizar diferentes operaciones como hallar la longitud de la cadena, la transformación de las letras de una cadena en mayúsculas o minúsculas, entre otras.

3. ¿En qué aspectos las cadenas de caracteres son fundamentales para el manejo de información textual en la programación?

“Podría definirse como una notación o conjunto de símbolos y caracteres que se combinan entre sí siguiendo las reglas de una sintaxis predefinida, con el fin de posibilitar la transmisión de instrucción a un ordenador”(Peña Basurto & Cela Espín, 2000, p. 2).

Son fundamentales por lo que pueden almacenar conjuntos de valores, letras o símbolos y caracteres que contengan algo de similitud.

4. ¿Puede proporcionar ejemplos de situaciones específicas en las que las cadenas de caracteres son esenciales para la manipulación de datos en un entorno de programación?

-Manipulación de texto

Ejemplo de concatenación de cadenas de caracteres

```
cadena1 = "Hola"
```

```
cadena2 = " mundo"
```

```
resultado = cadena1 + cadena2
```

```
print(resultado) # Salida: Hola mundo
```

Ejemplo de búsqueda de subcadena

```
texto = "El lenguaje Python es poderoso"
```

```
if "Python" in texto:
```

```
    print("Se encontró la palabra 'Python'")
```

```
else:
```

```
    print("No se encontró la palabra 'Python'")
```

-Manipulación de cadenas en una página web

// Ejemplo de manipulación de cadenas en JavaScript

```
let mensaje = "Hola mundo";
```

```
let longitud = mensaje.length; // Obtener la longitud de la cadena
```

```
console.log("Longitud de la cadena:", longitud); // Salida: 11
```

```
let subcadena = mensaje.substring(0, 4); // Obtener una subcadena
```

```
console.log("Subcadena:", subcadena); // Salida: Hola
```

-Procesamiento de texto en un programa

```
public class ManipulacionCadenas {
```

```
    public static void main(String[] args) {
```

```
        // Ejemplo de manipulación de cadenas en Java
```

```
        String texto = "El lenguaje Java es versátil";
```

```
        int longitud = texto.length(); // Obtener la longitud de la cadena
```

```
        System.out.println("Longitud de la cadena: " + longitud); // Salida: 29
```

```
        String[] palabras = texto.split(" "); // Dividir la cadena en palabras
```

```
        System.out.println("Palabras en el texto:");
```

```
        for (String palabra : palabras) {
```

```
            System.out.println(palabra);
```

```
        }
```

```
    }
```

```
}
```

- **CONCLUSIONES Y RECOMENDACIONES**

CONCLUSIONES

- Se cumple que la función para hallar la extensión o longitud de una cadena se obtiene con la función `len("cadena")`. Y que el método `.strip()` retira y devuelve una cadena sin sus espacios de los extremos y los del medio.
- Se deduce que las cadenas son aquellas que almacenan la serie de elementos que se le ubique o a su vez el usuario los coloque.
- Se evidencia que los aspectos fundamentales de estas cadenas son las que recopila los conjuntos de símbolos.

RECOMENDACIONES

- Se recomienda que se utilicen cada una de las funciones cuando se presenten situaciones en los programas, también estos pueden ser utilizados dentro de bucles.
- Se recomienda que las cadenas almacenen los elementos con cada uno de ellos al grupo de cadena que se halla definido.
- Se recomienda que cuando se utilice conjuntos de símbolos se utilice la función de `cad.replace()`.

- **BIBLIOGRAFÍA**

Cátedra. (n.d.). *Unidad IV Cadenas de caracteres o Strings*.
Díaz, G., & Ve, G. (n.d.). *Programación Digital I*.
Peña Basurto, M. A., & Cela Espín, J. M. (2000). *Introducción a la programación en C*. UPC.

- **EVIDENCIA
EJERCICIO A**

```
#--coding:cp1252--
```

```
# Ejemplo A
```

```
#Realizar un programa que ingrese una cadena de caracteres por teclado que representa una frase
```

```
 #(palabras separadas por espacios), el programa debe contar cuantas palabras tiene la frase
```

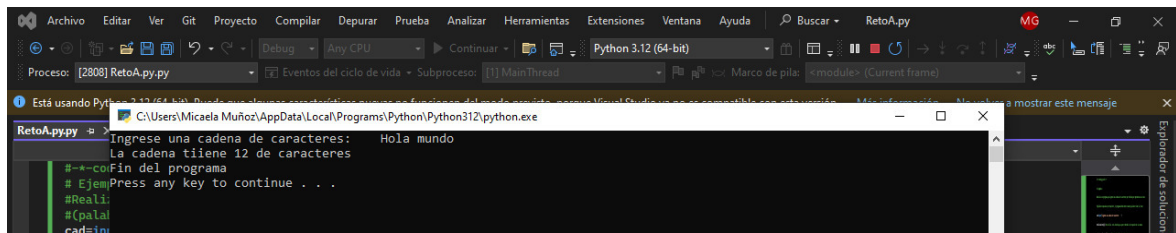
```
cad=input("Ingrese una cadena de caracteres: ")
```

```
total=cad.strip() #Se utiliza esta función para que se devuelva los espacios de la cadena
```

```
cadena=total.count(" ")#Se utiliza esta función para contar los caracteres
```

```
print(f"La cadena tiene",cadena+1,"de caracteres")
```

```
print("Fin del programa")
```



EJERCICIO B

`--coding:cp1252--`

#Realizar un programa que se ingrese una cadena de caracteres por teclado que represente una

frase (palabras separadas por espacios), el programa debe presentar la inicial de la letra

en mayúscula de cada palabra

`i = 0`

`iniciales = ""`

`frase = input("Introduce una cadena de caracteres,POR FAVOR: ")# Se solicita al usuario que introduzca una cadena de caracteres`

`while i < len(frase):`

`if frase[i] != ' ':`

`iniciales += frase[i]`

`while i < len(frase) - 1 and frase[i+1] != ' ':`

`i += 1`

`if i < len(frase) - 1 and frase[i+1] == ' ':`

`iniciales += ' '`

`i += 1`

`else:`

`i += 1`

`iniciales = iniciales.upper() # Funcion para convertir a mayusculas`

`print("Iniciales:", iniciales.replace(" ", ""))`

`print(" Fin del programa ")`

