

# Tarea 1

**Curso:** Programación con Python Nivel 1.

**Instructor:** Arturo Zamora (I-2023).

**Estudiante:** Luis Andrés Sánchez Núñez.

**Link de Repositorio:** <https://github.com/SpaceParrot21/Python-UCR>

**Link de Tarea 1:** [https://github.com/SpaceParrot21/Python-UCR/tree/main/Tareas/Tarea\\_1](https://github.com/SpaceParrot21/Python-UCR/tree/main/Tareas/Tarea_1)

## Instrucciones generales:

La tarea se debe entregar como un solo archivo, este debe contener el link al repositorio de la parte 1 y la documentación de la parte 2.

### Parte 1. Git

Cree su propio repositorio en GitHub y cargue en él el archivo del programa HelloWorld.py creado en la primera clase. Debe adjuntar en el documento de la tarea el link a su repositorio. Todas las tareas del curso se entregarán a través de los repositorios de cada uno.

**Respuesta:** [https://github.com/SpaceParrot21/Python-UCR/blob/main/Tareas/Tarea\\_1/HelloWorld.py](https://github.com/SpaceParrot21/Python-UCR/blob/main/Tareas/Tarea_1/HelloWorld.py)

### Parte 2. Resolución de problemas

Ejecute los 3 pasos de resolución de problemas para trazar una solución a cada uno de los siguientes problemas. Debe documentar cada uno de los pasos, además en el paso 2, debe crear una solución en pseudocódigo. La forma en la que desee crear la documentación depende de usted, debe entregar un único archivo en la tarea.

#### Problema 1.

Se necesita crear un programa que reciba del usuario una frase y decida si esa frase es un palíndromo o no. Un palíndromo se puede leer de igual forma de izquierda a derecha, que de derecha a izquierda. Ejemplo: "Anita lava la tina".

Paso 1: Entender el problema

- Necesitamos que el programa reciba una frase/cadena de caracteres de un usuario y logre determinar si la frase es un palíndromo.

## Paso 2: Planear

- El usuario ingresa una frase/cadena de caracteres.
- La frase/cadena de caracteres ingresados se almacena en una variable.
- Cada carácter/índice va a ser comparada por el programa.
- Si no hay coincidencia por algún carácter/índice el programa imprime que la palabra no es un palíndromo.
- Si hay coincidencia el programa imprime que la palabra no es un palíndromo.
- Fin del programa.

## Paso 3: Dividir y conquistar

- El usuario ingresa una frase/cadena de caracteres.
- La frase/cadena de caracteres ingresados se almacena en una variable llamada "frase".
- El programa invertirá la frase/cadena de caracteres original almacena en la variable "frase" y comparara cada carácter/índice con los caracteres/índices de la frase original.
- Si no hay coincidencia en algún carácter/índice el programa imprime que la palabra no es un palíndromo.
- Si hay coincidencia el programa imprime que la palabra no es un palíndromo.
- Fin del programa.

## Problema 2.

Un servidor crea logs por cada acción que se realiza en él. El administrador desea un programa que todos los días borre todos los logs excepto si el log contiene la palabra "error"; si contiene esta palabra, se debe copiar el log al directorio "Errores" y se debe enviar un correo al administrador.

### Paso 1: Entender el problema

- Necesitamos crear un programa que borre todos los logs almacenados que no contengan la palabra "error". Si encuentra un log que contenga la palabra "error" mover este log a el directorio "Errores" y enviarle un correo electrónico a la dirección del administrador del servidor.

### Paso 2: Planear

- El programa toma los logs del servidor.

- Inspeccionar todos los días los logs generados por un servidor en busca de la palabra “error”.
- Si el log no contiene la palabra “error”, elimina el log del servidor.
- Si hubo una coincidencia con la palabra “error”, entonces enviar un correo a la dirección de correo electrónico del administrador.
- Repetir esta tarea todos los días.
- Fin del programa.

### Paso 3: Dividir y conquistar

- Especificar la dirección donde se encuentran los logs generados por el servidor.
- Abrir el log generado por el servidor.
- Leer el contenido del log.
- Buscar una coincidencia con palabra “error” en el log que se leyó.
- Si la palabra “error” no se encuentra, eliminar el log del servidor.
- Si hubo una coincidencia con la palabra “error”, entonces enviar un correo al administrador.
- 
- Inspeccionar todos los días los logs generados por un servidor en busca de la palabra “error”.
- Si el log no contiene la palabra “error”, eliminar el log del servidor.
- Si contiene la palabra “error”, enviar un correo a la dirección de correo electrónico del administrador, permitir al programa conectar con el servicio de correo electrónico.
- Definir la dirección de correo electrónico al que se desea enviar el correo electrónico.
- Definir el “Subject” del correo.
- Definir el “Contenido” del correo.
- Adjuntar el log donde se encontró la palabra “error”
- Configurar la frecuencia con la que se realizara esta tarea.
- Fin del programa.

## Problema 3

Cree una solución que permita al usuario ingresar un número entero. Dado dicho número, el programa debe determinar si los dígitos de este número se pueden ordenar de forma tal que el resultado sea un múltiplo de 5.

### Paso 1: Entender el problema

- Crear un programa que reciba un numero entero por parte de un usuario. El programa tiene que ordenar o realizar una operación para que el resultado sea un múltiplo de 5. Los múltiplos de 5 terminan en 0, o en 5.

## Paso 2: Planear

- El usuario ingresa un numero entero.
- Determinar si el numero ingresado es un múltiplo de 5.
- Si no es un múltiplo de 5, multiplicar el numero por 5 para que dé como resultado un múltiplo de 5.

## Paso 3: Dividir y conquistar

- Indicarle al usuario que debe ingresar un numero entero.
- El usuario ingresa un numero entero.
- Guardar el numero ingresado en una variable.
- Verificar si el numero ingresado es un decimal, indicarle al usuario que debe ingresar un numero entero.
- Determinar si el numero ingresado es un múltiplo de 5. Verificar si el numero termina en 0 o en 5. O alternativamente dividir el numero ingresado entre 5, como resultado deber dar un numero entero.
- Si el ultimo número/índice del número ingresado termina en 0 o 5 indicarle al usuario que el número es un numero múltiplo de 5.
- Si el ultimo número/índice del número ingresado no termina en 0 o en 5 que lo multiplique por 5. El Resultado sería un múltiplo de 5.
- Mostrar