

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

Facultad de Ciencias Químicas e Ingeniería

Programas de Ingeniero en Computación e Ingeniero en Software y Tecnologías Emergentes

INFORMACIÓN DE LA MATERIA

Nombre de la materia y clave: Lenguaje de Programación Python (36305).

Grupo y periodo: 532 (2022-2)

Profesor: Manuel Castañón Puga.

INFORMACIÓN DE LA ACTIVIDAD

Nombre de la actividad: Práctica de laboratorio 2.4.2 Errores y excepciones

Lugar y fecha: A 3 de diciembre de 2022 en el Edificio 6E, Salón 204.

Carácter de la actividad: Individual.

Participante(es): Emmanuel Alberto Gómez Cárdenas

REPORTE DE ACTIVIDADES

1. Utilice el repositorio en GitHub con el portafolio de prácticas de laboratorio que creó en la Meta 1.2.
2. Clone el repositorio en su computadora y agregue una carpeta de código para la Actividad de taller 1.3.3. Puede hacerlo utilizando una IDE (Visual Studio Code, PyCharm, etc.).

3. Mejore el programa de la actividad de taller 2.4.1 (**Escribir un programa en Python que implemente un traductor de texto con caracteres alfanuméricos a caracteres en morse**) agregando la funcionalidad necesaria para prevenir errores y manejar excepciones utilizando la instrucción **try**.

```
def morse2Text(morse):
    morse_code = {'A': '.-', 'B': '-...', 'C': '-.-.', 'D': '-..', 'E': '.',
                  'F': '..-', 'G': '--.', 'H': '....', 'I': '..', 'J': '-.---',
                  'K': '-.-', 'L': '.-..', 'M': '--', 'N': '-.', 'O': '---',
                  'P': '-.-.', 'Q': '--.-', 'R': '.-.', 'S': '...', 'T': '-',
                  'U': '..-', 'V': '...-', 'W': '---', 'X': '-.-.-', 'Y': '-.-.-',
                  'Z': '--..', '1': '.----', '2': '..---', '3': '...--',
                  '4': '....-', '5': '.....', '6': '-.....', '7': '--....',
                  '8': '---...', '9': '----.', '0': '-----'}

    morse_text = ''
    temp = ''
    flag = False
    for i in range(len(morse)):
        if morse[i] == '.' or morse[i] == '-':
            temp += morse[i]
            flag = True
        elif morse[i] == ' ': # si es un espacio
            if morse[i-1] == ' ':
                morse_text += ' '
            elif flag:
                morse_text += list(morse_code.keys())
                               [list(morse_code.values()).index(temp)]
                flag = False
                temp = ''
            else:
                try:
                    morse_text += list(morse_code.keys())
                                   [list(morse_code.values()).index(temp)]
                    flag = False
                    temp = ''
                except ValueError:
                    print("\n\nError: Caracter: '" + morse[i] + "' no valido")
                    return
        if flag:
            morse_text += list(morse_code.keys())
                           [list(morse_code.values()).index(temp)]

    return morse_text
```


4. Respalde (commit) y suba (push) su código en el repositorio de GitHub para hacer la entrega de la práctica.



Enlace al repositorio de GitHub:

<https://github.com/LPP-FCQI-UABC/lpp-2022-2-532-portafolio-AlbGmx>

RESUMEN/REFLEXIÓN/CONCLUSIÓN

El manejo de excepciones es importante ya que sin este cada que haya un problema el programa dejaría funcionar de manera incorrecta o dejaría de funcionar completamente.

Al atrapar los errores se evita que este se propague hacia otras funciones y ocasione más problemas.

Atrapar el error es solo el primer paso, también es importante llevar un registro del porqué y en donde se generó dicho problema, así es más sencillo minimizar o eliminarlo por completo.

Doy fe de que toda la información dada es completa y correcta.

Nombre y firma del alumno.

Gómez, Emmanuel A.