

UNIVERSIDAD DE GUADALAJARA, CUCEI

Generar un programa que sea capaz de restaurar el estado de ejecución.

ALUMNO:

Ruiz Salcedo Carlos Salvador

Profesor: MICHEL EMANUEL LOPEZ FRANCO

I7036 - Computación Tolerante a Fallas

2023A

sección: D06

OBJETIVO:

Desarrollar un programa que restaure el estado de ejecución de un programa

DESARROLLO DE LA ACTIVIDAD:

Librería pickle:

El modulo pickle implementa protocolos binarios para serializar y deserializar una estructura de objetos Python. «*Pickling*» es el proceso mediante el cual una jerarquía de objetos de Python se convierte en una secuencia de bytes, y el «*unpickling*» es la operación inversa, mediante la cual una secuencia de bytes de un archivo binario (binary file) ó un objeto tipo binario (bytes-like object) es convertido nuevamente en una jerarquía de objetos. *Pickling* (y *unpickling*) son alternativamente conocidos como «serialización», «ensamblaje», 1 o «aplanamiento»; sin embargo, para evitar confusiones, los términos utilizados aquí son «pickling» y «unpickling»

Implementación:

Para la implementación de este programa se utilizo Python y el ide de pycharm donde gracias a la herramienta pickle nos ayuda con archivos serializables los cuales utilizaremos para guardar el progreso del programa

```
import pickle

List = ["1", "2", "3", "4"]

ArchivoBinario = open("ArchivoLista", "wb")
pickle.dump(List, ArchivoBinario)
ArchivoBinario.close()

Archivo2 = open("ArchivoLista", "rb")
|
Lista2 = pickle.load(Archivo2)

print(Lista2)
```

Aquí podemos observar como se ingresan datos por defecto posteriormente se crea el archivo y hacemos uso de la librería dump tomando la lista en un archivo binario, posteriormente lo cerramos para guardar el contenido y para visualizar el contenido abrimos el archivo cargamos los datos y los imprimimos.

```

C:\Users\Administrator\PycharmProjects\ser\venv\Scripts\python.exe C:\Users\Administrator\PycharmProjects\ser\main.py
['1', '2', '3', '4']
Process finished with exit code 0

```

Nombre	Fecha de modificación	Tipo	Tamaño
.idea	06/02/2023 11:19 p. m.	Carpeta de archivos	
venv	06/02/2023 11:13 p. m.	Carpeta de archivos	
ArchivoLista	06/02/2023 11:27 p. m.	Archivo	1 KB
main.py	06/02/2023 11:47 p. m.	Python File	1 KB

```

['1', '2', '3', '4']

```

CONCLUSIONES:

Una vez que he revisado esta técnica que se enfoca en la revisión de código me ha parecido muy útil conocerla de una mejor manera ya que para nuestra clase de tolerancia a fallas mas concretamente en software utilizar estos parámetros e indicaciones que podemos leer en este documento me parecen sumamente de gran ayuda ya que podemos observar las distintas tipos de pruebas que se le pueden hacer al código y de ahí poder obtener resultados de posibles fallos y poder controlarlos.

BIBLIOGRAFIA:

IEEE Transactions on Software Engineering (Volume: 18, Issue: 11,
November 1992)