

CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E
INGENIERÍA
DEPARTAMENTO DE CIENCIAS COMPUTACIONALES



SEMINARIO DE SOLUCIÓN DE PROBLEMAS DE
SISTEMAS OPERATIVOS
BECERRA VELZQUEZ VIOLETA DEL ROCIO
INGENIERIA EN COMPUTACIÓN
SECCION D01

PACHECO ROMERO VICTOR MANUEL
216589519

LINK VIDEO

https://drive.google.com/file/d/1p9_JffnxNfb-g_-8vsYR2xcCNP1QkLVA/view?usp=sharing

índice

Objetivo.....	4
Desarrollo	4
Conclusiones.....	5

Índice de ilustraciones

Índice de imágenes

Ilustración 1- Llenado automático de datos	4
Ilustración 2-Función de teclas.....	5

Objetivo

Desarrollar un programa de procesamiento por lotes con multiprogramación.

Desarrollo

Para el desarrollo de esta actividad utilice Python, al comenzar la practica lo primero que implemente fue una función la cual me serviría para llenar automáticamente mis datos además de hacer las modificaciones correspondientes que la maestra solicito para casa uno de estos datos ingresados automáticamente.

```
def llenado_automatico(num_lotes,i):  
    os.system("cls")  
    id = i+1  
    num1 = (random.randrange(0, 100))  
    num2 = (random.randrange(0, 100))  
    operacion = (random.randrange(1, 5))  
    tiempo_estimado = (random.randrange(6, 16))  
  
    if((operacion == 4 or operacion == 5) and num2 == 0):  
        num2 = (random.randrange(1, 100))  
        continuar = (num2==0)  
  
    lote = Lote(id,operacion,tiempo_estimado,num1,num2, num_lotes)  
    ejecutar_lotes.append(lote)
```

Ilustración 1- Llenado automático de datos

Seguidamente de esto implemente la funcion kbhit, la cual me permitiría ingresar datos de el teclado y si este correspondía con los asignado me haria una funcion distinta, la letra "I" me serviría para hacer una interrupción por entrada-salida, la letra "E" me retornaría error en el proceso que se esta ejecutando, la letra "P" pausaría el programa hasta que sea presionada la letra "C", esta letra me permitiría conitnuar con el programa justo donde lo pause, de ser presionada otras teclas distintas a las mencionadas el programa seguiría ejecutandose sin problema alguno.

Para implementar la letra "I" tuve que apilar y desencolar los datos de mi lista para mantener los datos actules y continuar con el siguiente proceso. Para la letra "E" solamente tuve que agregar una condicion en la cual si se presionaba el programa me mandaría ERROR en vez de el resultado de la operación. Para la letra "P" solo lo meti en una condicion para que esta no saliera hasta que la letra "C" fuera presionada para continuar con el programa.

```
def teclas(num_lote, cant_lotes, tiempo):
    salida = False
    condicion = True
    if msvcrt.kbhit():
        tecla = msvcrt.getwch()
        if(tecla == 'i' or tecla == 'I'):
            lote_actual[0].tiempo_estimado = tiempo
            lote_actual.append(lote_actual[0])
            lote_actual.remove(lote_actual[0])
            ejecucion_de_lotes(num_lote, cant_lotes)
            salida = True

        elif(tecla == 'e' or tecla == 'E'):
            lote_actual[0].tiempo_estimado = 0
            lote_actual[0].error = 1
            salida = True

        elif(tecla == 'p' or tecla == 'P'):
            while(condicion):
                c = input("Presiona la letra c para continuar con el proceso: ")
                if(c == "c" or c == "C"):
                    break
                else:
                    print("Esa no es la letra c")
                    continue
            salida = True

    return salida
```

Ilustración 2-Función de teclas

Conclusiones

Esta actividad me pareció bastante interesante, además pude aprender un nuevo método que fue el “Kbhit” para recibir entradas con el teclado y como ejecutar esas entradas en el programa, sinceramente fue una actividad que me costo debido a la manipulación de datos que se encontraban en la lista, sobre todo tuve problemas con la letra “l” ya que fue algo complicado entender como apilar los datos y como hacer que se encolaran y desencolaran para añadirse en el proceso, pero finalmente pude concluir la actividad, me gusto gracias a que pude entender y simular el procesamiento por lotes con multiprogramación.