



**Instituto Politecnico Nacional**



**ESCOM “ESCUELA SUPERIOR DE CÓMPUTO”**

*TEORÍA COMPUTACIONAL*

*PRÁCTICA 1: EXPRESIONES REGULARES*

PROFE: Luz María Sánchez García

ALUMMNO: Rojas Alvarado Luis Enrique

GRUPO: 2CM11

## INTRODUCCION

El proposito de la práctica consistió en construir una expresión regular para comprobar si una expresión es correcta o no, probando cadenas hasta que la expresion se cumpla y pobando todas las etradas posibles para determinar si son cadenas validas o no de acuerdo al tipo de expresion desarrollada con anterioridad, así como conocer las dierentes reglas para construir dicha expresión y poder dar solucion a un problema que requiere uso de éstas para ser validada posteriormente.

## PLANTEAMIENTO DEL PROBLEMA

El problema que se presenta dependiendo del tipo de expresión que se desee realizar, ya sea un CURP, una fecha, o un teléfono, etc, es validar si una cadena escrita por el usuario corresponde a alguna de las expresiones compuestas por caracteres alfanumericos y determinar si lo que se ingresó coinside con la expresión a analizar. En este caso se eligió trabajar para validar un grupo de la ESCOM del cual más adelante se especificará su estructura.

Para darle solución a este problema, se opta por usar cualquier lenguaje de programacion que acepte expresiones regulares una vez desarrolladas con anterioridad para la proceder a validar si lo que ingresa el usuario es correcto o no. Una vez implementada la exresión en código se valida para cualquier cadena válida y no válida.

## DISEÑO DE LA SOLUCIÓN

Para la solución de este problema dado se eligió como programa para desarrollar estas expresiones a python ya que acepta las expresiones regulares con una librería ya implementada.

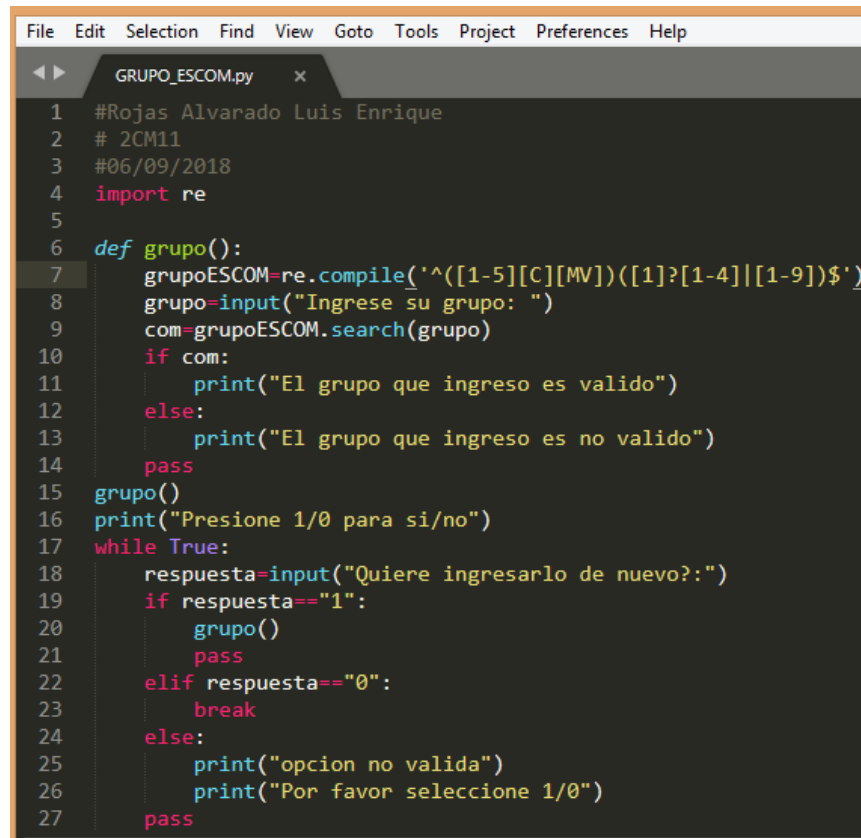
La estructura y expresion regular desarrollada para un grupo de ESCOM es la siguiente:

NIVEL	CARRERA	TURNO	GRUPO	
2	C	M	1	1
[1-5]	[C][MV]		[1]?[0-4][1-9]	

- **NIVEL:** El nivel sólo va de 1 al 5 ya que la carrera cuenta con 5 niveles.
- **CARRERA:** La carrera es fija ya que sólo existe 1 en ESCOM
- **TURNO:** Sólo puede ser matutino (M) ó vespertino (V).
- **GRUPO:** Está sólo en el rango de 1-14 puesto que solo hay 14 grupos

## IMPLEMENTACIÓN DE LA SOLUCIÓN

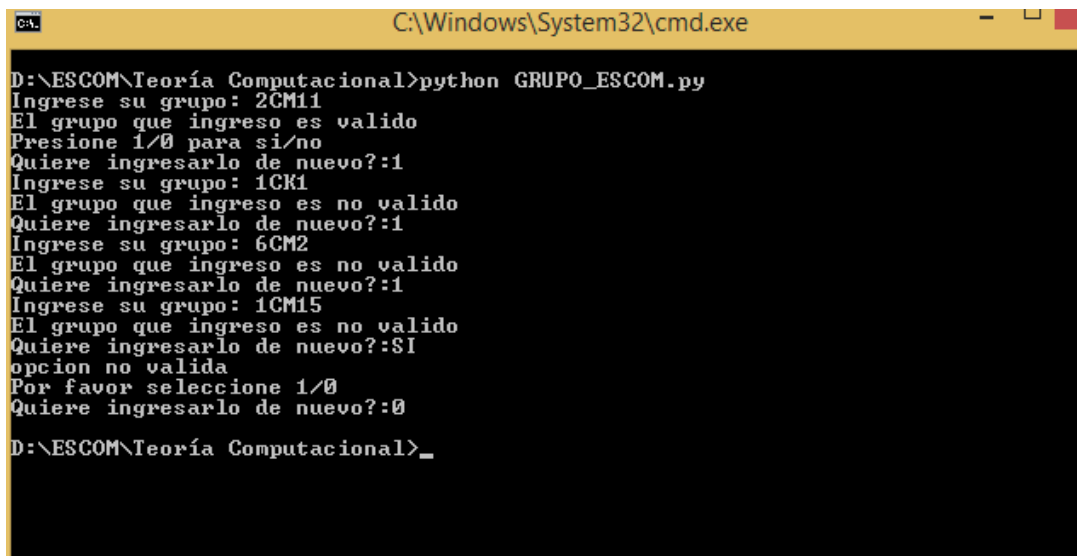
El código en python para la expresión anterior es el siguiente:



```
File Edit Selection Find View Goto Tools Project Preferences Help
GRUPO_ESCOM.py x
1 #Rojas Alvarado Luis Enrique
2 # 2CM11
3 #06/09/2018
4 import re
5
6 def grupo():
7     grupoESCOM=re.compile('^([1-5][C][MV])([1]?[1-4]|[1-9])$')
8     grupo=input("Ingrese su grupo: ")
9     com=grupoESCOM.search(grupo)
10    if com:
11        print("El grupo que ingreso es valido")
12    else:
13        print("El grupo que ingreso es no valido")
14    pass
15    grupo()
16    print("Presione 1/0 para si/no")
17    while True:
18        respuesta=input("Quiere ingresarlo de nuevo?:")
19        if respuesta=="1":
20            grupo()
21            pass
22        elif respuesta=="0":
23            break
24        else:
25            print("opcion no valida")
26            print("Por favor seleccione 1/0")
27    pass
```

## FUNCIONAMIENTO

Se corre el programa en consola de windows:



```
C:\Windows\System32\cmd.exe
D:\ESCOM\Teoría Computacional>python GRUPO_ESCOM.py
Ingrese su grupo: 2CM11
El grupo que ingreso es valido
Presione 1/0 para si/no
Quiere ingresarlo de nuevo?:1
Ingrese su grupo: 1CK1
El grupo que ingreso es no valido
Quiere ingresarlo de nuevo?:1
Ingrese su grupo: 6CM2
El grupo que ingreso es no valido
Quiere ingresarlo de nuevo?:1
Ingrese su grupo: 1CM15
El grupo que ingreso es no valido
Quiere ingresarlo de nuevo?:SI
opcion no valida
Por favor seleccione 1/0
Quiere ingresarlo de nuevo?:0
D:\ESCOM\Teoría Computacional>_
```

## CONCLUSIONES

En ésta práctica se realizó una expresion regular para validar (en mi caso) un grupo de ESCOM dada ya su estructura. Dependiendo que tan bien implementada esté la expresión regular hecha con anteoridad se refljará en la aplicación e implementación de la misma puesto que al no estar correcta, se empezará a validar cosas que ni siquiera se quieren validar. Aplicar correctamente los comandos para la impementación de la epresión regular y hacer un buen uso de estas dará como resultado un buen funcionamiento de tu programa al momento que el usuario quiera ingresar cualquier tontería o el grupo correcto.

## BIBLIOGRAFÍA

- 1) Platzi, 2015, GUuía de expresiones regulares en Python, <https://platzi.com/blog/expresiones-regulares-python/> recuperado el 05/09/2018.