

### ¿Que se hizo?

Se hicieron dos programas uno llamado kmp y otro llamado validación de correo en el cual el primero corresponde a buscar patrones en un determinado tamaño de una cadena en cuanto el Segundo valida si un correo electrónico cuenta con el formato para dar por hecho que es un correo electrónico.

### ¿En dónde se realizó?

Se realizó en el IDLE de Python 3.7 de 32-bits

### ¿Cómo se realizó?

De acuerdo a los conocimientos de expresiones regulares se implementaron código como `re.match('[a-z]', cadena)` para determinar que la cadena tenga los caracteres señalados

### ¿Para qué se realizó?

Se realizó el primeo para buscar patrones en una determinada cadena y después enlistarlos para saber en qué posición se ubican determinado patrón.

El segundo se realizó para validar mediante expresiones regulares un correo electrónico con una señal de entrada para el correo electrónico dado.

### seudocódigo KMP

1. Se ingresa la cadena a observar
2. Se ingresa el patrón determinado
3. Se genera un for en un rango de (0, al tamaño de la cadena)
4. Se ingresa una variable "c" el cual se encarga de terminar de donde a donde se observará el patrón.
5. Se ingresa una variable que busca el patrón en la variable anterior.
6. Se agrega una condición if para revisar si se ha encontrado el patrón en la variable "c".
7. Se imprime la posición del patrón si se ha encontrado tal.

Cadena	Patrón	Tiempo(segundos)
10	1	1.2
10	2	1.04
10	3	1

10	4	0.98
10	5	1.04
10	6	1
10	7	0.99
10	8	1.07
10	9	1.29
10	10	0.95

### **seudocódigo VALIDACION.**

1. Se ingresa el correo a validar
2. Se busca la posición del "@" y el ".com"
3. Se realiza un for i en el rango (0,a): donde "a" es la posición del "@"
4. Se agrega una variable "d"el cual va tomar carácter por carácter la expresión regular.
5. Se agrega un if para saber si la variable d es Falsa si es así se agrega 1 a al contador "x".
6. Se agrega otro for j en el rango de ((a+1), b) donde "b" es la posición del ".com"
7. Se repite el paso 4,5 obviamente con sus variables y expresión regular correspondiente
8. Por último, se incorpora un if el cual verifica las condiciones y determina si el correo es válido o no es válido.

### **Conclusión...**

En el caso de búsqueda, remplazar, formatos cadena entre otras cosas las expresiones regulares nos facilitan el trabajo al emplearlo en el.

### **Referencia...**

<https://librosweb.es/libro/python/capitulo-6/metodos-de-validacion.html>

<https://platzi.com/blog/expresiones-regulares-python/>