

## Programación – Ejercicios Adaptados de Certamen 2 de 2019-2

1. Realice el ruteo del siguiente programa e indique en el recuadro de la derecha lo que imprime como salida. Cada vez que el valor de una variable cambie, escríbalo en una nueva fila de la tabla. Si una variable es de tipo string, su valor debe ir entre comillas simples ' ' o dobles " ". La tabla tiene suficientes filas.

```
def f(p):
    if '-' in p:
        x = p[1]
    else:
        x = p[3]
    return x
```

```
q = 'HBRU 93'
s = f(q)
q = 'XS-7122'
c = f(q)
s = s + c
q = 'CM-4513'
c = f(q)
s = s + c
print(s)
```

**Salida:**

--

[illegible]

## Programación – Ejercicios Adaptados de Certamen 2 de 2019–2

2. La policía de Pythonia le ha encargado construir un programa para cursar multas a quienes no respeten la restricción vehicular, la que se define a partir de las patentes de los autos.

Existen patentes con dos formatos distintos:

- 4 letras, un espacio y 2 dígitos (Ejemplo: 'CRTJ 32').
- 2 letras, un guión y 4 dígitos (Ejemplo: 'RX-2134').

Para definir la restricción de los **catalíticos** se analiza la última letra de la patente, es decir, la letra que está más a la derecha:

- Lunes: patentes cuya última letra sea menor o igual a 'G'.
- Miércoles: patentes cuya última letra sea mayor que 'G' pero menor o igual a 'N'.
- Viernes: patentes cuya última letra sea mayor que 'N'.

Para el caso de los **no catalíticos** la restricción se decide en base al último dígito:

- Lunes: Patentes terminadas en 0, 1, 2 o 3.
- Miércoles: Patentes terminadas en 4, 5 o 6.
- Viernes: Patentes terminadas en 7, 8 o 9.

El resto de los días no hay vehículos con restricción, de ningún tipo.

Tome en cuenta que el formato de la patente no tiene relación alguna con el tipo de auto: catalítico o no catalítico.

Escriba la función `tiene_restriccion(patente, catalitico, dia)` que recibe como parámetros la patente de un vehículo particular, un valor booleano que es `True` si el auto es catalítico y `False` si no lo es, y el día de la semana en que el vehículo transitó (todo en mayúscula). La función debe **retornar** `True` si se le debe cursar un parte al vehículo o `False` en caso contrario. Puede suponer que todos los datos son correctos y se apegan a los formatos especificados.

Ejemplo:

```
>>> tiene_restriccion('CRTJ 32', True, 'LUNES')
False
>>> tiene_restriccion('ZZ-9999', True, 'VIERNES')
True
>>> tiene_restriccion('RX-2134', False, 'MIERCOLES')
True
```