

# GUIÓN DE LA PRÁCTICA 5

---

## OBJETIVO:

- Resolución de problemas con algoritmos de programación dinámica

## Coincidencia de patrones con textos

Dado un texto y un patrón diremos que el patrón coincide con el texto si dicho patrón es capaz de representar al texto completo mediante sus caracteres.

Consideraciones sobre el **texto**:

- El texto es cualquier cadena no vacía formada por letras (cada letra es una de estas 52: ["a".."z" , "A".."Z"]).

Consideraciones sobre el **patrón**:

- El patrón es cualquier cadena no vacía formado por letras (["a".."z" , "A".."Z"]) y los caracteres "\*" o "?".
- "\*" hace referencia a una secuencia de caracteres cualquiera (incluido la **secuencia vacía**).
- "?" hace referencia a un carácter cualquiera (incluido el **carácter vacío**).

Para resolver este problema con programación dinámica podemos crear una tabla en la que las filas son las letras del texto y las columnas son las letras del patrón a verificar. Los valores de la tabla serán verdaderos (T) cuando el patrón hasta una cierta columna coincide con el texto hasta una cierta fila. Será falso (F) si el patrón no coincide con el texto correspondiente.

Se podría hablar de diferentes casos:

1. **Caso base**: si el patrón pudiera estar vacío, solo coincidirá con el texto si el texto también estuviere vacío. Y viceversa. (ver [letras rojas de los ejemplos](#)).
2. **Cuando ambas letras coinciden**: cuando en la fila y en la columna tenemos la misma letra entonces pueden solo puede ocurrir una cosa para que el patrón coincida con el texto:
  - Si en la posición anterior de cada cadena (fila anterior y columna anterior) había coincidencia, ahora seguirán coincidiendo tras añadir la nueva letra (ver [ejemplo 1](#)).
3. **Cuando ambas letras no coinciden**: cuando en la fila y en la columna tenemos distintas letras, entonces el patrón no coincide con el texto en esa posición específica (ver [ejemplo 2](#)).
4. **Cuando encontramos una "?"**: cuando en la columna encontramos una "?" pueden ocurrir dos cosas para que el patrón coincida con el texto:

- Si en la posición anterior de cada cadena (fila anterior y columna anterior) había una coincidencia, ahora seguirán coincidiendo tras añadir la letra en lugar de la "?", ya que nos sirve cualquier letra (ver ejemplo 3).
  - Si en la posición anterior del patrón había una coincidencia (misma fila y columna anterior), ahora seguirán coincidiendo si ignoramos a la "?", por aquello de que podemos hacer como si no existe (carácter vacío vale) (ver ejemplo 3).
5. **Cuando encontramos un "\*" :** cuando en la columna encontramos un "\*" pueden ocurrir tres cosas para que el patrón coincida con el texto:
- Si en la posición anterior de cada cadena (fila anterior y columna anterior) había una coincidencia, ahora seguirán coincidiendo tras añadir la letra en lugar del "\*", ya que nos sirve cualquier letra (ver ejemplo 4).
  - Si en la posición anterior del patrón había una coincidencia (misma fila y columna anterior), ahora seguirán coincidiendo si ignoramos al "\*", ya que podemos hacer como si no existe (secuencia vacía vale) (ver ejemplo 4).
  - Si en la posición anterior del texto había una coincidencia (fila anterior y misma columna), ahora seguirán coincidiendo, ya que un "\*" puede representar a cualquier número de letras del texto (ver ejemplo 4).

**Ejemplo 1:**

- Texto: **casa**
- Patrón: **casa**
- Coincidencia: **true**

		c	a	s	a
	T	F	F	F	F
c	F	T	F	F	F
a	F	F	T	F	F
s	F	F	F	T	F
a	F	F	F	F	T

**Ejemplo 2:**

- Texto: **casa**
- Patrón: **cosa**
- Coincidencia: **false**

		c	o	s	a
	T	F	F	F	F
c	F	T	F	F	F
a	F	F	F	F	F
s	F	F	F	F	F
a	F	F	F	F	F

**Ejemplo 3:**

- Texto: **casa**
- Patrón: **ca?a**
- Coincidencia: **true**

	c	a	?	a
T	F	F	F	F
c	F	T	F	F
a	F	F	T	T
s	F	F	F	T
a	F	F	F	F

**Ejemplo 4:**

- Texto: **casa**
- Patrón: **\*\*a**
- Coincidencia: **true**

	*	*	a
T	F	F	F
c	F	T	T
a	F	T	T
s	F	T	T
a	F	T	T

**TRABAJO PEDIDO**

SE PIDE diseñar e implementar un algoritmo utilizando programación dinámica para resolver este problema de forma óptima.

- Implementar en Java dicho algoritmo de tal forma que calcule para una cadena de texto concreta si los diferentes patrones pasados en el fichero de texto coinciden con dicha cadena o no.
- Razonar qué complejidad tiene el algoritmo diseñado. ¿Crees que se podría cambiar el diseño para mejorar la complejidad?

Los datos de entrada se leerán desde un fichero de texto con el formato especificado:

Por ejemplo, fichero **“test3.txt”**:

```
casa
casa true
cosa false
ca?a true
**a true
```

Representando lo siguiente:

- Primera línea: **texto que se quiere comprobar**
- Segunda y posteriores líneas: **información sobre los patrones**
  - o Primera parte: el patrón
  - o Segunda parte: si el patrón debe coincidir con el texto o no.

El programa deberá mostrar los resultados finales obtenidos para el fichero dado, incluyendo la tabla de cómo quedan todas las soluciones parciales antes de obtener la solución final para cada caso de prueba.

Se ha de entregar en un **.pdf** el trabajo que se le pide y además las clases **\*.java** que ha programado. Todo ello lo organizará en una carpeta, que es la que entregará comprimida en un fichero **practica5ApellidosNombre.zip**.

Esta práctica es para una semana lectiva y el **plazo límite** de entrega de esta práctica es **un día antes** de que comience la siguiente práctica, en la tarea que se creará a tal efecto en el Campus Virtual.

