

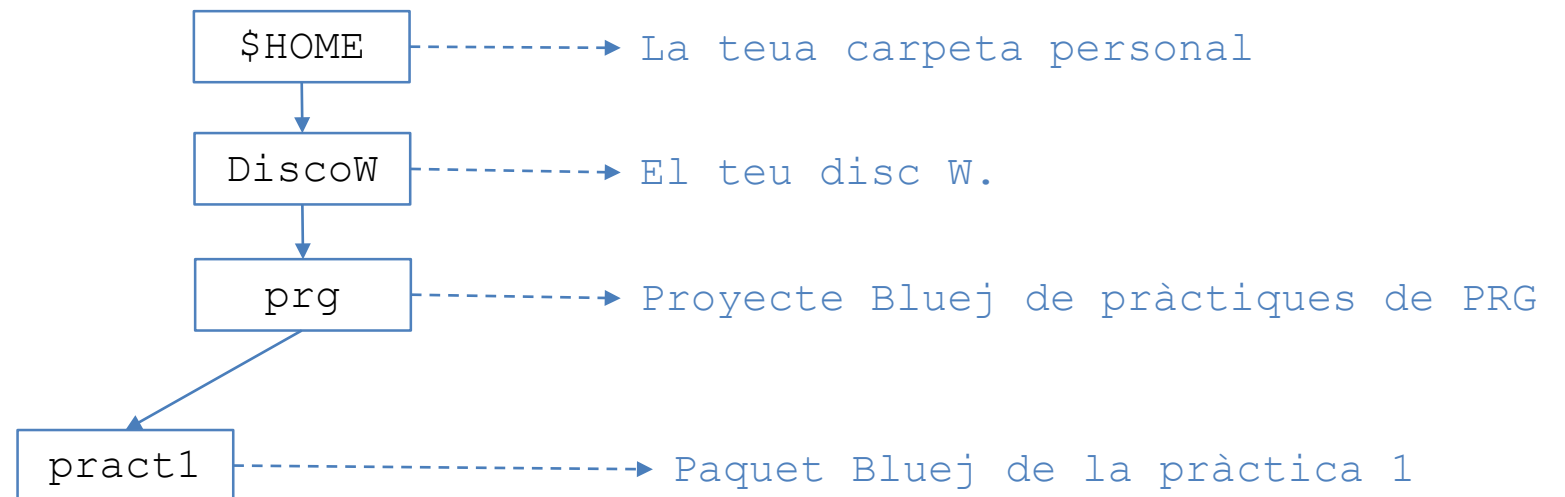
# Pràctica 2 PRG (2019/2020)

RECURSIÓ II

DURACIÓ 1 SESIÓN

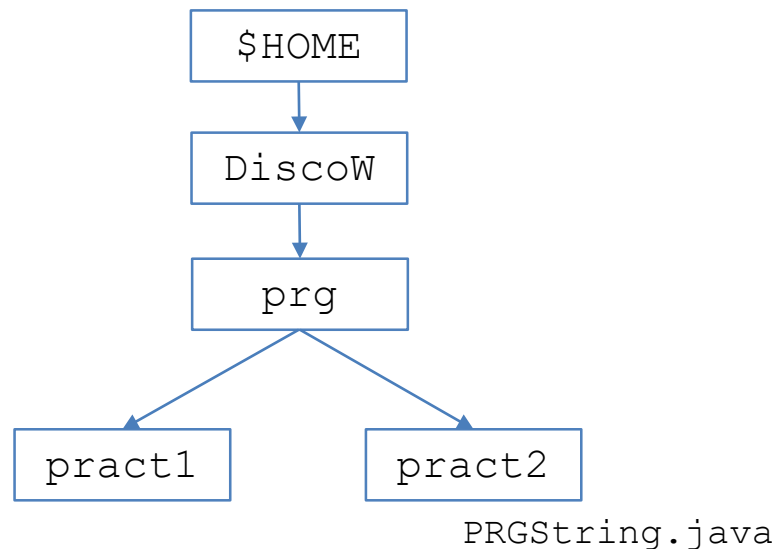
# Recordatori:

- ▶ Si no has finalitzat la pràctica anterior recorda fer-ho abans del parcial.
- ▶ L'estructura de carpetes que has de tindre per a començar aquesta pràctica és:



# PRÀCTICA 2: Activitat 1

1. Crea dins del projecte un paquet **pract2** per a la segona pràctica.
2. Descarrega, de **PoliformaT** la classe **PRGString.java** i afegeix-la al paquet **pract2**.





# PRÀCTICA 2: Activitat 2

► Completa el mètode **isPrefix(String)** de la classe **PRGString.java**

1. Capçalera del mètode:

```
public static boolean isPrefix(String a, String b)
```

2. Documenta el mètode: Paràmetres, Resultat, Precondició...

3. Cas Base: qual es el problema de talla més xicoteta possible i la seua solució directa?

4. Cas Recursiu, per a el resto de casos:

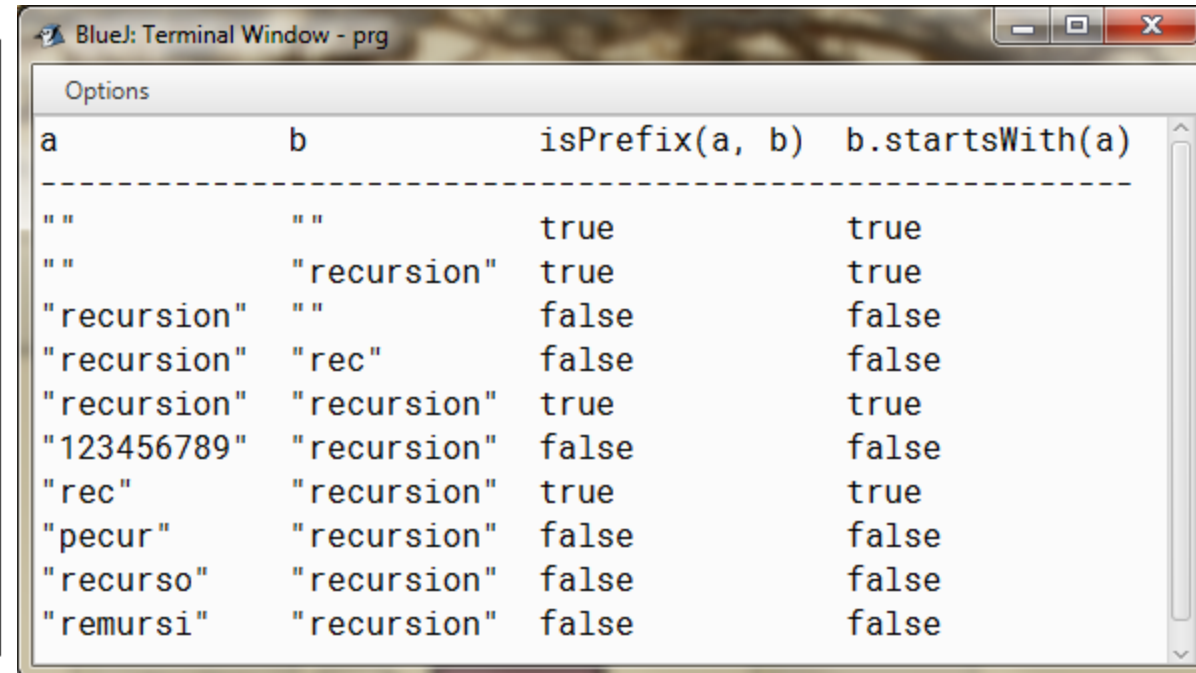
- Es tracta d'un Recorregut o d'una Cerca?
- Si es tracta d'una cerca: qual es la condició per a trobar allò que ens fa para? Pot ser més d'una.
- Qual es la crida recursiva que redueix el problema i aconsegueix la precondició?
- Que li falta al mètode per a resoldre el problema?
- Hi ha que combinar la crida recursiva amb algun paràmetre o no?

- 
1. Dissenyar
  2. Implementar
  3. Compilar
  4. CheckStyle
  5. Provar casos

# PRÀCTICA 2: Activitat 3

- ▶ Valida el mètode `isPrefix(String)` de la classe `PRGString.java`
- Crea una classe `TestIsPrefix.java` ha d'incloure un `main` que provei tots els casos cridant a un mètode amb la capçalera: `private static void testIsPrefix(String a, String b)` que escriba en pantalla les dues `String`, el resultat obtingut al invocar al mètode i el resultat esperat (pots usar el mètode `startsWith()` dels objectes `String`).

Caso	a	b	Resultado
a y b vacías	""	""	true
Solo a vacía	""	"recursion"	true
Solo b vacía	"recursion"	""	false
a de mayor longitud que b	"recursion"	"rec"	false
a y b de igual longitud y a es prefijo de b	"recursion"	"recursion"	true
a y b de igual longitud y a no es prefijo de b	"123456789"	"recursion"	false
a de menor longitud que b y a es prefijo de b	"rec"	"recursion"	true
a de menor longitud que b y a no es prefijo de b:			
- por el primer carácter	"pecur"	"recursion"	false
- por el último carácter	"recurso"	"recursion"	false
- por un carácter intermedio	"remursi"	"recursion"	false



BlueJ: Terminal Window - prg

Options

a	b	isPrefix(a, b)	b.startsWith(a)
""	""	true	true
""	"recursion"	true	true
"recursion"	""	false	false
"recursion"	"rec"	false	false
"recursion"	"recursion"	true	true
"123456789"	"recursion"	false	false
"rec"	"recursion"	true	true
"pecur"	"recursion"	false	false
"recurso"	"recursion"	false	false
"remursi"	"recursion"	false	false

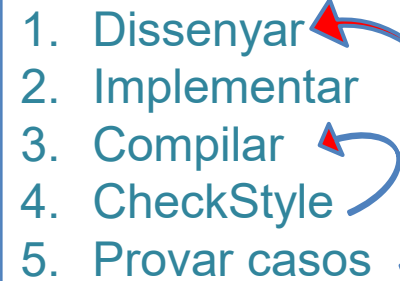
# PRÀCTICA 2: Activitat 4

- Completa el mètode `isSubstring(String)` de la classe **PRGString.java** aprofitant que ja tens implementat el mètode: `isPrefix(String)`

1. Capçalera del mètode:

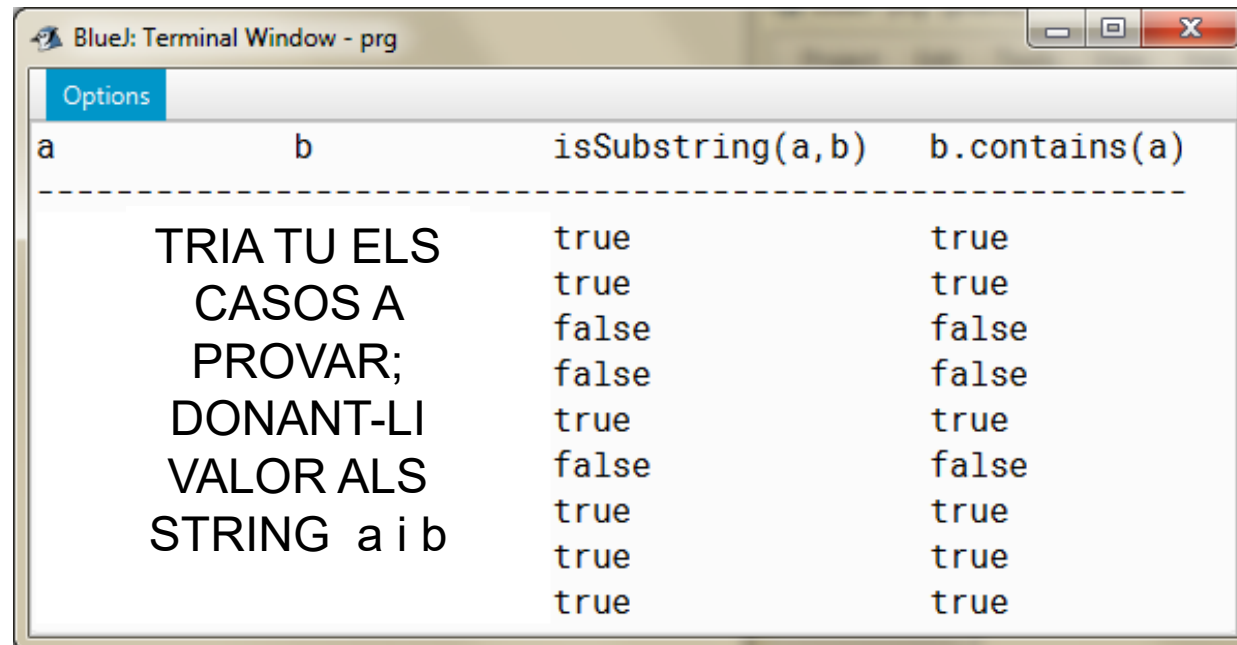
```
public static boolean isSubstring(String a, String b)
```

2. Documenta el mètode: Paràmetres, Resultat, Precondició...
3. Cas Base: qual es el problema de talla més xicoteta possible i la seua solució directa?
4. Cas Recursiu, per a el resto de casos:
- Es tracta d'un Recorregut o d'una Cerca?
  - Si es tracta d'una cerca: qual es la **condició** per a trobar allò que ens fa para? Pot ser més d'una.
  - Qual es la crida recursiva que redueix el problema i aconsegueix la precondició?
  - Que li falta al mètode per a resoldre el problema?
  - Hi ha que combinar la crida recursiva amb algun paràmetre o no?

- 
1. Dissenyar
  2. Implementar
  3. Compilar
  4. CheckStyle
  5. Provar casos

# PRÀCTICA 2: Activitat 5

- ▶ Valida el mètode `isSubstring(String)` de la classe `PRGString.java`
- Crea una classe `TestIsSubstring.java` ha d'incloure un `main` que prove tots els casos cridant a un mètode amb la capçalera: `private static void testIsSubstring(String a, String b)` que escriga en pantalla els dues `String`, el resultat obtingut al invocar al mètode i el resultat esperat (pots usar el mètode `contains(String)` dels objectes `String`).



BlueJ: Terminal Window - prg

a	b	isSubstring(a,b)	b.contains(a)
-----			
TRIA TU ELS		true	true
CASOS A		true	true
PROVAR;		false	false
DONANT-LI		false	false
VALOR ALS		true	true
STRING a i b		false	false
		true	true
		true	true
		true	true