1er EXAMEN DE PRÀCTIQUES DE PROGRAMACIÓ 27/04/2015 Duració: 1 hora

(Nota: L'examen s'avalúa sobre 10 punts, si bé el seu pes específic en la nota final de l'assignatura és de 0,8 punts)

A omplir pel professor

Pregunta 1. **(4 punts)**

Una cadena **a** és un **infix** d'un altra cadena **b** si la cadena **a** està continguda en la cadena **b** sense considerar els caràcters primer i últim de la cadena **b**. Es considera que la cadena buida és infix de qualsevol cadena i que una cadena de longitud menor o igual a 2 només té un infix: la cadena buida.

Per exemple, donades **a = "ant"**, **b = "atragantar"**, **c = "atr"**, **d = "tar"** i **e = "unt"**: la cadena **a** és infix de la cadena **b** però **b** no és infix de **a**; les cadenes **c**, **d** i **e** no són infixos de **b**.

Es demana: Implementar un mètode **recursiu**, amb el perfil mostrat en el requadre, que comprove si la cadena **a** és un infix de la cadena **b**. Aquest mètode ha d'usar obligatòriament el mètode **esPrefixe** vist en la pràctica 2 que comprova si la cadena **a** és un prefixe de la cadena **b** i el seu perfil és:

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

```
/** Comprova si la cadena a és un infix de la cadena b
 *  Precondició: a i b poden ser cadenes qualssevol
 */
public static boolean esInfix(String a, String b) {
   /* Cas base: la cadena buida és infix de qualsevol cadena */
   if (a.length() == 0) return true;
   /* Cas base: si a no és buida, la longitud de b ha de ser,
      com a mínim, 2 + la longitud d'a */
   else if (b.length()–2 < a.length()) return false;
   /* Cas general: a no és buida i a.length() <= b.length()-2 */
   else return esPrefixe(a, b.substring(1, b.length()-1)) ||
               esInfix(a, b.substring(1));
}
```

Pregunta 2. **(1 punt)**

Es demana: Respondre a la següent pregunta equivalent a la resolta durant la pràctica 1 de PRG: Quin és el mínim nombre de moviments de discos que s'han de realitzar per resoldre el problema de les Torres d'Hanoi amb 11 discos? (Justificar la resposta)

Si per a n discos es fan 2^n-1 moviments, per a 11 discos es faran 2047 moviments.

Pregunta 3. **(3 punts)**

A omplir pel professor

En el següent mètode, que mesura de manera empírica el cost en el cas promedi d'*inserció directa*, s'han comés 3 errors: 2 són per instruccions canviades de lloc i 1 és un error lògic.

```
 0 public static void mesuraInsercio() {
 1     int[] vAl;
 2     long ti, tf, tta;
 3     double tMedAl;
 4
 5      // Imprimir capçalera de resultats
 6     System.out.println("# MESURA D'ORDENACIÓ PER INSERCIÓ DIRECTA");
 7     System.out.printf("#%7s   %16s", "Talla", "T.Promedi(mseg)");
 8     System.out.println("#-----------------------");
 9     tta = 0;
10     for (int talla = TALLA_INI; talla <= TALLA_FI; talla += INCREMENT) {
11         vAl = creaArrayAleatori(talla);
12         for (int i = 0; i < REPETICIONS; i++) {
13             ti = System.nanoTime();
14             AlgorismesMesurables.insercio(vAl);
15             tf = System.nanoTime();
16             tta += tf - ti;
17         }
18         tMedAl = tta / REPETICIONS;
19         System.out.printf(Locale.US, "%8d   %16.4f\n", talla, (tMedAl/1e6));
20    }
21  }
```

Es demana: Indicar en els requadres següents per a cada error el número de línia i la seua solució.

La línia 9 hauria d'anar després de la 10.
La línia 11 hauria d'anar després de la 12.
En la línia 18 falta un casting a double o multiplicar per 1.0 alguna de les dues variables.

Pregunta 4. **(2 punts)**

Es desitja obtenir el cost empíric d'un algorisme A. Després de l'obtenció de la taula de temps d'execució, s'executa el programa *gnuplot* en què es defineix la funció **`f(x) = a*x + b`** i s'executa el comandament **`fit`** obtenint el següent resultat:

```
Final set of parameters            Asymptotic Standard Error
=======================            ==========================
a               = 2.0              +/- 1.819        (8.088%)
b               = -1               +/- 1.128e+05    (25.33%)
```

Es demana: Escriure en el requadre següent la funció d'ajust obtinguda i donar una estimació del temps que tardarà en executar-se l'algorisme A per a una talla 10.000, suposant que les dades vénen donades en la taula en mil·lisegons.

Funció d'ajust: f(x) = 2.0*x - 1

Estimació del temps d'execució per a talla 10000:
El temps requerit per a l'execució de l'algorisme A per a una talla 10000 seria f(10000) = 2.0 * 10000 - 1 = 19999 mil·lisegons.