

# Laboratori Programació I, 2020-21

## Exercicis P2: Composicions iteratives

### Grau d'Enginyeria Informàtica



En aquesta llista d'exercicis es treballen les composicions iteratives, encara que també haureu fer servir les seqüencials i alternatives.

**IMPORTANT:** Abans d'escriure el codi, heu de pensar en els diferents **tests de prova** que especifiquen com s'ha de comportar el vostre programa. Això vol dir, donada una "Entrada" quina "Sortida esperada" s'espera del vostre programa. Una vegada definits els tests de prova, escriviu el programa Java i comproveu que la "Sortida Esperada" coincideixi amb la sortida obtinguda pel vostre programa.

1. Feu un programa que sol·liciti a l'usuari fer  $n$  multiplicacions. El nombre enter,  $n$ , es sol·licita per teclat. El programa ha de generar de forma aleatòria el dos nombres a multiplicar. Si l'usuari ho fa bé el programa mostra el missatge "Ben fet!" i si ho fa malament li anima a estudiar més "Has d'estudiar la taula de multiplicar". (NMultiplicacions.java)

Per a generar un nombre aleatori entre 0 i 9:  
`int x = (int) (Math.random() * 10);`

2. Feu un programa que llegeix un nombre  $n$ , i imprimeix per pantalla tots els enters de 1 a  $n$ . (Nombres.java)
3. Feu un programa que llegeix un nombre  $n$  i escriu  $n$  asteriscs en una línia. (Asteriscs.java)
4. Feu un programa que donat un enter  $n$  escriu tres columnes, una amb els nombres de l'1 al  $n$ , la segona amb els seus quadrats i l'altra amb els seus cubs. (QuadratsCubs.java)
5. Feu un programa que llegeix dos enters positius i escriu tots els enters de l'interval tancat definit per aquests dos, en ordre creixent. Per exemple, si l'usuari introdueix els enters 4 i 14 la sortida ha de ser "14, 13, 12, 11, ..., 5, 4" (NombresInterval.java)
6. Feu un programa que llegeix un nombre enter  $n$  i una seqüència de  $n$  enters i calcula la seva mitjana. (Mitjana.java)
7. Feu un programa que llegeix un nombre enter  $n$  i una seqüència de  $n$  enters i imprimeix per pantalla el màxim. (Max.java)
8. Feu un programa que llegeix un nombre  $n$  i escriu  $n$  línies d'asteriscs tal que la línia  $i$  conté  $i$  asteriscs. (TriangleAsteriscs.java) Per exemple per  $n = 4$  hauria d'escriure:

\*

\*\*

\*\*\*

\*\*\*\*

9. Feu un programa que llegeix tres enters  $a$ ,  $b$  i  $c$  i escriu el valor que pren la funció  $f(x) = ax^2 + bx + c$  per  $x = 0, 1, 2, \dots, 20$ . (Funcio.java)
10. Feu un programa que llegeix un enter i calcula el nombre de dígits que té. Per exemple, si l'usuari introdueix l'enter 345, la sortida ha de ser "El nombre de dígits de 345 es 3".(NombreDeDigits.java)
11. Feu un programa que llegeix un nombre enter  $n$  i escriu el taulell d'escacs de mida  $n \times n$ . (Escacs.java) Per exemple, per  $n = 5$  el resultat és:

```
# # #  
# #  
# # #  
# #  
# # #
```

12. Feu un programa que llegeix un nombre enter  $n$  i escriu els primers  $n$  nombres de la successió de Fibonacci, descrita en profunditat a la pàgina web: [https://ca.wikipedia.org/wiki/Successió\\_de\\_Fibonacci](https://ca.wikipedia.org/wiki/Successió_de_Fibonacci). Per exemple, si l'usuari introdueix el nombre 10, la sortida per pantalla ha de ser "1, 1, 2, 3, 5, 8, 13, 21, 34, 55" (Fibonacci.java)
13. Feu un programa que llegeix un String i compta el nombre de vocals que hi ha en total. Per exemple, "Avui fa molta calor." dona el missatge "La frase té 8 vocals". Utilitza el mètode CharAt() per a resoldre'l. (CompteVocals.java)
14. Feu un programa que llegeix un String i mostra el codi ASCII de cadascun dels seus caràcters (Ascii.java). Per exemple amb "Hola", la sortida ha de ser:

```
H 72  
o 111  
l 108  
a 97
```

15. Feu un programa que retorni si les frases entrades per l'usuari son palíndroms o no (capicua). El programa demanarà frases mentre l'usuari no indiqui el contrari (amb la paraula "fi"). Suposa que totes les lletres de la frase s'escriuen en minúscula i que no hi ha signes de puntuació o exclamació a la frase. Per exemple: "abc cba", "sometemos" o "sosos" han de sortir com palíndroms, i "abc efg", "someter" o "soso" han de sortir com no palíndroms (Palindrom.java)

Exercicis on heu de fer ús dels esquemes de recorregut i cerca:

**En tots els problemes següents, cal que identifiqueu totes les seqüències existents i amb quin esquema algorísmic cal tractar-les (recorregut o cerca). La identificació es fa en un comentari abans de la composició iterativa corresponent.**

```
/* Identificacio de la sequencia: ...  
* Primer(): ...
```

```

* Seguent():...
* FiSeq():...
* Identificacio de l'esquema: Cerca o Recorregut
  (Quan es cerca, s'afegeix Condiçió de cerca: ...)
*/

```

16. Feu un programa per a veure si cada caràcter 'a' d'una paraula llegida per teclat està contenta. Una 'a' està contenta si té a la seva esquerra o a la seva dreta una altre 'a'. El programa ens diu si totes les 'a' estan contentes. Per exemple: si introduïm "zdaaoa" el programa mostra el missatge "No totes les a estan contentes", amb la cadena "zdaa" el programa mostra "Totes les a estan contentes" (Contentes.java).
17. Feu un programa per a jugar el joc del Nim. Dos jugadors col·loquen a un tauler un nombre de fitxes major o igual a 20 i menor estricta a 30 (aquest nombre es genera de forma aleatòria). Un jugador és l'ordinador, l'altre serà un jugador humà. Un nombre també aleatori indicarà quin jugador comença a jugar. Cada jugador en el seu torn retira una o dues fitxes. L'ordinador genera un nombre aleatori entre 1 i 2 per a retirar fitxes. El tauler s'ha d'anar actualitzant en cada torn. Guanya el jugador que aconsegueix retirar l'última fitxa. (Nim.java)

A continuació teniu un exemple de com ha de ser una partida:

```

Juguem amb 23 fitxes.

Comença l'ordinador
L'ordinador agafa 2 fitxes
Queden 21 fitxes
Quantes fitxes agafes (1 or 2)?
2
Queden 19 fitxes
L'ordinador agafa 1 fitxes
Queden 18 fitxes
Quantes fitxes agafes (1 or 2)?
2
Queden 16 fitxes
.
.
.
Queden 2 fitxes
L'ordinador agafa 1 fitxes
Queden 1 fitxes
Quantes fitxes agafes (1 or 2)?
1
Queden 0 fitxes
Has guanyat!

```

18. Feu un programa per jugar al següent joc. Un jugador serà l'ordinador i l'altre l'usuari. Cada jugador comença amb 0 punts i tira un dau. Suma el nombre de punts obtingut al total de punts. A partir d'aquesta primera tirada, cada jugador decideix successivament si tira un dau més o si es planta. Qui s'apropi més a 13 punts (sense superar-los) guanya la partida. Si algú té més de 13 punts perd automàticament. L'ordinador sempre juga primer, i demana sempre més tirades de daus sempre que la diferència entre 13 i el total de punts sigui major que 3. Quan acabi l'ordinador, és el torn de l'usuari. (TiraDaus.java) Segueix el procediment mostrat en el següent exemple:

```

Torn de l'ordinador.
0 + 6 = 6
6 + 5 = 11
L'ordinador es planta amb 11 punts.
Torn de l'usuari.
0 + 4 = 4
Tirar dau (S/N)? S

```

```
4 + 6 = 10
Tirar dau (S/N)? S
10 + 4 = 14
L'usuari es passa amb 14 punts.
Guanya l'ordinador amb 11 punts!
```

19. Feu un programa que donat un enter  $n$  i una seqüència de  $n$  estudiants especificats per nom i cognoms (String) i la nota (float), retorni les dades de l'estudiant amb millor nota. (MillorEstudiant.java)
20. Feu un programa que demani un enter  $n$ , seguidament llegeixi  $n$  enters i finalment retorni la mitjana dels nombres parells. (Mitjana.java)
21. Feu un programa que llegeixi un enter  $n$ , seguidament llegeixi  $n$  enters i finalment mostri per pantalla la mitjana, el total de nombres que hi ha per sobre de la mitjana, i cadascun dels nombres. (SobreMitjana.java)

Entrada:

10

2 8 9 5 7 3 6 4 1 10

Sortida:

La mitjana és 5.5

En total hi ha 5 nombres per sobre la mitjana

Els valors que estan per sobre la mitjana son: 8 9 7 6 10