

Estructura iterativa

Per a cada exercici proposat, teniu anotat el següent format:

(*directoryName*, *ClassName*, *methodName1*, *methodName2*,...)

Segueix les següents instruccions:

- a) Crea un directori de nom *EstructuralterativaCognomsNom*.
- b) Per a cada exercici, crea un directori amb el nom *directoryName* ubicat dins del directori *EstructuralterativaCognomsNom*.
- c) Per a cada exercici, crea una classe de nom *ClassName* dins del directori *directoryName*. El nom del fitxer on està la classe serà *ClassName.java*.
- d) Per a cada exercici, codifica dins la classe un o més mètodes amb els noms *methodName1*, *methodName2*...

1. (ascendentNumbers, Program, writeAscNumbers) Donats dos nombres enters n_1 i n_2 amb $n_1 < n_2$, escriu per pantalla tots els nombres enters dins l'interval $[n_1, n_2]$ en ordre creixent.

2. (descendentNumbers, Program, writeDescNumbers) Donats dos nombre enters n_1 i n_2 amb $n_1 < n_2$, escriu tots els nombres enters dins l'interval $[n_1, n_2]$ en ordre decreixent.

3. (squareRoots, Program, writeSquareRoots) Donats dos nombre enters n_1 i n_2 amb $n_1 < n_2$, escriu les arrels quadrades dels nombres enters dins l'interval $[n_1, n_2]$ en ordre creixent.

4. (ascii, Program, writeASCII) Escriu la seqüència ascendent dels codis i caràcters del codi ASCII estès.

5. (unicode, Program, writeUnicode) Donats dos nombres enters positius n_1 i n_2 amb $n_1 < n_2$, escriu els caràcters Unicode de dins de l'interval $[n_1, n_2]$ en ordre creixent.

6. (random, Program, writeRandomNumbers) Donats dos nombres enters n_1 i n_2 amb $n_1 < n_2$ i un nombre enter x , escriu x nombres enters aleatoris dins l'interval $[n_1, n_2]$. NOTA: Math.random() retorna un valor aleatori major o igual a 0 i menor a 1.

7. (multiples, Program, writeMultiples) Donats dos nombres enters positius n_1 i n_2 amb $n_1 < n_2$, escriu tots els nombres enters que són múltiples de n_1 més petits o iguals a n_2 en ordre creixent.

8. (evenNumbers, Program, writeEvenNumbers) Donats dos nombre enters n_1 i n_2 amb $n_1 < n_2$, escriu els nombres enters parells que hi ha dins l'interval $[n_1, n_2]$ en ordre creixent. El nombre zero es considera parell.

9. (divisors, Program, writeDivisors) Donat un nombre enter positiu, escriu els seus divisors positius.

10. (narcissisticNumbers, Program, writeNarcissisticNumbers) Escriu els nombres enters positius narcisistes de tres xifres. Els nombres narcisistes de tres xifres són de la forma:

$$abc = a^3 + b^3 + c^3$$

11. (minMax, ProgramTUI, main) Llegeix un nombre enter **n** i **n** nombres enters i escriu el nombre més gran i el més petit d'ells.

12. (numberOfEvenNumbers, Program, numberOfEvenNumbers) Donats dos nombre enters n_1 i n_2 amb $n_1 < n_2$, calcula quants parells hi ha dins l'interval $[n_1, n_2]$.

13. (sumInterval, Program, sumInterval) Donats dos nombre enters n_1 i n_2 amb $n_1 < n_2$, calcula la suma dels nombres enters que hi ha dins l'interval $[n_1, n_2]$.

14. (sumSquares, Program, sumSquares) Donats dos nombre enters n_1 i n_2 amb $n_1 < n_2$, calcula la suma dels quadrats dels nombres enters dins l'interval $[n_1, n_2]$.

15. (sumPositives, Program, sumPositives) Donats dos nombre enters n_1 i n_2 amb $n_1 < n_2$, calcula la suma dels nombres enters positius que hi ha dins l'interval $[n_1, n_2]$.

16. (sumEvenNumbers, Program, sumEvenNumbers) Donats dos nombre enters n_1 i n_2 amb $n_1 < n_2$, calcula la suma dels nombres enters parells que hi ha dins l'interval $[n_1, n_2]$.

17. (product, Program, multiply) Donats dos nombres enters positius **a** i **b**, calcula el resultat de realitzar la seva multiplicació a partir de sumes. És a dir:

$$a \cdot b = a + a + a + \dots + a \quad (\text{a sumat b vegades})$$

18. (myPow, Program, myPow) Donats el valor de la base, un nombre enter i l'exponent, un nombre enter no negatiu, calcula el valor de la potència a^b . No emprar la funció Math.pow().

19. (factorial, Program, factorial) Donat un nombre enter positiu calcula el seu factorial. El factorial d'un nombre enter positiu **n** es defineix com: $n! = 1 \times 2 \times 3 \times \dots \times (n-1) \times n$

20. (capital, Program, calculateFinalCapital) Donats el capital inicial, el % d'interès i el nombre d'anys de durada d'una inversió bancària capitalitzada segons interès compost, calcula el capital obtingut al finalitzar l'últim any. El capital acumulat des de l'inici d'un any i l'inici de l'any següent es determina per la fórmula: $C_2 = C_1 + C_1 * \text{interes} / 100$

21. (arithmeticProgression, Program, writeArithmeticProgression) Donats el primer terme, la diferència i el nombre de termes d'una progressió aritmètica, escriu els seus elements, la seva suma i el seu producte. En una progressió aritmètica cada terme és igual a l'anterior més la diferència.

22. (perfectNumber, Program, isPerfectNumber) Donat un nombre enter positiu, determina si és un nombre perfecte o no. Un nombre enter és perfecte si és positiu i és igual a la suma dels seus divisors positius, excepte ell mateix.

23. (amicableNumbers, Program, areAmicableNumbers) Donats dos nombres enters positius, determina si són amics o no. Dos nombres enters positius són amics si la suma dels divisors positius d'un és igual a l'altre número i a l'inrevés.

24. (even, Program, isEven) Donat un nombre enter positiu, determina si és parell. No emprar els operadors divisió entera ni mòdul. El nombre zero es considera parell.

25. (numberOfDigits, Program, numberOfDigits) Donat un nombre enter, calcula el nombre de xifres que té.
26. (palindromicNumber, Program, isPalindromic) Donat un nombre enter, determina si és capicua o no. Considerarem també capicues els nombres enters d'una xifra.
27. (division, Program, divide) Donats el dividend, un nombre enter positiu i el divisor, un nombre enter positiu, calcula el quocient i el residu de la divisió entera i retorna la cadena "quocient:residu". No emprar els operadors divisió entera ni mòdul.
28. (mySqrt, Program, mySqrt) Donat un nombre enter positiu, calcula la part entera de la seva arrel quadrada. No usar la funció Math.sqrt().
29. (primeFactorization, Program, primeFactorization) Donat un nombre enter positiu, escriu la seva descomposició en producte de nombres primers.
30. (occurrences, Program, occurrences) Donat un nombre enter positiu i un dígit, calcula quantes vegades el dígit apareix dins del nombre.
31. (fibonacci, Program, fibonacci) Donat un nombre enter positiu, escriu els nombres de Fibonacci (Leonard de Pisa) inferiors o iguals a ell. Els nombres de Fibonacci es defineixen de la manera següent: El primer és 0, el segon és 1, el següent és la suma dels dos anteriors i així successivament.
32. (sumDigits, Program, sumDigits) Donat un nombre enter positiu, calcula la suma de les seves xifres.
33. (evenNumbers2, ProgramTUI, main) Llegeix una seqüència de nombres no negatius acabada en zero i escriu quants nombres parells hi ha.
34. (sumEvenNumbers, ProgramTUI, main) Llegeix una seqüència de nombres enters acabada en zero i escriu la suma dels que són parells.
35. (average, ProgramTUI, main) Llegeix una seqüència de nombres no negatius acabada en zero i escriu la mitjana aritmètica dels nombres entrats. La mitjana aritmètica d'una seqüència numèrica es calcula sumant els elements de la seqüència i dividint pel nombre d'elements.
36. (minMax2, ProgramTUI, main) Llegeix una seqüència de nombres acabada en zero i escriu el més gran i el més petit de la seqüència.
37. (grades, ProgramTUI, main) Llegeix una seqüència de notes obtingudes per un grup d'alumnes acabada en -1 i escriu quants alumnes han obtingut les qualificacions MD, I, S, B, N, E. La correspondència entre notes i qualificacions és: MD=[0, 3) I=[3, 5) S=[5, 6) B=[6, 7) N=[7, 9) Ex=[9, 10]
38. (positiveNegative, ProgramTUI, main) Llegeix una seqüència de nombres enters acabada en zero i escriu si hi ha més nombres positius que negatius.
39. (isMultiple, ProgramTUI, main) Llegeix un nombre enter positiu x, un nombre enter n i una seqüència de n nombres enters i escriu si algun d'ells és múltiple de x (interrompent

la lectura de la seqüència si el trobes).

40. (consecutiveEquals, ProgramTUI, main) Llegeix una seqüència de més de dos nombres enters acabada en zero i escriu si hi ha dos nombres consecutius iguals (interrompent la lectura de la seqüència si el trobes).

41. (isArithmeticProgression, ProgramTUI, main) Llegeix una seqüència de tres o més nombres enters acabada en zero i escriu si la seqüència és una progressió aritmètica.

42. (greaterThan, ProgramTUI, main) Llegeix un nombre enter x i una seqüència de nombres enters acabada en zero i escriu si hi ha algun element més gran que x (interrompent la lectura de nombres si trobes l'element).

43. (maxDivisor, Program, maxDivisor) Donat un nombre enter positiu, calcula el seu divisor positiu més gran diferent d'ell mateix.

44. (primeNumber, Program, isPrime) Donat un nombre enter, determina si és un nombre primer o no. Un nombre enter és primer si és més gran que 1 i només és divisible per 1 i per ell mateix.

45. (squareNumber, Program, isSquareNumber) Donat un nombre enter, determina si és un quadrat perfecte o no. Un nombre enter és quadrat perfecte si existeix un altre nombre enter tal que el seu quadrat és igual a ell.

46. (gcd, Program, gcd) Donats dos nombres enters positius, calcula el seu màxim comú divisor. El MCD és el nombre enter positiu més gran que els divideix al dos.

47. (lcm, Program, lcm) Donats dos nombres enters positius, calcula el seu mínim comú múltiple. El MCM és el nombre enter positiu múltiple d'ambdós més petit.

48. (product2, ProgramTUI, main) Llegeix un nombre enter positiu n i n nombres enters i escriu el seu producte. No fer ús de l'operador $*$.

49. (primeNumbers, Program, writePrimeNumbers) Donats dos nombres enters positius n_1 i n_2 amb $n_1 < n_2$ i $1 < n_1$, escriu els nombres primers que hi ha dins l'interval $[n_1, n_2]$ en ordre creixent.

50. (perfectNumbers, Program, writePerfectNumbers) Donats dos nombres enters positius n_1 i n_2 amb $n_1 < n_2$, escriu els nombres perfectes que hi ha dins l'interval $[n_1, n_2]$ en ordre creixent.

51. Per als exercicis que no disposin de TUI, crea la classe ProgramaTUI, en la qual hi hagi el mètode que ja havíeu codificat i un nou mètode main en el qual es crearà una interfície d'usuari per terminal.