**Exercicis**

**E1.1.** Crea un programa amb un fil d’execució que mostri per consola els nombres primers entre els 100 primers enters.

1 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

**E1.2.** Crea un programa que faci el mateix que l'anterior però amb dos fils d’execució, cadascun dels quals processarà 50 nombres. Indica per consola el nom del fil que està fent el càlcul amb el nombre que ha calculat.

Thread-0: 3

Thread-0: 5

Thread-0: 7

Thread-0: 11

Thread-0: 13

Thread-0: 17

Thread-0: 19

Thread-0: 23

Thread-0: 29

Thread-1: 53

Thread-0: 31

Thread-1: 59

Thread-0: 37

Thread-1: 61

...

**E1.3.** Crea un programa que mostri els nombres enters entre l’1 i el 100 mitjançant dos fils. Cada fil imprimirà els nombres parells o imparells. Indica per consola el nom del fil que està fent el càlcul amb el nombre que ha calculat.

FilImparell: 1

FilImparell: 3

FilImparell: 5

FilImparell: 7

FilParell: 2

FilImparell: 9

FilParell: 4

FilImparell: 11

FilParell: 6

FilImparell: 13

FilParell: 8

FilImparell: 15

FilParell: 10

...

**E1.4.** Crea un programa format per dos fils d’execució. Un d'ells establirà el valor d'una variable d'un objecte Buffer (mitjançant un mètode anomenat setValue). Després mostrarà per pantalla el missatge: "Escriptor: valor".

L'altre fil llegirà aquest valor (funció getValue) i ho mostrarà per pantalla amb el format: "Lector: valor".

Cada fil s'executarà 10 vegades. A cada iteració posa a dormir 1 segon a cadascun dels fils d’execució.

Lector: 0

Escriptor: 88

Lector: 88

Escriptor: 96

Lector: 96

Escriptor: 83

Lector: 83

Escriptor: 37

Lector: 37

Escriptor: 62

Lector: 62

Escriptor: 46

Lector: 46

Escriptor: 41

Lector: 41

Escriptor: 19

Lector: 19

Escriptor: 58

Lector: 58

Escriptor: 71

**E1.5.** Crea un programa amb tres fils d'execució on cadascun llença un dau. El programa principal ha de mostrar el resultat de la suma de les tres tirades.

D3: 4

D1: 3

D2: 3

D1+D2+D3: 10

**E1.6.** Amb motiu del seu 111è aniversari en Bilbo celebra una cursa entre 3 hobbits: en Frodo (F), en Sam (S) i en Pippin (P). En Gàndalf fa de les seves i realitza un encanteri per tal d’intentar que el resultat final de la cursa sigui el següent (l’encanteri no és infal·lible):

1. Sam

2. Frodo

3. Pippin

Realitza un programa en Java que simuli la cursa. Per tal de simular la cursa, cada hobbit serà un thread que realitzarà un total de 30 mostres de missatges per la consola de la seva inicial. S’anirà informant de l’estat de la cursa per consola de la manera següent:

SFSSPSSPFPSPSFSFSSFSFSFSFSFSFSFSFSFSFSFSFSSFSFSFSFSFSFSFSFF

Arriba S

FFFPFPFP

Arriba F

PPPPPPPPPPPPPPPPPPPPPPP

Arriba P

**E1.7.** Utilitzant com a base l'exemple del model productor-consumidor, realitza un programa que mitjançant dos fils d’execució (un productor i un consumidor) es mostri per pantalla de manera sincronitzada la cadena OLE TU indefinidament.

OLE TU OLE TU OLE TU ...

**E1.8.** Modifica l'exercici E1.7 per tal de que s'imprimeixi el següent:

OLE OLE TU OLE OLE TU ...

**E1.9.** Modifica l'exercici E1.7 per tal de que s'imprimeixi el següent:

OLE TU OLE OLE TU OLE TU OLE OLE TU ...

**E1.10.** Realitza un programa que mitjançant dos fils d'execució ens mostri per consola la inscripció en Tengwar en la llengua negra de Mordor que identifica l'anell únic quan l'escalfem:

Three Rings for the Elven-kings under the sky,

Seven for the Dwarf-lords in their halls of stone,

Nine for Mortal Men doomed to die,

One for the Dark Lord on his dark throne

In the Land of Mordor where the Shadows lie.

One Ring to rule them all, One Ring to find them,

One Ring to bring them all and in the darkness bind them

In the Land of Mordor where the Shadows lie.

El programa haurà de llegir la inscripció a partir d'un fitxer. Després cadascun de dos fils llegirà i mostrarà una línia per pantalla de manera sincronitzada.

**Valoració dels exercicis**

Valoració E1 respecte al NF1: **25%**

|  |  |
| --- | --- |
| Exercici | Valoració |
| E1.1 | 5% |
| E1.2 | 10% |
| E1.3 | 10% |
| E1.4 | 10% |
| E1.5 | 10% |
| E1.6 | 15% |
| E1.7 | 10% |
| E1.8 | 5% |
| E1.9 | 5% |
| E1.10 | 20% |

**Criteris de correcció dels exercicis**

* Els exercicis que donin errors de compilació i no es puguin executar no seran avaluats.
* Funcionalitat adient a la demanda de l’enunciat de l’exercici.
* Evitar missatges per consola innecessaris.
* El tractament d'errors ha de ser adequat, de manera que si es produeixen hauria d'informar-se a l'usuari del tipus d'error produït.
* Cuidar l'estructura dels nostres programes Java (classes estructurades adequadament, mètodes, visibilitat atributs i mètodes, etc.).
* Cuidar la documentació i identació del codi font. Els arxius de codi font hauran d'incloure, en les seves primeres línies, un comentari que serveixi per resumir l'objectiu o funcionalitat del codi i el nom i cognoms de l'autor del mateix, juntament amb la data de la seva realització. També s’haurà de posar un comentari abans de cada mètode de cada classe, on s’explicarà l’objectiu principal del mateix i de les variables que recull i retorna.