

Pràctica 2: Programes concurrents amb Erlang

Programació Concurrent i en Temps Real — iTIC

Antoni Escobet Canal

20 de setembre de 2021

Índex

1 Organització	1
1.1 Lliurament	1
2 Exercicis	1

1 Organització

Aquesta sessió s'organitza com una seqüència de problemes de dificultat creixent l'objectiu dels quals és implementar petits programes escrits en Erlang. Amb l'objectiu de reforçar l'hàbit d'usar sistemes de control de versions, cal desenvolupar la pràctica amb el suport del sistema que ofereix <http://escriny.epsem.upc.edu>.

1.1 Lliurament

Cal lliurar els diferents fitxers font dels exercicis a través d'Atenea en la data fixada. Cal que el desenvolupament es faci usant Subversion a través de les facilitats que ofereix <http://escriny.epsem.upc.edu>.

2 Exercicis

EXERCICI 2.1

Generador i verificació del codi de control del ISBN-13. El format ISBN-13 està format per 12 dígits (entre el 0 i el 9) més un dígit de comprovació. Poden haver-hi guionets i es pot comprovar la seva validesa mitjançant la següent fórmula:

$$(x_1 + x_2 * 3 + x_3 + x_4 * 3 + x_5 + x_6 * 3 + x_7 + x_8 * 3 + x_9 + x_{10} * 3 + x_{11} + x_{12} * 3 + x_{13}) \text{mod} 10 == 0$$

Si el resultat val 0, llavors és un ISBN-13 correcte, en cas contrari no és vàlid.

a) Dissenyeu una funció que donat un codi de 12 nombres calculi el dígit de control. Per exemple:

```
1> ex2:generalISBN("978-84-8181-227").
2> 978-84-8181-227-5
```

El càlcul que fa, és:

$$(9 + 7*3 + 8 + 8*3 + 4 + 8*3 + 1 + 8*3 + 1 + 2*3 + 2 + 7*3 + X) \text{mod} 10 == 0 \quad X = 5$$

b) Dissenyeu una funció `validISBN(Isbn)` que retorni cert si és correcte i fals en cas contrari. Un exemple pot ser:

```
1> ex2:validISBN("978-84-8181-227-5").
2> true.
```

El càlcul que fa, és:

$$(9 + 7*3 + 8 + 8*3 + 4 + 8*3 + 1 + 8*3 + 1 + 2*3 + 2 + 7*3 + 5) \bmod 10 == 0$$

Podeu fer la verificació de la vostre funció amb el següent conjunt de proves:

```
1> ex1:validISBN("978-84-8181-227-5"). true
2> ex1:validISBN("978-84-8181-227-6"). false
3> ex1:validISBN("978-84-8181-227-X"). true
4> ex1:validISBN("979-84-8181-227-A"). false
5> ex1:validISBN("9788481812275"). true
6> ex1:validISBN("00"). false
7> ex2:validISBN("3-598-21515-X"). false
8> ex2:validISBN(""). false
9> ex2:validISBN("14-7"). true (analitza 000000000014-7)
```

EXERCICI 2.2 L'objectiu d'aquest exercici és implementar un mòdul 'pila' que implementi una pila. Les operacions han de ser les clàssiques d'una pila: `buida()`, `empila()`, `desempila()` i `cim()`. Per representar una pila useu una llista.

```
P = pila:buida().
[]
P1 = pila:empila(P,0).
[0]
P2 = pila:empila(p1,{a,b}).
[{a,b},0]
pila:desempila(P2).
[0]
pila:cim(P2).
{a,b}
```

Comproveu amb l'interpret d'`erlang` que la pila funciona correctament.

EXERCICI 2.3 Dissenyeu i implementeu un programa `Erlang` que llegeix una expressió en notació postfixa i en calcula el valor. Per simplificar assumiu que només s'admeten enters i les operacions sumar, restar, multiplicar i dividir. Així, per exemple, si el programa s'executa invocant la funció `start()`, es podria executar així:

```
1> start().
expressio> 12 4 + 2 /.
8
```

La operació que realitza, és: $(12+4)/2 = 8$

Aquest calcul es fa usant una pila. Així doncs haureu d'usar el mòdul `pila` de l'apartat anterior. Pel que fa a com llegir l'expressió, consulteu l'ús de la funció `io:scan_erl_exprs()` en el manual d'`Erlang`.



EXERCICI 2.4 Modifiqueu el programa anterior fent que sigui capaç d'acceptar nombres reals.

```
1> start().
```

```
expressio> 12.5 4 + 2.2 /.
```

```
7.5
```

La operació que realitza, és: $(12.5+4)/2.2 = 7.5$