Laboratorio di Programmazione a.a. 2011-12

Problema 11

18 Gennaio 2012

Descrizione – Parte I

Per *"Cifrario di Cesare"* oggi si intende una regola crittografica di sostituzione che "ruota" ciascuna lettera del testo in chiaro di un numero prefissato di posizioni nell'ordine alfabetico. L'entità della rotazione, detta anche *"chiave"* di decodifica, è concordata fra il mittente e il destinatario del messaggio. Stando ai resoconti di Svetonio, Giulio Cesare avrebbe utilizzato questo rudimentale sistema di crittazione con chiave 3. Per esempio, tenuto conto che le lettere dell'alfabeto Latino dell'epoca erano: A B C D E F G H I L M N O P Q R S T V X , La frase

ALEA IACTA EST IVLIVS CAESAR DIXIT

sarebbe stata trasformata nel messaggio cifrato

DOHD NDFAD HXA NBONBX FDHXDV GNCNA

Data una regola crittografica di tipo *"Cifrario di Cesare"*, la cui chiave non è conosciuta, rappresentata da una *procedura* in Scheme, si vuole determinare la regola di decrittazione, anche questa in forma di *procedura*. Le regole di crittazione e decrittazione si riferiscono all'alfabeto Latino dell'epoca Repubblicana.

Definisci una procedura con argomenti e valori procedurali che, data una regola del cifrario di Cesare, restituisce la corrispondente regola di decrittazione, ricostruita individuando la chiave appropriata. (La procedura che così si ottiene è più efficiente di quella generale discussa a lezione, che si applica a qualsiasi regola di permutazione delle lettere.)

Descrizione – Parte II

Le operazioni basilari di addizione (*add*), moltiplicazione (*mul*), elevamento a potenza (*pow*) nel dominio dei numeri naturali possono essere definite una dall'altra in modo induttivo nel seguente modo:

$add(m, 0) = m$
$add(m, n) = succ(add(m,n-1))$ *per* $n > 0$

$mul(m, 0) = 0$
$mul(m, n) = add(m, mul(m,n-1))$ *per* $n > 0$

$pow(m, 0) = 1$
$pow(m, n) = mul(m, pow(m,n-1))$ *per* $n > 0$

Se si sostituisce la funzione di un argomento $succ(v) = v+1$ c on la funzione di due argomenti $s2(u,v) = v+1$, allora tutte queste definizioni hanno una struttura comune:

$h(m, 0) = f(m)$
$h(m, n) = g(m, h(m,n-1))$ *per* $n > 0$

Tale struttura è caratterizzata dall'operatore funzionale H, tale che $h = H(f, g)$. In particolare si ha:

$add = H(i, s2)$

$mul = H(z, add)$

$pow = H(u, mul)$

dove i, z, u sono rispettivamente la funzione *identità* (che assume il valore dell'argomento), la funzione costante *zero* (che assume valore 0 per ogni argomento), la funzione costante *uno* (che assume valore 1 per ogni argomento).

H può essere modellata da una procedura `H` con argomenti e valore procedurali.

Definisci `H` in Scheme, quindi definisci le procedure corrispondenti a *add*, *mul* e *pow* applicando `H`, per esempio:

```
(define  mul  (H  (lambda (x) 0)  add))
```