8.9 (P488) Write a CLIPS program that will add two binary numbers without using any arithmetic functions. Represent the binary numbers using the following deftemplate:

```
(deftemplate binary-#
   (multislot name)
   (multislot digits))
```

Given a fact indicating which two named binary numbers are to be added, the program should create a new named binary number containing the sum. For example, the facts:

```
(binary-# (name A) (digits 1 0 1 1 1))
(binary-# (name B) (digits 1 1 1 0))
(add-binary-#s (name-1 A) (name-2 B))
```

should cause the following fact to be added to the fact list:

```
(binary-# (name { A + B }) (digits 1 0 0 1 0 1))
```

輸出入:

```
Please input the amount of binary numbers to be added: -5
Please input a positive integer!!
Please input the amount of binary numbers to be added: 4
Please input binary number #1: 1 1 a 0 0 1
Input error!! Please input binary numbers!!
Please input binary number #1: 1 1 0 0 1
Please input binary number #2: 1 0 1 1
Please input binary number #3: 1 0 0 1 1 1
Please input binary number #4: 1 1 1 0 1
{ { { 4 + 3 } + 2 } + 1 }: 1 1 0 1 0 0 0
```

小考7

```
(deftemplate binary-# (multislot name) (multislot digits))

(deftemplate binary-adder (multislot name-1) (multislot name-2) (slot carry) (multislot #-1) (multislot #-2) (multislot result))

(deffacts initial-fact (phase input-amount) (number 0))

(defrule input-amount
 (phase input-amount)
 =>
 (printout t "Please input the amount of binary numbers to be added: ")
 (assert (amount (read))))

(defrule input-binary-#
 (phase input-binary-#)
 (amount ?a)
 ?f <- (number ?n)
 (test (< ?n ?a))
 =>
 (retract ?f)
 (printout t "Please input binary number #" (+ ?n 1) ": ")
 (assert (binary-# (name (+ ?n 1)) (digits (explodeS (readline)))))
 (assert (number (+ ?n 1))))

(defrule create-adder
 (phase add-binary-#)
 ?f1 <- (binary-# (name S?n1) (digits S?d1))
 ?f2 <- (binary-# (name S?n2&~$?n1) (digits $?d2))
 =>
 (retract ?f1 ?f2)
 (assert (binary-adder (name-1 ?n1) (name-2 ?n2) (carry 0) (#-1 ?d1) (#-2 ?d2) (result))))
```

```
(defrule adder-case-1
 ?f <- (binary-adder (carry ?c) (#-1 S?n1 ?d1) (#-2 $?n2 ?d2) (result $?r))
 =>
 (modify ?f (carry (integer (/ (+ ?c ?d1 ?d2) 2)))
            (result (mod (+ ?c ?d1 ?d2) 2) ?r) (#-1 ?n1) (#-2 ?n2)))
```

```
(defrule convert-adder-to-number
 ?f1 <- (binary-adder (name-1 S?n1) (name-2 S?n2) (carry 0) (#-1) (#-2) (result S?r))
 =>
 (retract ?f1)
 (assert (binary-# (name { ?n1 + ?n2 }) (digits ?r))))
```

提示: (explodeS ?s) 可以將一個字串欄位轉成為multifield
(implode$ $?m) 可以將一個multifield欄位轉成為字串
(integerp ?n) 可以測試 ?n 的值是否為整數型態
(integer ?f) 可以取得浮點數 ?f 的整數值(例如15.89 ⇒ 5)