

```

;;; -*- Syntax: Common-Lisp; Package: USER; Base: 10; Mode: LISP -*-
;;;
;;; チャート型性格診断 テストプログラム
;;; Masayuki Ida, Aoyama Gakuin Univ. 1988.10.30
;;;

```

```

;;; Questions

```

知識部

Yes NO

```

(setq *1*
  '((princ "いちばん良く見るテレビ番組はお笑い番組である。 ") *2* *5*))
(setq *2*
  '((princ "生まれ変わるとしたら、もう一度男(女)がいい。 ") *5* *3*))
(setq *3*
  '((princ "今、クラスに好きな人がいる。 ") *4* *6*))
(setq *4*
  '((princ "実際の年齢よりも、年上にみられることが多い。 ") *7* *8*))
(setq *5*
  '((princ "動物が大好きで、飼育係を一度やってみたい。 ") *9* *6*))
(setq *6*
  '((princ "流行には敏感だ。服のセンスをほめられたこともある。 ") *10* *7*))
(setq *7*
  '((princ "声がでかい。 ") *11* *8*))
(setq *8*
  '((princ "手先が器用で、結構まめだ。簡単な料理は作れる。 ") *12* *13*))
(setq *9*
  '((throw 'deguchi "君はとっても優しいね。ちょっと甘えん坊のところもあるけれど。 ") ))
(setq *10*
  '((throw 'deguchi "君はルックスにこだわる。 )))
(setq *11*
  '((throw 'deguchi "人前で注目を浴びるのが好き。 )))
(setq *12*
  '((throw 'deguchi "好きなことには、ご飯を忘れても熱中する方。 )))
(setq *13*
  '((throw 'deguchi "君はクラス一のひょうきん者")))
;; 小四チャレンジ 11月号 福武書店 進研ゼミ を参考にして。

```

```

;;; diagnosis driver

```

推論部/実行部

```

(defun test ()
  (terpri)
  (princ "さあ、チャートテストを始めよう。 ")
  (terpri)
  (princ "質問ができるから、yesならy, noならnをいれてね。 ")
  (terpri)
  (catch 'deguchi
    (let ((exp *1*))
      (while t
        (eval (first exp))
        (princ "-->")
        (if (eq (get-reply) 'y)
            (setq exp (eval (second exp)))
            (setq exp (eval (third exp))))
        )))

```

```

(defun first (x) (car x))
(defun second (x) (car (cdr x)))
(defun third (x) (car (cdr (cdr x))))
(defun get-reply () (progn (princ (read)) (terpri)))

```

1単語読み, 1体を表示,
terpriし, 読んだ値を返す.

exp
現在対象を
与えている
を指す