```
  (PROGN (LISPXPRIN1 (QUOTE "FILE CREATED ")
                     T)
         (LISPXPRIN1 (QUOTE "13-JUN-72  4:20:07")
                     T)
         (LISPXTERPRI T))
(DEFINEQ

(DOCTOR
  [LAMBDA (FLG)
    (COND
      ((NULL FLG)
       (SETQ MEMSTACK NIL)))
    (PROG (KEYSTACK SENTENCE TIMON)
          (SETSEPR (QUOTE (%  %
 %
  )))
          (SETBRK (QUOTE (%. , ? ! - %( %) ; : #)))
          (CONTROL T)
          (SETQ FLIPFLOP 0)
          (SETQ TIMON (CLOCK))
          (RECONSTRUCT (QUOTE (TELL ME YOUR PROBLEMS, PLEASE TERMINATE
                                    INPUT WITH A PERIOD OR A QUESTION
                                                           MARK
                                %.))
                       T)
          (SETNONE)
      A   (PRIN1 (QUOTE "
*"))
          (COND
            ((NULL (SETQ SENTENCE (MAKESENTENCE)))
             (GO A)))
          (SETQ KEYSTACK (CDR SENTENCE))
          (SETQ SENTENCE (CAR SENTENCE))
          [COND
            ((EQUAL SENTENCE (QUOTE (GOODBYE)))

          (* Rate computed at $6 a minute, or $1 per 10
          seconds, CLOCK returns value in milliseconds.)
```

```
            (RETURN
              (RECONSTRUCT
                [APPEND (QUOTE (IT'S BEEN MY PLEASURE, THAT'S))
                        (CONS (PACK (LIST (QUOTE ^S)
                                          (QUOTIENT (SETQ TIMON
                                                      (IDIFFERENCE
                                                        (CLOCK)
                                                        TIMON))
                                                    10000)
                                          (QUOTE %,)
                                          (QUOTIENT (REMAINDER TIMON
                                                               10000)
                                                    100)))
                              (QUOTE (PLEASE %.]
                        T]
          (ANALYZE)
          (GO A])

(MAKESENTENCE
  [LAMBDA NIL
    (PROG (FLAG WORD SENTENCE KEYSTACK)
      A1  (SETQ KEYSTACK (CONS))
          (SETQ SENTENCE (CONS))
      A   (SETQ WORD (RATOM))
          [COND
            ((NUMBERP WORD)
             (SETQ WORD (PACK (LIST (QUOTE *)
                                    WORD]
          [COND
            ((EQ WORD RUBOUT)
             (RETURN (TERPRI)))
            ((MEMB WORD TRMLIS)
             (TERPRI)
             (RETURN (RPLACD SENTENCE KEYSTACK)))
            ((MEMB WORD PCTLIS)
             (COND
               ((NULL (CDR KEYSTACK))
                (GO A1))
               ((NULL (SETQ FLAG (MAKESENTENCE)))
                (RETURN))
               ([AND (CDDR FLAG)
                     (NOT (IGREATERP (GET (CDR KEYSTACK)
                                          (QUOTE PRIORITY))
                                     (GET (CDDR FLAG)
                                          (QUOTE PRIORITY]
                (RETURN FLAG))
               (T (RETURN (RPLACD SENTENCE KEYSTACK]
          (TCONC SENTENCE (COND
                            ((GETP WORD (QUOTE TRANSLATION)))
                            (WORD)))
```

```
            [COND
              ((SETQ FLAG (GETP WORD (QUOTE MEMR)))
               (SETQ MEMSTACK (APPEND FLAG MEMSTACK]
            (COND
              [[AND (SETQ FLAG (GETP WORD (QUOTE PRIORITY)))
                    (CDR KEYSTACK)
                    (IGREATERP FLAG (GET (CDR KEYSTACK)
                                         (QUOTE PRIORITY]
               (RPLACD KEYSTACK (CONS (CDR KEYSTACK)
                                      (CDR WORD]
              (FLAG (BCONC (CDR WORD)
                           KEYSTACK)))
            (GO A])

(ANALYZE
  [LAMBDA NIL
    (PROG (RULES PARSELIST CR)
          (BCONC [GETP (QUOTE NONE)
                       (COND
                         ([ZEROP (SETQ FLIPFLOP (IPLUS 2 (IMINUS
                                                           FLIPFLOP]
                          (QUOTE MEM))
                         ((QUOTE LASTRESORT]
                 KEYSTACK)
          (SETQ KEYSTACK (CDR KEYSTACK))
      A   (SETQ RULES (GET KEYSTACK (QUOTE RULES)))
      B   (COND
            ((OR (NULL RULES)
                 (EQ (CAR RULES)
                     (QUOTE NEWKEY)))
             (SETQ KEYSTACK (CAR KEYSTACK))
             (GO A))
            ((NLISTP (CAR RULES))
             (SETQ RULES (GETP (CAR RULES)
                               (QUOTE RULES)))
             (GO B)))
          (SETQ PARSELIST (CONS NIL NIL))
          (COND
            ((NOT (TEST (CAAR RULES)
                        SENTENCE))
             (SETQ RULES (CDR RULES)))
            [(NLISTP (SETQ CR (CAR (SETQ RULES (CAR (ADVANCE RULES]
            ((EQ (CAR CR)
                 (QUOTE PRE))
             (SETQ SENTENCE (RECONSTRUCT (CADR CR)))
             (SETQ RULES (CDDR CR)))
            (T (RECONSTRUCT CR T)
               (MEMORY)
               (RETURN)))
          (GO B])
```

```
(TEST
  [LAMBDA (D S)
    (PROG (CD PSV)
          (SETQ PSV (CDR PARSELIST))
      LP  (COND
            [(NULL D)
             (COND
               (S (GO RN))
               (T (SETQ PARSELIST (CAR PARSELIST))
                  (RETURN T]
            ((EQ Ø (SETQ CD (CAR D)))
             (GO TØ))
            ((NULL S)
             (GO RN))
            [(NUMBERP CD)
             (TCONC PARSELIST S)
             (COND
               ((SETQ S (NTH S CD))
                (GO T3))
               (T (GO RN]
            [(NLISTP CD)
             (COND
               ((EQ CD (CAR S)))
               (T (GO RN]
            [(CAR CD)
             (COND
               ((MEMBER (CAR S)
                        CD))
               (T (GO RN]
            ((TEST4 (CAR S)
                    (CDR CD)))
            (T (GO RN)))
          (TCONC PARSELIST S)
      T3  (SETQ S (CDR S))
          (SETQ D (CDR D))
          (GO LP)
      TØ  (TCONC PARSELIST S)
          (COND
            ((NULL (SETQ D (CDR D)))
             (SETQ PARSELIST (CAR PARSELIST))
             (RETURN T)))
      T1  (COND
            ((TEST D S)
             (RETURN T))
            ((SETQ S (CDR S))
             (GO T1)))
      RN  [RPLACD PARSELIST (COND
                    (PSV (RPLACD PSV NIL]
          (RETURN NIL])
```

```
(TEST4
  [LAMBDA (CS L)
    (PROG NIL
      LP  (COND
            ((GETP CS (CAR L))
             (RETURN T))
            ((SETQ L (CDR L))
             (GO LP)))
          (RETURN NIL])

(ADVANCE
  [LAMBDA (RULES)
    (RPLACA (CDAR RULES)
            (COND
              ((NULL (CDADAR RULES))
               (CDDAR RULES))
              ((CDADAR RULES])

(RECONSTRUCT
  [LAMBDA (RULE PF)
    (PROG (SENT CR V1 V2 TPF QMF)
          [COND
            ((NULL PF)
             (SETQ SENT (CONS]
      LP  (COND
            ((NULL RULE)
             (COND
               (PF [COND
                     ((NULL QMF)
                      (PRIN1 (QUOTE ?]
                   (TERPRI)))
               (RETURN (CAR SENT)))
            ((NUMBERP (SETQ CR (CAR RULE)))
             (GO T1))
            [PF (COND
                  ((MEMBER CR TRMLIS)
                   (PRIN1 CR)
                   (SETQ QMF T))
                  (T (COND
                       (TPF (SPACES 1))
                       (T (TERPRI)
                          (SETQ TPF T)))
                     (PRIN1 CR]
            (T (TCONC SENT CR)))
      T3  (SETQ RULE (CDR RULE))
          (GO LP)
      T1  [SETQ V1 (CAR (SETQ CR (NTH PARSELIST CR]
          (SETQ V2 (CADR CR))
```

```
      T2  [COND
            ((EQ V1 V2)
             (GO T3))
            (PF (COND
                  (TPF (SPACES 1))
                  (T (TERPRI)
                     (SETQ TPF T)))
                (PRIN1 (CAR V1)))
            (T (TCONC SENT (CAR V1]
          (SETQ V1 (CDR V1))
          (GO T2])

(MEMORY
  [LAMBDA NIL
    (PROG (PARSELIST X)
      LP  (COND
            ((NULL MEMSTACK)
             (RETURN)))
          (SETQ PARSELIST (CONS NIL NIL))
          [COND
            ((TEST (CAAR MEMSTACK)
                   SENTENCE)
             (RPLACA [SETQ X (CDAADR (GETP (QUOTE NONE)
                                           (QUOTE MEM]
                     (CONS (CAR X)
                           (CONS (RECONSTRUCT (CAAR (ADVANCE MEMSTACK)))
                                 (CDAR X]
          (SETQ MEMSTACK (CDR MEMSTACK))
          (GO LP])

(BCONC
  [LAMBDA (WHAT LIST)
    (COND
      ((NULL LIST)
       (CONS (SETQ LIST (CONS NIL WHAT))
             LIST))
      [(NULL (CAR LIST))
       (RPLACA LIST (CDR (RPLACD LIST (CONS NIL WHAT]
      ((RPLACA LIST (CAR (RPLACA (CAR LIST)
                                 (RPLACA (CONS LIST WHAT)
                                         NIL])

(RPLQQ
  [NLAMBDA RPLQ
    (RPLACD (CAR RPLQ)
            (CDR RPLQ])
```

```
(SETNONE
  [LAMBDA NIL
    (PROG (A)
          (SETQ A (GENSYM))
          (RPLACD A (GETP (QUOTE NONE)
                          (QUOTE LASTRESORT)))
          (PUT (QUOTE NONE)
               (QUOTE MEM)
               (LIST (QUOTE RULES)
                     (LIST (LIST (LIST Ø)
                                 (LIST NIL)
                                 A])
)
  (LISPXPRINT (QUOTE DOCTORFNS)
              T)
  (RPAQQ DOCTORFNS
         (DOCTOR MAKESENTENCE ANALYZE TEST TEST4 ADVANCE RECONSTRUCT
                 MEMORY BCONC RPLQQ SETNONE))
  (LISPXPRINT (QUOTE DOCTORVARS)
              T)
  [RPAQQ DOCTORVARS (TRMLIS PCTLIS RUBOUT DOCARM
                            (P (ADVISE (QUOTE INTERRUPT)
                                       DOCARM)
                               (GCTRP 100)
                               (GCGAG]
  (RPAQQ TRMLIS (%. ! ?))
  (RPAQQ PCTLIS (, ; %( %) :))
  (RPAQQ RUBOUT #)
  [RPAQQ DOCARM (COND
           ((EQ INTYPE 3)
            (PRIN1 (QUOTE "

...EXCUSE ME FOR JUST A MINUTE.
")
                   T)
            (RECLAIM)
            (COND
              ((STKPOS (QUOTE MAKESENTENCE))
               (PRIN1 (QUOTE
              "SORRY TO HAVE INTERRUPTED YOU, PLEASE CONTINUE...

")
                      T))
              (T (PRIN1 (QUOTE "NOW, WHERE WERE WE...OH YES,
")
                        T)))
            (SETQ INTYPE -1]
  (ADVISE (QUOTE INTERRUPT)
          DOCARM)
  (GCTRP 100)
  (GCGAG)
STOP
```