CS -79

360 O. S. FORTRAN IV
FREE FIELD INPUT/OUTPUT SUBROUTINE PACKAGE

by

R. W. Doran

This research was supported in part by the
United States Air Force Electronic Systems
Division under Contract F19628-C-0035.

STANFORD UNIVERSITY COMPUTER SCIENCE DEPARTMENT
COMPUTATIONAL LINGUISTICS PROJECT

OCTOBER 1967

Introduction

Programmers dealing with aspects of natural language processing have a difficult task in choosing a computer language which enables them to program easily, produce efficient code and accept as data freely written sentences with words of arbitrary length. List processing languages such as LISP are reasonably easy to program in but do not execute very quickly. Other, formula oriented, languages like FORTRAN are not provided with free field input.

The Computational Linguistics group at Stanford University Computer Science Department is writing a system for testing transformational grammars. As these grammars are generally large and complicated it is important to make the system as efficient as possible, so we are using FORTRAN IV (O.S. on IBM 360-65) as our language. To enable us to handle free field input we have developed a subroutine package which we describe here in the hope that it will be useful to others embarking on natural language tasks.

The package consists of two main programs, free field reader, free field writer, with a number of utility routines and constant COMMON blocks.

Free Field Reader (FREAD)

FREAD was written to enable us to read free field input including data containing longwords, special characters, and integers and to eventually control the system by the form of its input with the detection of orders.

FREAD is a REAL*8 function with one dummy argument (INTEGER*2). When it is called it returns the next entity it finds in the input stream. It communicates information concerning that entity through labelled common viz.

```
CØMMØN /ØRDCM/ NUM, ISPEC, NUMFL, ØRDFL
INTEGER*2 NUM, ISPEC
LØGICAL*1 NUMFL, ØRDFL
```

The 'input stream' refers to the sequence of characters from the punched card input, with column 72 of a given card assumed to be followed hard by column 1 of the next. Double quotes " are omitted from the input stream along with any characters between consecutive double quotes. This gives us the facility of allowing comments anywhere in the input.

The different entities are as follows:

1/. Special character.

This is taken to be any character whose EBCDIC 8 bit code has a value of less than 128 (except for blank, ", $).

2/. Integer.

A sequence of digits whose value is less than 2^{15}. An integer may be followed immediately by a word.

3/. Word.

A sequence of alphameric characters (i.e., those with 8 bit EBCDIC codes of value greater than 128) of length less than or equal to 40 (40 is arbitrarily set as 'being of arbitrary length') and not beginning with a **$**.

4/. Order

A word of length less than 8 and beginning with a **$**.

As FREAD encounters each entity it returns a value and sets conditions as follows:

1/. Special character.

FREAD is set equal to the character followed by 7 blanks. ISPEC is set to the value of the character's 8 bit code (see appendix 1). NUMFL and ØRDFL are .FALSE.

2/. Integer.

FREAD is set to the value of the integer. If the integer has a too-large value a system overflow interrupt will occur. ISPEC is zero, NUMFL is .TRUE., ØRDFL is .FALSE. (Here it would make good sense to have ISPEC carry the value of the integer. Our method is left over from the 7090 where FREAD was itself an integer f-unction, a-user could easily change this.)

3/. Word.

If the word is cf length less than or equal to 8 characters it is itself returned as FREAD (with trailing blanks), if it is of greater length a <u>keyword</u> is returned. ISPEC is zero, NUMFL and ØRDFL are .FALSE.

The <u>keyword</u> of a longword is obtained by replacing the tail end of the first 8 characters by an integer of value according as to the

position of this longword in the sequence of those encountered. For example, if PRØNØMINALIZATION is the 15th longword encountered it will always have the keyword PRØNØM15.

If a word of more than 40 characters is encountered FREAD prints an error message and truncates the word.

4/. Order

The order is returned as a word. ISPEC is zero, NUMFL is .FALSE., ØRDFL is .TRUE.

If an order has more than 8 characters an error message is written and the keyword returned (this is because keywords can alter as the input alters, but orders should be definite if they are to be of any use).

Example.

Suppose in a given run FREAD encounters the following input

$START(R=12NP),PRØNOM"CØMMENT"INALIZE,$$N$,N1$1$M67N,622N1.

it will behave in the order of the following table.

Entity	FREAD	ISPEC	ØRDFL	NUMFL
$START	$START	0	T	F
(	(	77	F	F
R	R	0	F	F
=	=	126	F	F
12	value 12	0	F	T
NP	NF	0	F	F
)		93	F	F
,	,	107	F	F
PRONØMINALIZE	PRØNØMI1	0	F	F
,	,	107	F	F

Entity	FREAD	ISPEC	ØRDFL	NUMFL
$$N$	$$N$	0	T	F
,	,	107	F	F
N1$1$M67N	N1$1$M62	0	F	F
,	,	107	F	F
622	value 622	0	F	T
N1	N	0	F	F
.	.	75	F	F

Further operational points

1/. As illustrated in the previous example N1$1$M67N returns N1$1$M62 which is an ambiguous keyword. Such words should be avoided as they will cause trouble on output.

2/. FREAD must be initialized by calling the subprogram INITLZ. This initializes all the tables of longwords so more than one independent run may be made in the same job.

3/. FREAD performs a readout, printing each card image as it is read. NUM in /ØRDCM/ is used if one wishes to number the input, If an integer is placed in NUM it will be written out in the left hand margin of the next card read out,

4/. A listing of the longwords and their keys may be obtained by calling LNGØUT.

5/. When interpreting the input from FREAD it is handy to have constants representing the 8 bit values of the various characters, For this purpose a common block /CNSTCM/ is provided (see the example programs in appendix 2 for a listing). /CNSTCM/ is initialized in a block data subprogram which is part of the package.

6/. Lower case letters may be used if the system you are using allows you to print them.

7/. The present capacity for longwords is between 50 and 250. (Most likely around 200.) This capacity can be readily increased if desired.

Free field Writer (FROUT)

FROUT is a subroutine with which one can write out data in a free field format with keywords which have been read by FREAD expanded if desired. When designing a free field writer it is difficult to allow enough flexibility to obtain an aesthetically arranged printout, The options we have included allow one to obtain a reasonable-looking result in most cases.

FROUT gets it data from a linear sump KSUMP in labelled common /MAINCM/ viz.

```
CØMMØN /MAINCM/ CHRTR, KSUMP, ISUMP, NCHRTR
REAL*8 CHRTR, KSUMP(2000)
INTEGER*2 ISUMP, NCHRTR
```

FROUT has 7 arguments; the first, ISTART, is INTEGER*2, the remainder B1---B6 are LØGICAL*1.

When FROUT is called it writes out the contents of KSUMP, from ISTART+1 up to ISUMP, in a compacted free field form with the following options:

B1:	.TRUE.	Print and punch
	.FALSE.	Print only
B2:	.TRUE.	Expand keywords
	.FALSE.	Do not expand keywords
B3:	.TRUE.	Start a new line on a period
	.FALSE.	Don't
B4:	.TRUE.	Start a new line on a comma
	.FALSE.	Don't
B5:	.TRUE.	Start anew line on an =
	.FALSE.	Don't

B6: .TRUE. Sequence number the output lines
.FALSE. Don't

Further Operational Points

1/. To obtain greater generality of use we assume that no subprogram which places data in KSUMP initializes ISUMP. Care should be taken in remembering to initialize ISUMP in the main control routine,

2/. The keywords of longwords encountered by FREAD may be placed in the sump by calling the subroutine KEYSM . This enables you to obtain a punched listing of the longwords in order, which when placed in front of input-to a later job will ensure that the longwords receive the same keywords as before.

3/. For convenience when filling KSUMP a common block /FCSTCM/ is provided which contains the special, symbols in REAL*8 form. For a listing of /FCSTCM/ see the example programs in appendix 2. The names of the constants are the same as for /CNSTCM/ (see appendix 1) with the initial N replaced by F . /FCSTCM/ is initialized by a block data subprogram which is part of the package.

4/. The output of FROUT is compacted, but if blanks are desired they may be inserted by placing FBLANK into KSUMP.

5/. /MAINCM/ includes two constants, PAGE and RECØRD. When FRØUT encounters PAGE in KSUMP it, skips to the head of a new page when printing (new card when punching); when it encounters RECØRD it skips to the start of a new line.

6/. FROUT should never fail on its input unless it is asked to output an integer of value greater than 2^{15}-1, in which case a system overflow interrupt will occur; If FROUT is given a bad keyword it

will write an error message.

7/. CHRTR & NCHRTR in /MAINCM/ are used for communicating with subprograms that use FREAD, e.g. returning an order to your control program. CHRTR is used to hold the current value of FREAD, NCHRTR for ISPEC if needed.

Appendix 1

Table of special characters in /CNSTCM/ and their codes

Name in /CNSTCM/	Character	Punch code	Hexadecimal code	Decimal value
NBLANK	blank		40	64
NCENT	¢	12-8-2	4A	74
NSTØP	.	12-8-3	4B	75
NLESS	<	12-8-4	4C	76
NLEFTP	(	12-8-5	4D	77
NPLUS	+	12-8-6	4E	78
NLØR	|	12-8-7	4F	79
NLAND	&	12	50	80
NXCLM	!	11-8-2	5A	90
NDOLLR	$	11-8-3	5B	91
NSTAR	*	11-8-4	5C	92
NRITEP	)	11-8-5	5D	93
NSCØLN	;	11-8-6	5E	94
NLNØT	¬	11-8-7	5F	98
NMINUS		11	60	96
NSLASH	/	0-1	61	97
N1211		12-11	6A	106
NCØMMA	,	0-8-3	6B	107
NPERC	%	0-8-4	6C	108
NLINE	_	0-8-5	6D	109
NGREAT	>	0-8-6	6E	110
NQUERY	?	0-8-7	6F	111
NCØLØN	:	8-2	7A	122
NBØUND	#	8-3	7B	123
NAT	@	8-4	7C	124
NQUØTE	'	8-5	7D	125
NEQUAL	=	8-6	7E	126
NDQUØT	"	8-7	7F	127

Appendix 2

This section contains two examples of programs using our free field package. The first illustrates how subprograms using FREAD and FROUT are controlled, using our routines for reading, expanding, and storing a phrase structure grammar (PSGINN) and then writing it out in an expanded form (PSGSMP). The second shows how to construct a program which itself uses FREAD.

```
LEVEL 2 FE3 67                          OS/360 FORTRAN H                          DATE  67.265/01.34.15

          COMPILER OPTIONS - NAME=  MAIN,OPT=00,LINECNT=57,SOURCE,EBCDIC,NOLIST,DECK,LOAD,NOMAP,NOEDIT,NOID

                C
                C   THISFIRST TEST PROGRAM SKIPS TO A NEWPAGE,READS IN A PHRASE
                C  STRUCTURE GRAMMAR,EXPANDS IT,STORES IT IN KSUMP,OUTPUTS THE
                C  EXPANDED GRAMMAR IN TWO DIFFERENT FORMS,AND F I N A L L Y PRINTS OUT
                C A REFERENCE LIST OF THE LONG WORDS IT ENCOUNTERED
                C
    ISN 0002            COMMON /MAINCM/ CHRTR,KSUMP,ISUMP,NCHRTR                          MAINCM
    ISN 0003            REAL*8 CHRTR,KSUMP(2000)                                          MAINCM
    ISN 0004            INTEGER*2 ISUMP,NCHRTR                                            MAINCM
    ISN 0005            COMMON /FCSTCM/                                                   FCSTCM
                       1 FBLANK,FLAND ,FMINUS,FSLASH,FCENT ,FSTOP ,FLESS ,FLEFTP,FPLUS ,  FCSTCM
                       2 FLOR  ,FXCLM ,FDOLLR,FSTAR ,FRITEP,FSCOLN,FLNOT ,F1211 ,FCOMMA,  FCSTCM
                       3  FPERC ,FLINE ,FGREAT,FQUERY,FCOLON,FBOUND,FAT   ,FQUOTE,FEQUAL,  FCSTCM
                       4 FDQUOT, PAGE  ,RECORD                                            FCSTCM
    ISN 0006            REAL*8                                                            FCSTCM
                       1 FBLANK,FLAND ,FMINUS,FSLASH,FCENT ,FSTOP ,FLESS ,FLEFTP,FPLUS ,  FCSTCM
                       2 FLOR  ,FXCLM ,FDOLLR,FSTAR ,FRITEP,FSCOLN,FLNOT ,F1211 ,FCOMMA,  FCSTCM
                       3 FPERC ,FLINE ,FGREAT,FQUERY,FCOLON,FBOUND,FAT   ,FQUOTE,FEQUAL,  FCSTCM
                       4 FDQUOT, PAGE  ,RECORD                                            FCSTCM
    ISN 0007            LOGICAL*1 T/.TRUE./,F/.FALSE./
    ISN 0008            ISTART  =  0
    ISN 0009            ISUMP = 1
    ISN 0010            KSUMP(ISUMP) = PAGE
    ISN 0011            CALL INITLZ
    ISN 0012            CALL FROUT(ISTART,F,T,T,T,T,T)
    ISN 0013            CALL PSGINN
    ISN 0014            CALL PSGSMP
    ISN 0015            CALL FROUT(ISTART,F,T,T,T,T,F)
    ISN 0016            CALL FROUT(ISTART,F,F,F,F,F,T)
    ISN 0017            CALL PSGSMP
    ISN 0018            CALL LNGOUT
    ISN 0019            RETURN
    ISN 0020            END

****** END OF COMPILATION ******

12
```

```
"AF TEST PS-RULES   OLASOPE OYELARAN, A U G . 2 2, 1967 ."

SS = # S #.

S = NP V P .

VP = (PRE)  V  (((NP) (PP) (AGNT), SS, AP) (ADV)).

v = A U X(VB(ADV), C U P).

AUX = ((DO, (H A V E EN)(BEING)))AUXA.

AUXA= (MOD)(PRES, P A S T )(ASP).

ADV = (SS, ADVB (SS), PP).

AP = ((PRE) ADJ (SS), SS).

PP = PRE NP.

NP = (NP S S, (D)N N U ,SS).

D = (PRE) (ART(ADJ) (SS), (D) ADJ).

PRE = (NEG) (PRT).

ART = (WH)(INDEF, O F F , .

V = LOVE, PROCRASTINATE, TEMPORIZE, ALTERNATE, LOSE.

ADVB = QUICKLY, MAGNIFICENTLY, FANTASTICALLY, SORROWFULLY, SADLY.

CCP = BE.

ADJ = STUPENDOUS, BIG, GIANT, TURQUOISE, CATASTROPHIC.

N = ELEPHANT, HIPPOPOTAMUS, GNU, UNICORN, AARDVARK, RURU, NIGHTHAWK.

INDEF = A .

DEF = THE .

$END
```

```
SS =
#S#.
S =
NP VP.
VP =
V SS ADV,
V SS,
V PP AGNT ADV,
V PP AGNT,
V PP ADV,
V PP,
V NP PP AGNT ADV,
V NP PP AGNT,
V NP PP ADV,
V NP PP,
V NP AGNT ADV,
V NP AGNT,
V NP ADV,
V NP,
V AP ADV,
V AP,
V AGNT ADV,
V AGNT,
V ADV,
V,
PRE V SS ADV,
PRE V SS,
PRE V PP AGNT ADV,
PRE V PP AGNT,
PRE V PP ADV,
PRE V PP,
PRE V NP PP AGNT ADV,
PRE V NP PP AGNT,
PRE V NP PP ADV,
PRE V NP PP,
PRE V NP AGNT ADV,
PRE V NP AGNT,
PRE V NP ADV,
PRE V NP,
PRE V AP ADV,
PRE V AP,
PRE V AGNT ADV,
PRE V AGNT,
PRE V ADV,
PRE V.
V =
AUX VB ADV,
AUX VB,
AUX COP.
AUX =
HAVE EN BE ING AUXA,
HAVE EN AUXA,
DO AUXA,
BE ING AUXA,
AUXA.
AUXA =
PRES ASP,
PRES,
PAST ASP,
PAST,
```

```
MOD PRES ASP,
MOD PRES,
MOD PAST ASP,
MOD PAST.
ADV =
SS,
PP,
ADVB SS,
ADVB.
AP =
SS,
PRE ADJ SS,
PRE ADJ,
ADJ SS,
ADJ.
PP =
PRE NP.
NP =
SS,
NP  SS,
N NU,
D N NU.
D  =
PRE D ADJ,
PRE A R T SS,
PRE A R T ADJ SS,
PRE ART ADJ,
PRE ART,
PRE ADJ,
D ADJ,
ART SS,
ART ADJ SS,
ART ADJ,
ART,
ADJ.
PRE =
PRT,
NEG PRT,
NEG.
ART =
WH INDEF,
WH DEF,
INDEF,
DEF.
V =
TEMPORIZE,
PROCRASTINATE,
LOVE,
LOSE,
ALTERNATE.
ADVB =
SORROWFULLY ,
SADLY,
QUICKLY,
MAGNIFICENTLY,
FANTASTICALLY.
COP =
HE.
ADJ =
TURQUOISE,
STUPENDOUS,
```

```
GIANT,
CATASTROPHIC,
BIG.
N =
UNICORN,
RURU,
NIGHTHAWK,
HIPPOPOTAMUS,
GNU,
ELEPHANT,
AARDVARK.
INDEF =
A.
DEF =
THE.
```

```
1       SS =#S#.S =NP VP.VP =V SS ADV,V SS,V PP AGNT ADV,V PP AGNT,V PP ADV,V PP,V NP PP AGNT ADV,V NP PP AGNT,V NP PP ADV,V NP PP
2       ,V NP AGNT ADV,V NP AGNT,V NP ADV,V NP,V AP ADV,V AP,V AGNT ADV,V AGNT,V ADV,V,PRE V SS ADV,PRE V SS,PRE V PP AGNT ADV,PRE
3        V PP AGNT,PRE V PP ADV,PRE V PP,PRE V NP PP AGNT ADV,PRE V NP PP AGNT,PRE V NP PP ADV,PRE V NP PP,PRE V NP AGNT ADV,PRE V
4        NP AGNT,PRE V NP ADV,PRE V NP,PRE V AP ADV,PRE V AP,PRE V AGNT ADV,PRE V AGNT,PRE V ADV,PRE V.V =AUX VB ADV,AUX VB,AUX CO
5       P.AUX =HAVE EN BE ING AUXA,HAVE EN AUXA,DO AUXA,BE ING AUXA,AUXA.AUXA =PRES ASP,PRES,PAST ASP,PAST,MOD PRES ASP,MOD PRES,M
6       OD PAST ASP,MOD PAST.ADV =SS,PP,ADVB SS,ADVB.AP =SS,PRE ADJ SS,PRE ADJ,ADJ SS,ADJ.PP =PRE NP.NP =SS,NP SS,N NU,D N NU.D =P
7       RE D ADJ,PRE ART SS,PRE ART ADJ SS,PRE ART ADJ,PRE ART,PRE ADJ,D ADJ,ART SS,ART ADJ SS,ART ADJ,ART,ADJ.PRE =PRT,NEG PRT,NE
8       G.ART =WH INDEF,WH DEF,INDEF,DEF.V =TEMPORI2,PROCRAS1,LOVE,LOSE,ALTERNA3.ADVB =SORROWFO,SADLY,QUICKLY,MAGNIFI4,FANTAST5.CO
9       P =BE.ADJ =FURQUOI8,STUPEND7,GIANT,CATASTR9,BIG.N =UNICORN,RURU,NIGHTHI1,HIPPOP10,GNU,ELEPHANT,AARDVARK.INDEF =A.DEF =THE.
```

TABLE OF LONG WORDS

KEYWORD	EXPANSION
PROCRAS1	PROCRASTINATE
TEMPORI2	TEMPORIZE
ALTERNA3	ALTERNATE
MAGNIFI4	MAGNIFICENTLY
FANTAST5	FANTASTICALLY
SORROWF6	SORROWFULLY
STUPEND7	STUPENDOUS
TURQUOI8	TURQUOISE
CATASTR9	CATASTROPHIC
HIPPOP10	HIPPOPOTAMUS
NIGHTH11	NIGHTHAWK

```
COMPILER OPTIONS - NAME= MAIN,OPT=00,LINECNT=57,SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,NOID

                C
                C   THIS TEST PROGRAM READS IN A LIST OF WORDS,NUMBERING THE INPUT,
                C AND THEN WRITES OUT THE WORDS,ONE PER LINE,WITH SEQUENCE NUMBERS.
                C
ISN 0002              IMPLICIT INTEGER*2(A-Z)
ISN 0003              COMMON /MAINCM/ CHRTR,KSUMP, ISUMP,NCHRTR                          MAINCM
ISN 0004              REAL*8 CHRTR,KSUMP(2000)                                           MAINCM
ISN 0005              INTEGER*2 ISUMP,NCHRTR                                             MAINCM
ISN 0036              COMMON /ORDCM/ NUM, ISPEC,ORDFL ,NUMFL                             ORDCM
ISN 0007              INTEGER*2 NUM, ISPEC                                               ORDCM
ISN 0009              LOGICAL*1 ORDFL,NUMFL                                              ORDCM
ISN 0009              COMMON /FCSTCM/                                                    FCSTCM
                     1 FBLANK, FLAND ,FMINUS,FSLASH,FCENT ,FSTOP ,FLESS ,FLEFTP,FPLUS  ,  FCSTCM
                     2 FLOR  ,FXCLM ,FDOLLR,FSTAR ,FRITEP,FSCOLN,FLNOT ,F1211 ,FCOMMA,    FCSTCM
                     3  FPERC ,FLINE ,FGREAT,FQUERY,FCOLON,FBOUND,FAT   ,FQUOTE,FEQUAL,   FCSTCM
                     4 FDQUOT, PAGE  ,RECORD                                            FCSTCM
ISN 0010              RE AL*13                                                           FC STCY
                     1  FBLANK, FLAND ,FMINUS,FSLASH,FCENT ,FSTOP ,FLESS , FLEFTP,FPLUS ,  FCSTCM
                     2 FLOR  ,FXCLM ,FDOLLR,FSTAR ,FRITEP,FSCOLN,FLNOT ,F1211 ,FCOMMA,    FCSTCM
                     3  FPERC ,FLINE ,FGREAT,FQUERY,FCOLON,FBOUND,FAT   ,FQUOTE,FEQUAL,   FCSTCM
                     4 FDQUOT, PAGE  ,RECORD                                            FCSTCM
ISN 0011              COMMON/CNSTCM/ NBLANK,NLAND ,NMINUS,NSLASH,NCENT ,NSTOP ,NLESS ,   CNSTCY
                     1  NLEFTP, NPLUS ,NLOR  ,NXCLM ,NDOLLR,NSTAR ,NRITEP,NSCOLN,NLNOT  ,  CNSTCM
                     2  N1211 ,NCOMMA,NPERC ,NLINE ,NGREAT,NQUERY,NCOLON,NBOUND,NAT   ,   CNSTCM
                     3 NQUOTE,NEQUAL,NDQUOT                                             CNSTCM
ISN 3312              INTEGER*2       NBLANK,NLAND ,NMINUS,NSLASH,NCENT ,NSTOP ,NLESS  ,  CNSTCM
                     1  NLEFTP, NPLUS ,NLOR  ,NXCLM ,NDOLLR,NSTAR ,NRITEP,NSCOLN,NLNOT  ,  CNSTCM
                     2 N1211 ,NCOMMA,NPERC ,NLINE ,NGREAT,NQUERY ,NCOLON,NBOUND,NAT   ,   CNSTCM
                     3 NQUOTE,NEQUAL,NDQUOT                                             CNSTCM
ISN 0013              LOGICAL*1 T/.TRUE./,F/.FALSE./
ISN 0014              REAL*8 FREAD
ISN 0015              CALL IYITLZ
ISN 0016              ISTART = 0
ISN 0017              ISUMP = 1
ISN 0018              KSUMP(ISUMP) = PAGE
ISN 0019              CALL FROUT(ISTART,F,F,F,F,F,F)
ISN 0020              J =1
ISN 0021              NUM = J
ISN 0022          100 CHRTR = FREAD(DUMMY)
ISN GO23              IF(ORDFL) GO TO 200
ISN 002s              ISUMP = ISUMP + 1
ISN 0026              KSUMP(ISUMP) = CHRTR
ISN 0027              IF(ISPEC.NE.NSTOP) GO TO 100
ISN 0029              J = J + 1
ISN 0030              NUM = J
ISN 0031              ISUMP = ISUMP + 1
ISN 0032              SUMP(1 SUMP) = RECORD
ISN 0033              GO TO 100
ISN 0034          200 CALL FROUT(ISTART,F,T,T,F,F,T)
ISN 003s              RETURN
ISN 0036              END
```

```
 1  ANTIDISESTABLISHMENTARIANISM.
 2  BIOGRAPHY.
 3  CORPULENT .
 4  DIDACTIC.
 5  ELEMENTARY.
 6  FLOCCINAUCINIHILIPILIFILIPICATION .
 7  GRAND . HORRIBLE. INCREDIBLE. JUXTAPOSED.
11  KANGAROO. LAMPOON.MINUET.NEPOTISM.
15  ORANGATANG. PETTY.QUESTIONABLE.RHETORIC.
19  STAMINA.
20  TAMATATAWHAKATANGIHANGIKOAUAUTAMATEA .
21  UNDULATION. VIBRAPHONE.WISHYWASHY.
24  XANADU. YPKES. LEN.                    $
```

1 ANTIDISESTABLISHMENTARIANISM.

2 BIOGRAPHY.

3 CORPULENT.

4 DIDACTIC.

5 ELEMENTARY.

6 FLOCCINAUCINIHILIPILIFILIPICATION.

7 GRAND.

8 HORRIBLE.

9 INCREDIBLE.

10 JUXTAPOSED.

11 KANGAROO.

12 LAMPOON.

13 MINJET.

14 NEPOTISM.

15 ORANGATANG.

16 PETTY.

17 QUESTIONABLE.

18 RHETORIC.

19 STAMINA.

20 TAMATATAWHAKATANGIHANGIKOAUAUTAMATEA.

21 UNDULATION.

22 VIBRAPHONE.

23 WISHYWASHY.

24 XANADU.

25 YPRES.

26 ZEN.