```
TO SAVE.CLEAN :FILENAME
ERASE [SAVE.CLEAN SETITEM DIFF ITEM LISTED MAP REDUCE QUOTED FILTER.ONE FILTER ;
]
SAVE :FILENAME
END


TO SETITEM :INDEX :LIST :VALUE
IF :INDEX > COUNT :LIST [OP LPUT :VALUE :LIST]
IF EQUALP :INDEX COUNT :LIST [OP SE BL :LIST :VALUE]
OP SE SETITEM :INDEX BL :LIST :VALUE LAST :LIST
END


TO DIFF :A :B
OP :A - :B
END


TO ITEM :INDEX :LIST
IF EMPTYP :LIST [OP []]
IF EQUALP :INDEX 1 [OP FIRST :LIST]
OP ITEM DIFF :INDEX 1 BF :LIST
END


TO LISTED :THING
IF LISTP :THING [OP :THING]
OP FPUT :THING []
END


TO MAP :TEMPLATE :LIST
IF EMPTYP :LIST [OP []]
OP FPUT ( RUN LPUT QUOTED FIRST :LIST LISTED :TEMPLATE ) ( MAP :TEMPLATE BF :LIS
T )
END


TO REDUCE :TEMPLATE :LIST
IF EMPTYP BF :LIST [OP :FIRST :LIST]
IF EMPTYP BF BF :LIST [OP RUN SE :TEMPLATE LIST ( QUOTED FIRST :LIST ) ( QUOTED
FIRST BF :LIST )]
OP RUN SE :TEMPLATE LIST ( QUOTED FIRST :LIST ) ( QUOTED REDUCE :TEMPLATE BF :LI
ST )
END


TO QUOTED :THING
IF LISTP :THING [OP :THING]
OP WORD "" :THING
END


TO FILTER.ONE :TEMPLATE :ARG
IF ( RUN SE :TEMPLATE QUOTED :ARG ) [OP :ARG] [OP []]
END


TO FILTER :TEMPLATE :LIST
IF EMPTYP :LIST [OP []]
IF NOT EMPTYP FILTER.ONE :TEMPLATE FIRST :LIST [OP FPUT ( FILTER.ONE :TEMPLATE F
IRST :LIST ) ( FILTER :TEMPLATE BF :LIST )] [OP FILTER :TEMPLATE BF :LIST]
END


TO ; :LIST
END
```