RESEQUENCE PROGRAM

Prepared For

Computer Usage Development Corporation

October 26, 1967

Programmer - Loren E. Kelley

1. FUNCTION

This program is designed to resequence source program decks written in COBOL, Assembly language or Autocoder.

2. FEATURES

For all three types of programs, the user has the option of incrementing the sequence numbers of the output deck by 1's, 10's or 100's. Assembly language program users may request one through nine blank cards to be inserted before tags in the program. COBOL language users may request one to nine blank cards to be inserted before paragraphs in the COBOL program. (No blanks are inserted before the data division of the COBOL program is reached) the blanks thus introduced are punched with the desired identification sequence as well as the appropriate sequence number. No provision is made for inserting blanks into Autocoder programs.

A listing is made of all programs processed, which lists all cards as resequenced and output along with the matching old sequence numbers.

3. PROGRAM CONTROL

It must be noted that job control cards must be removed from the source decks prior to processing. The reason for this is that the program treats these cards as though they were part of the source program and consequently punches sequence numbers and identification sequences into the reproduced job control cards, thus rendering the, at best, ineffective.

Program control for the resequence program is achieved through the use of special control cards which must be placed as the first card of each deck to be processed. Only columns

1 through 10 and 73 through 80 are used, and in the following manner:

Columns 1 through 6 of all control cards contain "******". These six asterisks merely identify the card as a control card.

Column 7 is used to identify the type of deck to be resequenced "A" for Assembly language, "C" for COBOL, and "D" for Autocoder.

Column 8 is used to indicate whether or not blank cards are requested; "Y" = yes, "N" = no.

Column 9 contains the number of blank cards desired (0 through 9).

Column 10 is used to indicate the desired increment interval of the new sequence numbers; "I" = by ones, "T" = by tens, "H" = by hundreds.

Columns 73 and 74 are used by Assembly language program users as an identification indicator for the source program. These two columns may contain either letters or numbers of alternately may be left blank.

Columns 73 through 80 are to be used by COBOL program users to indiciate the identification sequence to be supplied to the COBOL source program if left blank there is no identification sequence supplied to the duplicated deck.

Sample Control Cards

```
Column:   1 2 3 4 5 6 7 8 9 10                    73 74      80
Contents: * * * * * * A N 0 I                      ID
          * * * * * * A N 0 T                      ID
          * * * * * * A N 0 H                      ID
          * * * * * * A Y 1 I                      ID
          * * * * * * A Y 1 T                      ID

          etc.                                     7  8
                                                   3  0
          * * * * * * C N 0 I                      123456.7
          * * * * * * C N 0 T                      12345678
          * * * * * * C N 0 H                      1.234567
          * * * * * * C Y 2 T                      MNOP2401

          etc.

          * * * * * * D N 0 I
          * * * * * * D N 0 T
          * * * * * * D N 0 H
```

4. SPECIAL OPERATING INSTRUCTIONS

A. I/O

I/O devices required by this program are one printer, an operator's console and one card read-punch.

B. Error Routines

The program is set up to handle two error conditions, 1) missing control card, 2) missing deck type indicator (card column 7).

1. Limitations: If two programs are to be processed back to back and the control card for the second

deck is omitted, both decks will be treated as one by the program.

2. Error One: When the program finds that the first card of the first deck to be resequenced is not a control card, a message is printed on the operator's console "first card is not a control card, supply proper card if available, then reply--proceed". The machine then enters a wait state until the operator replies proceed. The program then returns to check again for a control card. If there is still no control card, the error routine is repeated.

3. Error 2: If the deck type column is either blank or not "A", "C", or "O", a message is printed on the operator's console "deck type is not indicated on control card. Supply correct control card if available, the reply--proceed". The machine then enters a wait state, as above, until the operator replies. The program then returns to check again for a control card.

Since it is clear that these error provisions provide a minimum of protection for the user's programs, it is essential that care be taken in preparation of control cards.