Z77-5349

TECHNICAL INFORMATION EXCHANGE

SORTING ALGORITHMS AND OPERATING PROCEDURES FOR CARD SORTING ON THE IBM/360 MODEL 20 WITH MFCM 2560

Mr. W. N. Holmes
IBM Australia Pty. Limited
Box 88, P. O. St. Kilda
Victoria, Australia

IBM Corporation, Technical Publications Dept., 112 E. Post Road, White Plains, N.Y. 10601

IBM®

September 14, 1965

This paper describes, compares, and evaluates four different card sorting algorithms, giving extensive examples, tabulations and timing charts.

This paper also presents certain systems aspects and operating procedures which make system/360 card sorting more attractive than conventional card sorting.

CONTENTS

	Page No.
Abstract	1
Introduction	2
Straight Merge	4
Retrospective Merge	5
Compressive Merge	7
Long Input Strings	10
Synchronisation	12
Variable Algorithms	14
Reduction in Sorting Time	16
Continuous Operation	18
Reversed Files	20
Comparison with Unit Record Sorting	22

Appendices

Index	24
Section A - Merge Examples	25
Section B - Logic Diagrams	39
Section C - Test Results	42
Section D - Graphical Summary	46
Section E - Test Program	49

TITLE: Sorting Algorithms & Operating Procedures for Card Sorting on the IBM/360/20 with MFCM 2560.

AUTHOR: W.N. Holmes

DATE: December 30, 1964

DIRECT ENQUIRIES TO:
W.N. Holmes,
IBM Australia Pty. Limited,
P.O. Box 88,
St. Kilda, Victoria, Australia.

ABSTRACT: This paper describes, compares and evaluates four different card sorting algorithms, giving extensive examples, tabulations and timing charts.

This paper also presents certain systems, aspects and operating procedures which make System /360 card sorting more attractive than conventional card sorting.

INTRODUCTION

This paper was inspired by several intra-office discussions on the utility of the 360/20/2560 for card sorting and the efficiency of the supporting program.

The investigations carried out to support this paper were hastily executed and the results included in the appendices must not be regarded as anything but flimsy evidence, based as they are on small samples called in an informal manner from the middle of the Smiths in the Melbourne Telephone Directory.

The algorithms investigated were merely the first few which sprang to mind and many more remain even unmentioned. In particular, digit sorting similar to conventional card sorting can be performed with the MFCM, but this has been ignored.

The casual reader will be mainly interested in the practical conclusions drawn in the section entitled "Comparison with Unit Record Sorting".

The paper will be most easily followed if the appendices are separated from the body of the paper for easy reference while reading the narrative.

Relevant IBM Publications are:-

C26-3601	IBM System/360 Model 20 Punched Card Utility Programs
F28-8001	General Information Manual Sorting Methods for IBM DP Systems

STRAIGHT MERGE

The concept of merging to create a file with a single sequence is quite distinct from that of digit sorting as with an 082 card sorter. It is assumed that the reader is familiar with both concepts.

The straight merge simply feeds two halves of an input file (one half from each hopper) and merges, string for string, then flip/flop stacks output, string by string, into two stackers. The process is exemplified in Appendix A1, with a logic diagram in Appendix B1.

Basically, the gain G of the process will be 2.0, the gain being defined so ---

$$G = \frac{\overset{\text{input}}{\text{No. of output strings}}}{\underset{\text{output}}{\text{No. of input strings}}}$$

However, with small input strings, the gain may rise slightly above 2.0. This adventitious process is illustrated in Appendix A2.

Using a straight merge, 2N input strings may be merged into a single sequence with N passes of the file.

RETROSPECTIVE MERGE

To improve the gain available with straight merging, the circumstances of adventitious gain (as shown by Appendix A2) may be amplified. The value of the control field of the last card selected into EACH stacker is relevant. This information is available to the controlling program and it would be a pity not to use it.

A "retrospective" algorithm might be loosely stated as follows. Feed the lower or only "possible" card into the highest "possible" stacker. If neither card is "possible", feed the lower card into the highest stacker.

By "possible" card is meant a card that can be stacked without causing sequence break. Also, a "possible" stacker will not break sequence if the card is selected into it. The ranking of stackers is by control field of the last stacked card. The algorithm as stated is for ascending sequence sorting.

The retrospective merge is exemplified in Appendix A3, with a logic diagram in Appendix B2. The example of Appendix A3 may be compared with that of App. A1. The retrospective merge is compared with other merges in Appendix A4.

The appendices referred to above show input strings of average length 2. The process whereby a gain larger than 2 is achieved is more clearly illustrated in Appendix A6 which illustrates a retrospective merge with input string length of about 5.

One characteristic of the algorithm for retrospective merging is that its scope is not limited to a mere two stackers. The merge illustrated in Appendix A5 shows the merge of Appendix A3 but using three stackers. The tabulated test results of Appendix C4 give an indication of the dependence of gain on the number of stackers used.

COMPRESSIVE MERGE

In seeking to improve the retrospective merge, its operation must be examined in detail. Consider the merge illustrated in Appendix A3. Notice that card 731 in the primary feed is held (waiting for the 500 and 583 cards to be added to sequence in stacker 2) despite its close proximity to the 709 card in stacker 1. This card would have been held up even if it had been a 710 card. The 731 card is, in fact, followed by cards 504 and 558 which might advantageously be merged with cards 500 and 583 in the secondary feed.

The retrospective merge is designed to feed as many cards as possible without causing a sequence break. Taking a more indirect approach, it might be expected that a merge designed to minimise the gap in current stacker sequences, regardless of sequence breaks, would achieve a higher gain, at least with short input strings.

Hence, an algorithm for "compressive" merging might be stated as follows. Calculate the "gap" for each feed in combination with each stacker. Select the feed/stacker combination which gives the smallest gap and feed that card into that stacker.

The gap across a sequence break is calculated according to the modulus of the control field. For example, with a positive three digit numerical field, the gap between 999 in the stacker and 001 in the feed is 002. With alphabetic fields the gap might be calculated by depth of match of the two fields, or by binary subtraction, or some such subterfuge. With /360 internal coding, the use of binary subtraction would tend to emphasise the sequence break, but not as utterly as retrospective merging. In fact, some experimenting with such a binary subtraction with all fields might yield some interesting effects, giving, as it would, a blend of retrospective and compressive merging.

The compressive merge is exemplified in Appendix A7, with a logic diagram in Appendix B3. Note that the = exit from the gap comparison was not implemented in the program used to generate the test results of this paper. The example of Appendix A7 may be compared with those of Appendices A1 and A3. Also, the tabulated test results of Appendix C2 show that the expected higher gain is definitely attained with short input strings.

The compressive, like the retrospective merge, is not limited to use with two stackers. Appendix A8 (cf. A5) shows the compressive merge using three stackers, and Appendix C4 tabulates test results showing the dependence of gain on the number of stackers used.

The appendices referred to above show input strings of average length 2. The process whereby a gain larger than 2 is achieved is more clearly illustrated in Appendix A9 which shows a compressive merge with input string length of about 5. Comparison of Appendices A9 and A6 clearly displays the essential difference in operation between retrospective and compressive merging.

The results described so far show that sophisticated merges give gain significantly higher than a straight merge for short input strings. However, with long input strings, the compressive and retrospective merges degenerate to operation very similar to the straight merge. This effect will be seen in Appendix A10, which shows a retrospective merge with input string length of about 8. The similarity to a straight merge is immediately evident. The compressive merge tends to give similar results although cards may be fed in a slightly different sequence or large gaps in input sequence may hold up one feed for a time, giving very poor gain locally.

The extra gain achieved by the retrospective and compressive merges comes from the switching of the output from one stacker to another. At the point of switching , the control field in the stacker currently being fed approaches that in another stacker, and the gap in sequence allows the high stacker to take over the output, thereby reducing the gap which would occur by straight merging. In effect, each stacker has a switching point which (when using two stackers) steadily backs down the sequence until it crosses a sequence break, thus causing one less than the straight number of strings in one stacker.

Hence, what might be called local gain approximately equal to one half the average gap will be obtained at each stacker switch. This gain should actually be greater than half the average gap since stacker switching might be expected to occur preferentially at the larger gaps. In commercial sorting applications cards usually cluster, and this will allow greater gain than a random distribution. In the extreme, groups of identical cards will reduce the effective string length since they will be fed and stacked as though they were a single card.

The use of more than two stackers will increase the gain possible since each stacker will have a switching point so that each output string will tend to spread itself over all available stackers. Appendices A11 and A12 illustrate this effect.

SYNCHRONISATION

Consideration of Appendices A11 and A12 reveals weaknesses in the operation of both retrospective and compressive when the input comprises long strings. The effect is that a blocking condition may occur to prevent one feed from contributing to the output until the block is removed at a significantly later stage. The effect can be so pronounced that merging of the input strings does not actually take place for some time and the controlling program only works one feed.

Appendix A11 shows that a high card, in conjunction with one or more low stackers, is a blocking condition for the retrospective merge. Appendix A12 shows that a card fed such that a large gap in sequence separates it from every stacker constitutes a blocking condition for the compressive merge.

The blocking effect may be prevented by forcing the input to merge, string for string, using the knowledge of the previously stacked cards to control the stacking of subsequent cards but not the feeding of the cards. The method might be thought of as synchronisation of the primary and secondary feeds.

Benefit from synchronisation would not be expected unless the average input string length were greater than the number of stackers used. Notice that application of this technique removes the distinction between retrospective and compressive merging, emphasising that the relative success of compressive merging at low string length is due to its better control over the cards fed, not from more efficient stacking.

Synchronisation was applied to the retrospective merge section of the program which produced the test results of the appendices to this paper. The results of synchronisation may be seen in Appendices A4, A13, C2, C3, C4, D1 and D2. In particular, the comparison in Appendix A4 between the synchronised retrospective merge and the straight merge demonstrates their similarity, at least on the input side.

The synchronisation described above is not the only method of removing the blocking effect, but it has the virtue of simplicity.

VARIABLE ALGORITHMS

The algorithms and variations already described are of varying effectiveness depending on such factors as input string length, stackers used and range and type of control field. This variability suggests that increased overall effectiveness could be obtained by keeping a measure of certain factors such as input string length (in each feed, perhaps) and switch from one type of merge to another when conditions appear to be favourable. For example, a weighted moving average string length or control field gap size might be kept (and revised for each card read to enable a switch to be made part way through a long string) or a range (perhaps weighted moving average again) for each digit or character of a field and/or for each field as an entity.

Appendix D3 gives an estimate of the performance a variable algorithm might achieve, compared with the performance of simpler merges. It must be emphasised that this appendix is based on rather insubstantial test results and extensive guesswork.

VARIABLE ALGORITHMS (continued)

Notice that retrospective and compressive merges may be interchanged at any point. If synchronism of the feeds is to commence or cease, certain points of interchange may be optimum. The straight merge would not be directly interchangeable unless only two stackers were in use throughout. However, when input strings become very long so there is no point in avoiding the straight merge, the extra stackers may be dropped one by one at sequence breaks, until a straight merge may be commenced.

The algorithms described in the preceding text are suggested as definite means to significantly reduce sorting time with the 360/20/2560. However, these are not the only means.

If a card file is being sorted into sequence solely so that it can be tabulated, then the final merge pass will best be written as part of the tabulating program to save one pass of the file. In fact, any such tabulating program should be written for the general case of two input files (including the special case, as say, an end of file run-out condition) which are incidentally merged.

Often a card file must be sorted to a series of related sequences (ringing the changes) to enable a series of related reports to be produced. With unit record sorting, the order in which the sorts are performed can be designed to minimise the number of card passes through the sorter by maximising the pre-sequencing. This can also be done with the 360/20 sorting, but the entire control field must be specified at every stage to force preservation of the pre-sequencing. However, the proportionate saving will probably not be as great.

One incidental aspect of MFCM sorting is that mishandling of the cards will not invalidate the sort and indeed, if it occurs during an early pass, will probably go unnoticed.

Another aspect of MFCM sorting which promotes efficient operation is that the front of the file is not a significant point. In other words, the operator merely takes the cards from the front of the stackers and puts them in the hoppers (perhaps via card racks if the file is large) until they all feed into only one stacker. The file is then in sequence. There is no pause between passes to adjust the device. The aspect of continuous operation is examined more closely in the next section.

CONTINUOUS OPERATION

It might be thought that, with a straight merge, the procedure of taking cards out of stacker 1 (2) and putting them into hopper 1 (2) would ensure trouble-free continuous operation. This is not necessarily the case. For the first pass of the sort the operator will arbitrarily split the input file into two halves, one for each hopper. Since it would be very difficult to split the file initially into an equal number of strings, the merge would eventually fall into imbalance, the cards in one stacker/hopper loop building up, in the other disappearing. A little hand-merging of strings with pencil and paper will uncover this phenomenon. Hence the operator should eventually exercise control over the transfer of strings from stacker to hopper.

Also, when a more sophisticated merge is used, the problem of feeding two hoppers from say five stackers arises. For small string lengths, selection of large slabs from each stacker in turn will only introduce a few extra strings. However, when relatively few strings are left (if not earlier) the operator must exercise control over the transfer of strings from stacker to hopper.

With a straight merge (either in its own right or as the last phase of a complex merge) the merge should split the strings between pairs of stackers. Within each pair of stackers, the feed would be switched at the first sequence break after, say, 500 cards. This would enable the operator to balance the hoppers without splitting any strings.

The use of marker cards (physically distinctive cards punched all 9's) is also recommended. These cards should be inserted at intervals throughout the initial input deck. They would indicate to the operator both the precise location of some sequence breaks and, indirectly, the progress of the sort and would be removed from the back of the deck on completion of the sort. In effect, the cards in any stacker between two marker cards would constitute a macro-string and the operator could replenish the lower hopper with the front or only macro-string in one of the stackers.

REVERSED FILES

Some reference is made in the IBM reference manual C26-3601-1 to the problem of reversing the sequence of a file from say descending to ascending order. Under any type of merge a strict reversal will be a lengthy procedure. However, if the file is predominantly in a sequence the reverse of that required for a report, a better shorter procedure is possible.

Sort, if necessary, the file to the strict reverse of the sequence required. Then, for the report, feed the file from the back face up nine edge first into the read hopper. This will supply input cards to the reporting program in the correct sequence but the card record will be inverted with columns 1 to 80 presented as columns 80 to 1. It is a relatively simple loop to invert the card using two registers for addressing, one incrementing, the other decrementing. The program requires only two instructions, on the other hand, if the translate special feature is installed. The coding used can be as follows.

First, some areas should be set up:

```
CARD    DC    CL80    CARD INPUT AREA

DRAC    DC    CL80    REVERTED CARD AREA
```

Then a reformatting mask must be defined:

```
FMAT    DC    XL8'4F4E4D4C4B4A4948'

        DC    XL8'4746454443424140'

        DC    XL8'3F3E3D3C3B3A3938'

        - -   -----------------------

        DC    XL8'0706050403020100'
```

With the input record in DRAC the following instructions can be executed:

```
        MVC    CARD, FMAT    SET MASK

        TR     CARD, DRAC    INVERT
```

The inverted card image is now in the area CARD.

If it is anticipated that inverted files may occur within an installation, reporting programs may be written to read files optionally either normally or reversed. To determine if the file is in fact predominantly reversed in sequence, the average string length of a file (samples if large) can be determined and reverse sequence holds if the average is significantly less than 2. Alternatively, if the weighted moving average input string length calculated during a sort falls significantly (by some statistical measure) below 2, the sort can be halted by the program to allow reversed sorting to be substituted.

COMPARISON WITH UNIT RECORD SORTING

Two questions should be considered.

(i) **Should a card sorter be included in an installation?**

If enough time for all necessary sorting is available on the /360/20/2560 without rental increase or overtime at suitable times of the reporting cycle, then there is no necessity for a card sorter. Advantages of simpler operating and reduced space requirements are gained by not including a sorter.

Notice that the 360/20/2560 would not be justified only on the basis of sorting, since great advantage in operating time will rarely be gained (more often lost) and it would be an expensive sorter. Operating time for sorts is discussed under the next question. The conditions within an installation may, on the other hand, prevent a card sorter from being justified.

(ii) **On which machine should a particular file be sorted?**

It will not always be practical to sort a card file on the card sorter, for example if the control field contains binary or packed decimal data (summary cards) or is variable in format, or if a complex sequence determination is required. This last assumes that special sorting programs can be written within an installation because such programs would be small even if complex.

A sort performed on the /360/20/2560 will be more reliable since the necessary sequence checking will prevent "mis-sorting".

The time taken for a sort (exclusive of operator time) on either machine is proportional to the size of the file, the number of file passes and the feed speed. With a card sorter, the number of passes depends on the size and complexity of the control field. With the MFCM, the number of passes depends on presequencing and file size (hence double dependence) but is not dependent on the nature of the control field. Thus, for any particular sort there is a cutover file size, above which the MFCM is slower, below which faster.

The cutover file size, using the fairly sophisticated sorting algorithms and procedures of this paper, is roughly 5000 cards for a nine digit numeric sort, assuming random initial distribution and an 082 sorter. The effective cutover file size is raised (favouring MFCM) by complex control fields, operator time, presequencing and most other incidental factors. Appendix D3 will be found useful in estimating other cutover points, remembering that certain operating procedures described in this paper can reduce the number of passes below the number shown in that appendix.

INDEX TO APPENDICES

SECTION A MERGE EXAMPLES

1, 2 Straight Merge
3, 5, 6 Retrospective Merge
4 Comparison of Merges
7-9 Compressive Merge
10 Retrospective Merge (long input strings)
11 13 Merges using 5 stackers

SECTION B LOGIC DIAGRAMS

1 Straight Merge
2 Retrospective Merge
3 Compressive Merge

SECTION C TEST RESULTS

1 Explanation of Abbreviations
2 Sort Type v. Gain
3 String Length v. Gain
4 Stackers used v. Gain

SECTION D GRAPHICAL SUMMARY

1 String Length v. Gain
2 Input v. Output String Length
3 Passes v. Sort Type

SECTION E TEST PROGRAM LISTING

Appendix A1 — Straight Merge.

```
PRMRY  SECNY  ST.1   ST.2      PRMRY  SECNY  ST.1   ST.2
731                            978           755
       709                     628*          978
       500*   709                     664           021*
504*          731              922                  628
       583           500              095*          664
558                  504       412*                 922
869                  558              922    095*
       795           583       074*          412
       211*          795              030*   922
164*                 869              142           030*
440           164*             936                  074
       426    211                     672           142
       158*   426                     066*          672
336*          440              377*                 936
       746           158*             195    066*
563                  336              696    195
050*                 563       829           377
       227*          746              494*   696
403           050*             145*          829
       359    227              413                  145*
       336*   359              066*                 413
156*          403                     755           494
723                  156*             254*          755
       765           336       238           066*
858                  723       054*          238
       284*          765              295    254
043*                 858              186*   295
111           043*             883                  054*
793           111                     850           186
       140*   284                     136*          850
732*          793              141*                 883
       644           140*             862    136*
       993           644       811           141
342*                 732       394*          811
       551*          993              325*   862
186*          342*                    309*          325*
       922    551              855                  394
       551*   922              253*                 855
014*                 186*      066*          253*
       676           551              702    309
       558*          676              061*   702
297           014*                    285           061*
551           297              057*                 066
735           551                     523           285
       784    558                     631           523
753           735                     881           631
147*          753                     045*          881
       436*   784                     261    045*
683                  147*      084           057
       062*          436       118           084
982                  683       531           118
355*                 982              452    261
       345    062*                    798    452
       559    345              252*          531
223*          355                     755*   798
       032*   559              456                  252*
       065           032*      723                  456
       644           065       960                  723
791                  223              329*          755
       837           644       214*                 960
337*                 791       530           214*
       148*          837              728    329
       137*   148*             134*          530
147*          337                     450*   728
       536           137*      568                  134*
007*                 147              231*          450
       445*          536       402*                 568
657           007*                    979    231*
       021*   445              138*          402
755           657                     732*   979
PRMRY  SECNY  ST.1   ST.2      PRMRY  SECNY  ST.1   ST.2
```

Appendix A2 - Adventitious Gain with straight merge

```
PRMRY SECNY ST.1  ST.2
882         060*
      226*  353
529*        882
      619         226*
835               529
      754         619
      675*        754
725*              835
      860   675*
516*        725
      269*  860
      568         269*
213*              516
      209*        568
      238   209*
959         213
      393   238
      400   393
      389*  400
166*        959
255               166*
581               255
      470         389
      297*        470
357*              581
      832   297*
806         357
528*        806
      056*  832
      194         056*
      538         194
876               528
      747         538
      185*        747
630*              876
      092*  185*
672         630
619*        672
      794         092*
352*              619
      735*        794
543         352*
205*        543
      081*  735
      256         081*
156*              205
      259         256
      119*        259
      644   119*
263         156
522         263
941         522
      418*  644
560*        941
      898         418
497*              560
      634*        898
638         497*
      282*  634
713         638
466*        713
      761         282*
266*              466
      496*        761
899         266*
      792   496
      934   732
201*        899
      905*  934
239
                  201*
PRMRY SECNY ST.1  ST.2
```

One sequence derived where 2 were expected!

Appendix A3 - Retrospective Merge

```
PRMRY SECNY ST.1  ST.2    PRMRY SECNY ST.1  ST.2
731                       412*        922
      7C9                       664         021*
      500*  7C9           074*              412
      583         5CC           C95*        664
      795         583     936         C74*
504*        731                 922   C95
      211*  795                 C30*        922
      426   211*          377*              936
      158*  426           829         377
558         5C4           145*        829
869         558                 142         030*
164*              869           672         142
      746         158*    413               145
44C               164     066*              413
336*              44C           066*        672
      227*  746           238         C66*
      359   227*                195   C66
563         336                 696   195
      336*  359           054*        238
050*              563           494*        696
403               05C*          755   494
      765         336           254*        755
156*        4C3           883               054*
      284*  765                 295         254
723         156*                186*        295
      140*  284           141*        883
858               723     811         141*
043*              858           850   186
111               043*    394*              811
793               111     855         394
      644         14C           136*        850
      593   644           253*              855
732*        793                 862         136*
342*              732     066*              253
      551*  593                 325*  862
186*        342*                309*        325
      922   551           857*        C66*
      551*        922           702   309
      676   551                 C61*        702
      558*  676           084               857*
014*              186*          285         061
      784         558     118               084
      436*  784           531               118
297         C14*                523         285
551         297                 631   523
      C62*  436           252*        531
735         551                 881   631
753               735           045*  881
147*              753           261   C45*
      345         062*    456         252
683               147           452   261
      559         345           798         452
      C32*  559           723               456
982         683           96C               723
355*        982                 755*        798
223*              355           329*  755
      C65   C32*          214*              960
      644   C65           53C               214*
791         223                 728         329
      837         644     134*              530
337*              791           450*        728
147*        337           568         134*
      148*        837           231*  450
CC7*              147*    402*        568
      137*        148           979         231*
657         CC7*          138*              4C2
      536   137                 732*  979
      445*        536           387*        732
      C21*  445           335         138*
755               657     105*        335
978               755           762   387
628*              978           620*        762
922         628                 475*  620
PRMRY SECNY ST.1  ST.2    PRMRY SECNY ST.1  ST.2
```

Appendix A4 (cont.) ~ Comparison of Compressive Merge & Synchronised Retro. Merge

```
PRMRY  SECNY  ST.1  ST.2

       868    169*
239           201
959           239
247*                959
287           247
051*          287
102                 051*
557                 102
198*          557
589                 198
617           589
771           617
954           771
       045*   868
161*          954
       497    045*
779           161
       120*         497
386*                779
123*          386
       668          120*
782                 123
       536*   668
783           782
471*          783
667                 471
       616          536
       365*         616
685                 667
643*                685
       887    365*
       567*         887
       133*   567
160*          643
       497          133*
779                 160
918           779
671*          918
       293*         497
720                 671
898                 720
459*                898
       696    293*
791           459
       830    696
932           791
193*                932
       647*   830
651                 193*
       155*         647
966                 651
636*          966
       246    155*
       144*   246
865           636
708*                865
554*          708
       610          144*
544*                554
       325*         610
       400    325*
       179*   400
295*          544
       202          179*
       077*         202
463                 295
781                 463
346*          781
       317    077*
       265*   317

PRMRY  SECNY  ST.1  ST.2
```

```
PRMRY  SECNY  ST.1  ST.2

959                 239
       169*   905
247*          959
       868    169*
287                 247
051*                287
       045*         868
       497          045*
102                 051
557                 102
       120*   497
198*          557
       668          120
589                 198
617           589
771           617
       536*   668
954           771
161*          954
779           161*
       616          536
       365*         616
386*                779
       887    365
123*          386
       567*         887
782                 123*
       133*   567
783           782
471*          783
       497          133
667                 471
       293*         497
685                 667
643*                685
       696    293*
160*          643
       830          696
       647*         830
779                 160*
       155*   647
918           779
671*          918
       246    155*
       144*         246
720                 671
898                 720
459*                898
       610          144*
791           459
       325*   610
932           791
193*          932
651                 193
       400          325
       179*         400
966                 651
636*          966
       202    179*
       077*   202
865           636
708*                865
       317          077*
       265*         317
554*          708
       528    265*
       922          528
544*                554
       912*         922
295*          544

PRMRY  SECNY  ST.1  ST.2
```

Appendix A4 ~ Comparison of Straight Merge versus Retrospective Merge

```
PRMRY  SECNY  ST.1  ST.2

959                 239
       169*         905
247*                959
       868    169*
287           247
051*          287
       045*   868
       497          045*
102                 051
557                 102
       120*         497
198*                557
       668    120*
589           198
617           589
771           617
       536*   668
954           771
161*          954
779                 161*
       616          536
       365*         616
386*                779
       887    365*
123*          386
       567*   887
782                 123*
       133*         567
783                 782
471*                783
       497    133*
667           471
       293*   497
685           667
643*          685
       696          293*
160*                643
       830          696
       647*         830
779           160*
       155*   647
918           779
671*          918
       246          155*
       144*         246
720                 671
898                 720
459*                898
       610    144*
791           459
       325*   610
932           791
193*          932
651                 193*
       400          325
       179*         400
966                 651
636*                966
       202    179*
       077*   202
865           636
708*          865
       317          077*
       265*         317
554*                708
       528    265*
       922    528
544*          554
       912*   922
295*                544*

PRMRY  SECNY  ST.1  ST.2
```

```
       169*   905
       868          169*
239                 201
959                 239
       045*         868
247*          959
       497    045*
287           247
051*          287
       120*   497
102                 051*
557                 102
       668          120
198*          557
589                 198
617           589
771           617
       536*   668
       616          536
       365*         616
954           771
161*          954
779           161*
       887    365
386*                779
123*          386
       567*         887
       133*   567
782                 123*
       497          133
       293*         497
783           782
471*          783
       696    293*
667           471
685                 667
643*                685
160*          643
       830          696
       647*         830
       155*   647
       246          155*


779                 160
       144*         246
918           779
671*          918
720                 671
898                 720
459*                898
       610    144*
791           459
       325*   610
932           791
193*                932
651                 193*
       400          325
       179*         400
966                 651
636*          966
       202    179*
       077*   202
865           636
708*                865
554*          708
       317          077*
       265*         317
544*                554
       528    265*
       922    528
295*          544
       912*         922
```

Appendix A5 - Retrospective Merge using 3 Stackers

```
PRMRY  SECNY  ST.1   ST.2   ST.3      PRMRY  SECNY  ST.1   ST.2   ST.3
731                                   412*          922
       709                            074*                 412
       5C0*   7C9                            664                  C21*
       583           5C0              936                         C74
       795           583                     C95*          664
504*          731                            922                  095
558                         5C4              C30*   922
869                         558       377*          936
       211*   795                     829                         377
164*          869                     145*                 829
440           164*                           142    030*
       426    211                            672    142
       158*   426                     413           145
336*          440                     066*                        413
       746           158*                    C66*                 672
563                  336              238                  066*
050*                        563              195           066
       227*                 746              696    195
403                         050*      054*          238
       359                  227              494*                 696
       336*          359                     755    494
       765                  336              254*                 755
156*                 4C3                     295           254
       284*   765                            186*          295
723           156*                    883                         054*
       140*   284                            850                  186
858                  723                     136*   850
043*                 858              141*          883
111                  043*                    862    136*
793                  111              811           141
       644           14C              394*                 811
       993                  644       855                         394
732*                        793       253*                 855
342*          732                     066*          253
186*                 342                     325*          862
       551*                 993              309*   325
       922           551              057*                 066*
       551*   922                            702           3C9
       676           551                     061*                 702
       558*          676              084                         057*
014*                        186*             285                  061
       784                  558       118                         084
       436*          784              531                         118
297           C14*                           523                  285
551           297                            631    523
       C62*   436                     252*          531
735           551                            881    631
753                         735              045*   881
147*                        753              261    C45*
       345           062*             456           252
683                  147                     452    261
       559           345                     798           452
       C32*   559                     723                  456
982           683                     960                  723
355*                        982              755*          798
223*                 355                     329*                 755
       065                  032*             728    329
       644                  065              450*   728
791                         223       214*                 960
       837           644              530                  214*
337*          791                            231*          450
147*                        337       134*                 53C
       148*   837                     568                         134*
007*          147*                           979                  231
       137*   148                     402*                 568
657                  C07*             138*                        402
       536           137                     732*   979
       445*                 536              387*          732
       C21*   445                     335           138*
755                         657       105*          335
978                         755              762    387
628*                        978              620*          762
922           628                            475*                 62C
PRMRY  SECNY  ST.1   ST.2   ST.3      PRMRY  SECNY  ST.1   ST.2   ST.3
```

Appendix A6 - Retrospective Merge with input string length 5

```
PRMRY  SECNY  ST.1   ST.2      PRMRY  SECNY  ST.1   ST.2
297                            531           523
       858                     631           531
436           297              881           631
551           436                     85C    811
559           551                     855    850
683           559                     136*   855
982           683                     253           136
       C43*   858                     325           253
       111           043              702           325
       140           111              C57*          702
       732           14C       045*          881
       922           732       252           C45*
       186*   922                     061    C57
032*          982                     084    061
065           C32*                    118    084
223           C65                     285    118
       558    186              261           252
337           223              755           261
C07*          337                     452    285
       735    558                     456    452
       753           735              723    456
       062*          753              798           723
137                  C07*      134*                 755
       147           062              960           798
445                  137              214*          960
       345           147       450                  134*
       355           345              329           214
       644           355              530           329
628                  445       568                  45C
922           628                     728    530
       791    644              979           568
       837    791                     231*   728
       147*   837              138*          979
074*          922              335           138*
095           C74*                    402    231
377           C95              387           335
       148    147              62C           387
       536    148                     732    402
829           377              788                  620
       657           536              762           732
       755           657              1C5*          762
       978           755       802                  788
066*                 829       852                  802
       C21*          978       154*                 852
       412           021*             191           105*
195                  066       348                  154
238                  195              475           191
494                  238       424                  348
       664    412              727           424
883           494                     613    475
       922    664                     632    613
141*          883                     761    632
       936    922              867           727
       C30*   936                     134*   761
       142    C30*             042*          867
168           141              200           C42*
       145    142                     187    134
       413    145                     482    187
394           168              228           2C0
862                  394       388           228
       672           413       435                  388
       696           672       61C                  435
       755           696              529           482
       C54*          755              690           529
066*                 862       669                  61C
       254           054*      671                  669
309                  C66       820                  671
       295    254                     320*          690
       811    295                     362    320
523           309                     367    362
PRMRY  SECNY  ST.1   ST.2      PRMRY  SECNY  ST.1   ST.2
```

Appendix A7 — Compressive Merge

```
PRMRY SECNY ST.1  ST.2
731
      709
      500*  709
504*        731
      583         500
558               504
869               558
      795         583
      211*  795
164*        869
440         164*
      426   211
      158*  426
336*        440
      746         158*
563               336
050*        563
      227*  746
403         050*
156*              403
723         156
      359   227
      336*  359
858               723
043*              858
111               043*
793               111
      765         336
      284*  765
732*        793
342*              732
      140*  284*
186*        342
      644         140*
014*              186
      993   644
      551*  993
297         014*
551               297
735               551
      922         551
753               735
147*              753
683         147
      551*        922
      676   551
      558*  676
982         683
355*              982
223*              355*
      784         558
      436*  784
791         223*
      062*  436
337*              791
      345         062*
147*              337
      559         345
      032*  559
      065   032*
      644   065
007*        147
      837         644
      148*        837
      137*  148
657               007*
      536         137
      445*  536
755         657
978         755
PRMRY SECNY ST.1  ST.2
```

```
PRMRY SECNY ST.1  ST.2
628*        978
      021*        445
      664   021*
922               628
      095*        664
      922   095
412*              922
      030*        922
      142         030*
      672   142
074*        412
936               074
      066*  672
377*        936
      195   066*
      696         195
829               377
      494*        696
145*              829
413         145
066*        413
      755   494
238               066*
054*              238
      254*  755
      295         254
      186*        295
883         054*
      850   186
      136*        850
141*              883
      862         136*
811               141
394*        811
      325*  862
      309*        325
855               394
      702   309*
      061*        702
253*              855
      285         061*
066*              253
      523         285
      631   523
      881   631
      045*  881
      261   045*
057*        066
      452   261
      798         452
      755*        798
084               057*
118               084
531               118
252*        531
456               252
723               456
960         723
      329*  755
214*        960
530         214*
134*              530
      728   329
      450*        728
      231*  450
568               134*
      979         231
402*        568
138*              402
      732*  979
PRMRY SECNY ST.1  ST.2
```

Appendix A8 – Compressive Merge using 3 stackers

```
PRMRY SECNY ST.1  ST.2  ST.3
731
      709
      500*  709
504*        731
      583         500
558               504
869               558
      795         583
      211*  795
164*        869
440                     164
      426               211
      158*              426
336*                    440
      746   158*
      227*        746
      359   227
563         336
      336*  359
050*                    563
403               050*
156*        403
723               156
858                     723
043*                    858
      765         336
111                     043*
793                     111
      284*  765
732*        793
      140*              284
      644   140*
      993         644
342*              732
186*                    342
014*        186
      551*        993
297               014*
551         297
735                     551
      922               551
753                     735
147*                    753
683               147
      551*              922
      676   551
      558*  676
982         683
355*                    982
223*              355
      784         558
      436*  784
791                     223*
337*        791
147*                    337
      062*              436
      345   062*
007*        147
      559   345
      032*        559
657               007*
      065         032
      644         065
      837               644
755                     657
978                     755
      148*              837
      137*        148
628*                    978
      536               137*
PRMRY SECNY ST.1  ST.2  ST.3
```

```
PRMRY SECNY ST.1  ST.2  ST.3
      445*  536
922         628
412*        922
074*              412
      021*        445
      664   021*
936         074
      095*        664
      922   095
      030*        922
377*              936
      142         030*
      672               142
829                     377
      066*              672
      195         066
      696   195
      494*              696
145*                    829
413               145
066*        413
      755   494
238                     ~~066*~~
054*              238
      254*  755
      295         254
      186*        295
      850               186
      136*  850
883         054*
      862   136
      325*        862
141*              883
811         141
      309*              325
      702   309
      061*              702
394*                    811
855         394
253*                    855
      285         061*
066*              253
      523         285
      631   523
      881   631
      045*              881
      261               045*
057*                    066
      452               261
      798         452
      755*  798
084         057*
118         084
531         118
252*              531
456         252
723                     456
960               723
      329*        755
      728   329
214*              960
530               214*
134*                    530
      450*              728
      231*  450
      979         231
      732*              979
568                     134*
402*        568
      387*  732
PRMRY SECNY ST.1  ST.2  ST.3
```

Appendix A9 – Compressive Merge
with input string length 5

```
PRMRY  SECNY  ST.1   ST.2

297
       858
436           297
551           436
559           551
683           559
982           683
       043*   858
       111           043
       140           111
       732           140
032*          982
065           032*
223           065
337                  223
007*                 337
       922           732
       186*          922
137                  007*
445           137
       558    186
628           445
       735    558
922           628
       753    735
       062*   753
       147           062
       345           147
074*          922
095           074*
377           095
       355           345
       644           355
829                  377
       791           644
       837           791
066*                 829
       147*          837
       148    147
       536    148
195                  066*
238           195
494           238
883           494
       657    536
       755    657
       978    755
141*          883
168                  141
394                  168
       021*   978
       412    021*
862                  394
       664           412
       922           664
066*                 862
309           066
       936           922
       030*          936
       142           030*
       145    142
       413    145
523           309
       672    413
531           523
631           531
881           631
       696    672
       755    696
       054*   755

PRMRY  SECNY  ST.1   ST.2
```

```
PRMRY  SECNY  ST.1   ST.2

       254           054
045*          881
252           045*
261                  252
       295           254
755                  261
       811           295
134*                 755
       850           811
       855           850
       136*          855
450           134
       253    136
       325    253
       702    325
568           450
979           568
138*                 979
       057*   702
       061           057*
       084           061
       118           084
       285           118
335                  138
       452           285
387                  335
620                  387
       456           452
       723           456
       798    723
       960    798
       214*   960
788                  620
802                  788
852                  802
154*                 852
348           154*
       329    214
       530    329
424           348
727           424
       728    530
867           727
       231*   728
042*                 867
200                  042*
228                  200
388                  228
       402           231
435                  388
       732           402
       762    732
       105*   762
610                  435
669                  610
671                  669
820                  671
289*          820
       191    105*
       475    191
597           289
       613    475
825           597
       632    613
       761    632
       134*          761
951                  825
240*                 951
       187           134*
       482

PRMRY  SECNY  ST.
```

Appendix 10 – Retrospective Merge
with input string length 8

```
PRMRY  SECNY  ST.1   ST.2

       165           153
       202           165
       206           202
       557           206
237                  227
454                  237
551                  454
568                  551
       627           557
652                  568
       641           627
       651           641
       749    651
662           652
832           662
       821    749
       084*   821
056*          832
060           056*
194           060
       334    084
213           194
255           213
357           255
       351    334
       553    351
528           357
619           528
       587    553
       643    587
735           619
       673           643
       814           673
266*                 735
       824           814
       946           824
       010*          946
       086           010*
       192           086
       242           192
       375           242
418                  266
       480           375
497                  418
       664           480
557                  497
589                  557
617                  589
668                  617
       810    664
771           668
779           771
830           779
       963    810
123*          830
       156*   963
155           123*
246           155
       515    156
325           246
400           325
459           400
528           459
       546    515
651           528
       567    546
       582    567
       698    582
698                  651
716                  698

PRMRY  SECNY  ST.1   ST.2
```

```
PRMRY  SECNY  ST.1   ST.2

       806           698
865                  716
       882           806
009*                 865
       082*          882
346                  009*
       211           082
       426           211
373                  346
444                  373
       440           426
       644           440
483                  444
501                  483
603                  501
860           603
       723    644
       858    723
       993    858
938           860
121*          938
       043*   993
       062    043*
       111    062
       345    111
152           121
282           152
352           282
       536    345
523           352
721                  523
       551           536
       683           551
       791           683
763                  721
787                  763
840                  787
       007*          791
866                  840
889                  866
972                  889
107*                 972
       095           007*
       142           095
310                  107
       377           142
479                  310
       494    377
557           479
       702    494
648           557
696           648
709           696
       811    702
767           709
928           767
       855    811
       881    855
       045*   881
000*          928
231           000*
       057    045
       252    057
311           231
       387    252
315                  311
419                  315
       473           387
697                  419
       613           473

PRMRY  SECNY  ST.1   ST.2
```

Appendix 11 ~ Retrospective Merge using 5 stackers & input string length 8

```
        PRMRY SECNY ST.1  ST.2  ST.3  ST.4  ST.5
              277                     270
        458                           271
              388                     277
              402               388
              617               402
        739                     458
              644         617
              782         644
        744               739
        155*              744
        170                                 155
        353                                 170
        417                           353
        580                           417
        596                     580
        652                     596
        747                     652
        882               747
              945         782
        959               882
              193*        945
              280                           193
              331                           280
              335                           331
              382                           335
out           562                           382
              682                     562
of            684               682
              845               684
synch.        951               845
              086*        951
              158   C86
              227   158
              359   227
              403   359
              551                           403
              922                           551
              032*              922
        166*              959
              065         032*
              342         065
        185               166
        352               185
              445         342
        543               352
              664   445
        560         543
        638                                 560
        713                           638
              755                     664
        898                           713
              922                     755
        899                           898
        201*                          899
              978               922
              030*              978
              074               030*
              145               074
              195               145
              238               195
        239                     201
              254               238
        282                     239
              295               254
        287                     282
        536                     287
              325               295
              394               325
              523         394
              631         523
        PRMRY SECNY ST.1  ST.2  ST.3  ST.4  ST.5
```

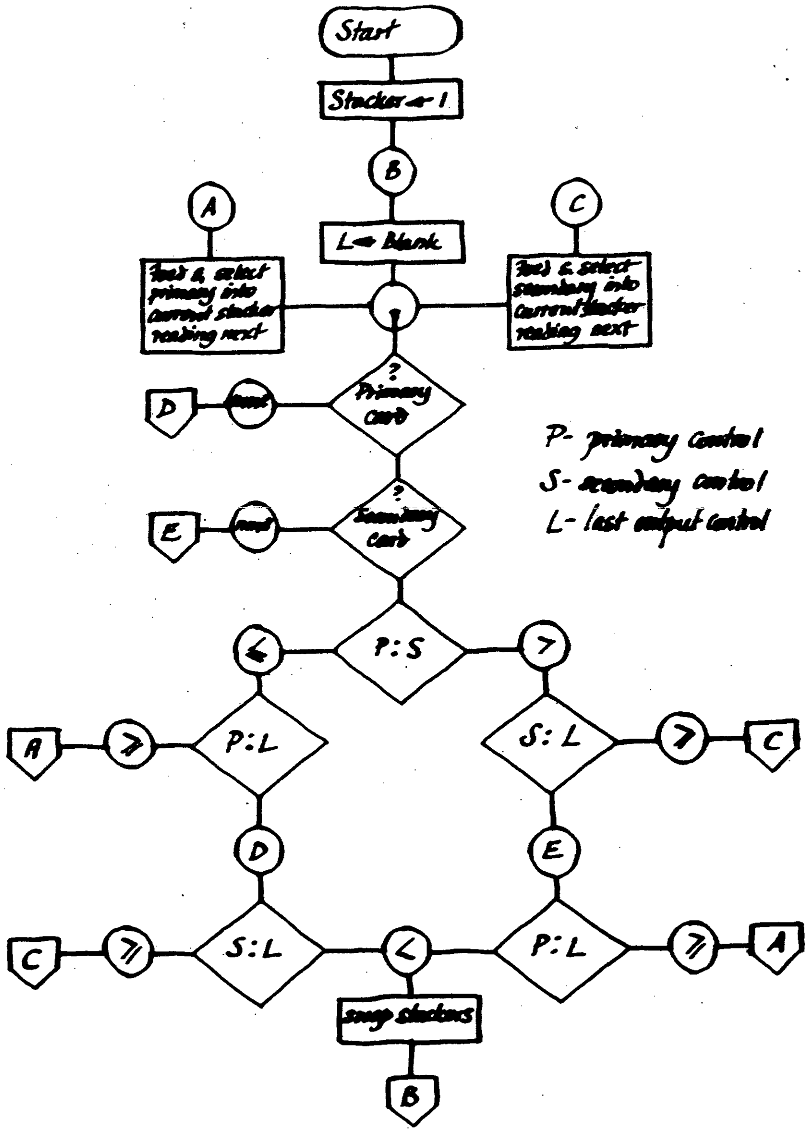
Appendix 12 ~ Compressive Merge using 5 stackers input string length 8

```
        PRMRY SECNY ST.1  ST.2  ST.3  ST.4  ST.5
              784         713
              830         784
              911         830
              089*        911
        543         352
        560                     543
        638                     560
        713                     638
        898                     713
              128         089*
              194         128
out           535         194
              548                     535
of            579                     548
              775                     579
synch.        813               775
              857               813
              C05*              857
        899                     898
        201*                    899
        239               201
        282               239
        287               282
        536               287
              057               005*
              223               057
        616                                 536
        696                           616
        782                           696
        783                           782
        144*                          783
        160                     144
        179                     160
        202                     179
        317                     202
              270               223
        581               317
        636                                 581
              277               270
              388               277
              402   388
              617   402
              644                           617
        708                                 636
              782                           644
        781                                 708
        980                                 781
              945                           782
              193*                    945
        051*                          980
        085                           051*
        238                           085
              280                     193
              331               280
              335         331
              382         335
        345                           238
        429               345
        435         429
        591         435
              562         382
              682   562
        638         591
        828         638
              684   682
              845   684
        115*                                828
              951                           845
              086*                          951
              158                           086*
        PRMRY SECNY ST.1  ST.2  ST.3  ST.4  ST.5
```

Appendix A13 – Synchronised Retrospective Merge using 5 stackers
input string length 8

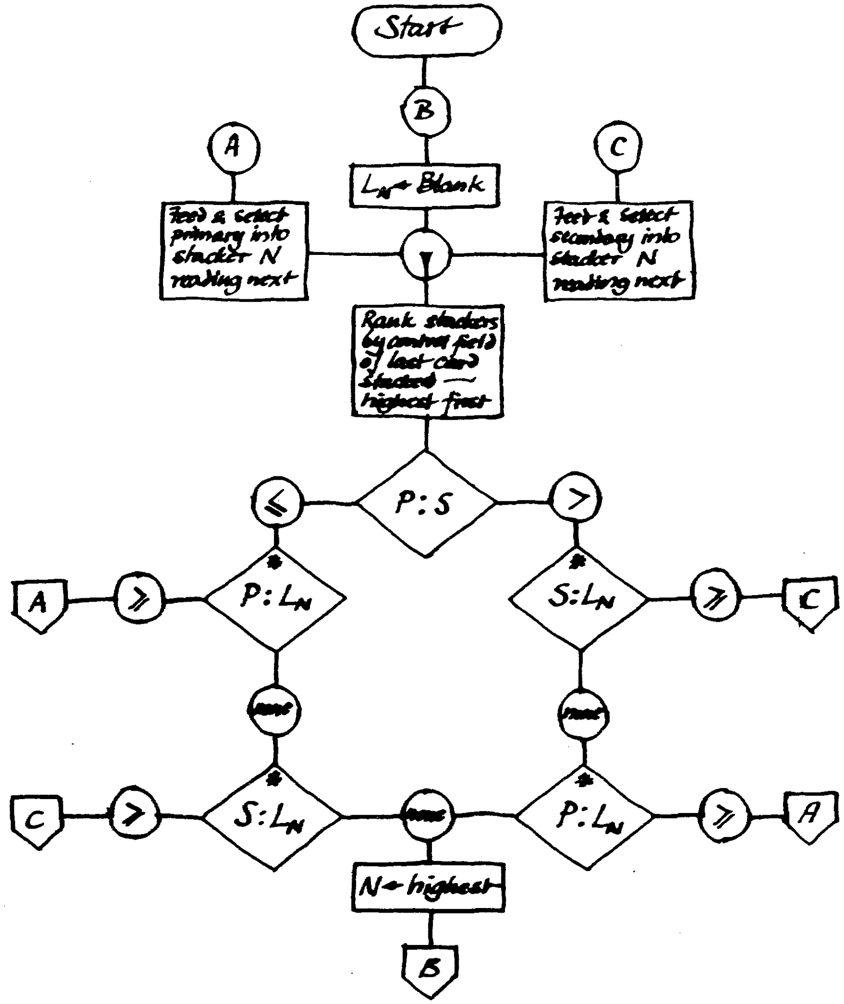
```
PRMRY SECNY ST.1  ST.2  ST.3  ST.4  ST.5

560         543
638         560
      713   622
713         638
898                                 713
      784                           713
      830                           784
      911                           830
899                                 898
201*                                899
      089*                          911
      128                           089*
      194                           128
      535                     194
239                           201
282                           239
287                           282
536                           287
      548         535
616               536
      579         548
      775         579
696               616
782         696
      813   775
783         782
144*        783
      857   813
      005*  857
      057   005*
      223   057
160                                 144
179                                 160
202                                 179
317                                 202
      270                           223
      277                           270
      388                           277
581                           317
      402               388
      617               402
636                     581
      644         617
708               636
      782         644
781               708
980               781
      945         782
      193*        945
051*              980
085               051*
238         085
      280   193
345         238
      331                           280
      335                     331
      382                     335
429                           345
      562                     382
435                           429
591                           435
      682                     562
638                     591
828                     638
      684               682
      845               684
115*                    828
      951               845
      086*              951
      158         086
PRMRY SECNY ST.1  ST.2  ST.3  ST.4  ST
```



Appendix B1 – Logic Diagram for Straight Merge/Split.

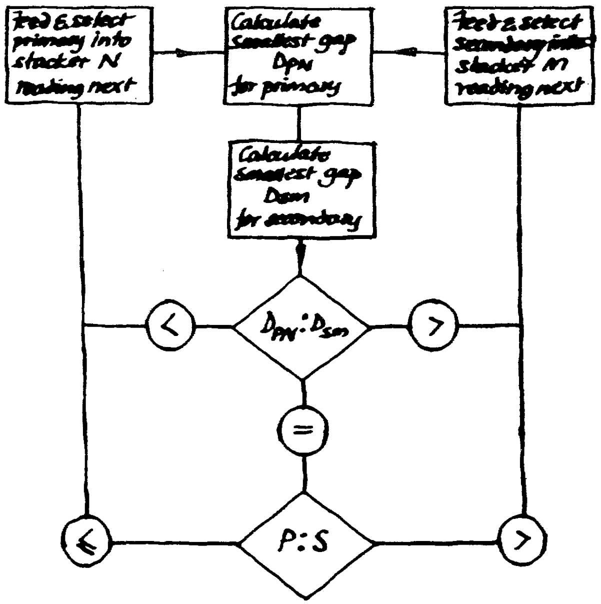
Appendix B2 - Logic Diagram for Retrospective Merge



Last Card logic is not included.

* – A search is made through established rank for the first stacker N which satisfies the '≥' exit condition.

Appendix B3 - Logic Diagram for Compressive Merge



Last card logic is not included.

N	Total number of cards passed
Np	Number of cards through primary
Ns	Number of cards through secondary
Sp	Number of strings through primary
Ss	Number of strings through secondary
S_1	Number of strings into stacker 1

- -

S_5	Number of strings into stacker 5
L_1	Average length of input strings
L_o	Average length of output strings
G	Gain = L_o/L_1
T	Type of sort/merge
M	Straight merge/split
R	Retrospective merge
C	Compressive merge
R_s	Retrospective merge with synchronisation
K	Number of stackers used
X	Variable Merge

Appendix C1 - EXPLANATION OF ABBREVIATIONS

L_1	T	G	L_o	N	N_p	N_s	S_p	S_s	S_1	S_2	S_3	S_4	S_5
2	R	3·9	8·1	802	387	415	184	206	20	20	20	20	19
	C	4·6	9·4	809	415	394	203	195	18	17	17	17	17
	R_s	4·0	8·1	827	416	411	204	204	21	21	20	20	20
5·3	R	2·8	14·8	754	364	390	67	75	11	10	10	10	10
	C	2·9	15·2	686	390	296	72	57	9	9	9	9	9
	R_s	3·0	15·7	768	390	378	72	73	10	10	10	10	9
8·4	R	2·3	19·3	772	382	390	45	47	8	8	8	8	8
	C	2·2	18·7	746	390	356	46	42	8	8	8	8	8
	R_s	2·4	20·3	773	390	383	46	46	8	8	8	7	7

Appendix C2 - Sort Type versus Gain

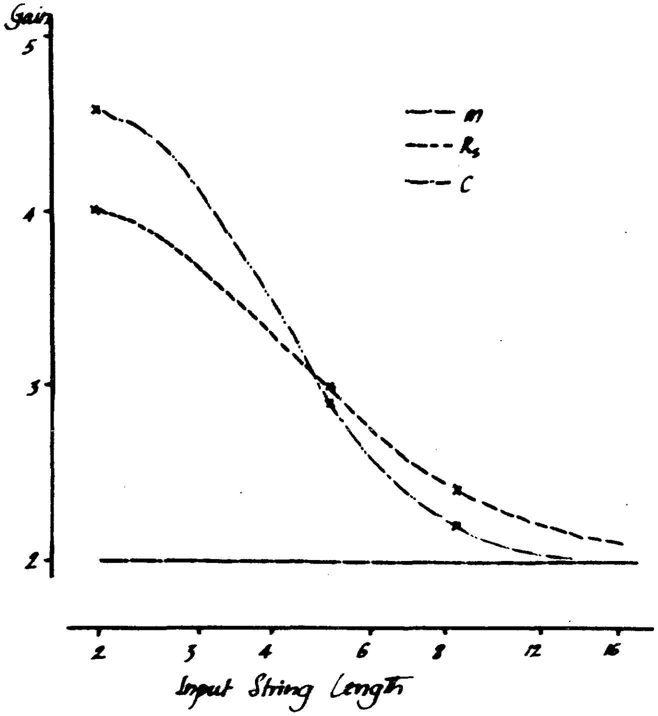
Stackers used = 5

T	L_1	G	L_0	N	N_p	N_s	S_p	S_s	S_1	S_2	S_3	S_4	S_5
R	2	3·9	8·1	802	387	415	184	206	20	20	20	20	19
	5·3	2·8	14·8	754	364	390	67	75	11	10	10	10	10
	8·4	2·3	19·3	772	382	390	45	47	8	8	8	8	8
C	2	4·6	9·4	809	415	394	203	195	18	17	17	17	17
	5·3	2·9	15·2	686	390	296	72	57	9	9	9	9	9
	8·4	2·2	18·7	746	390	356	46	42	8	8	8	8	8
R_s	2	4·0	8·1	827	416	411	204	204	21	21	20	20	20
	5·3	3·0	15·7	768	390	378	72	73	10	10	10	10	9
	8·4	2·4	20·3	773	390	383	46	46	8	8	8	7	7

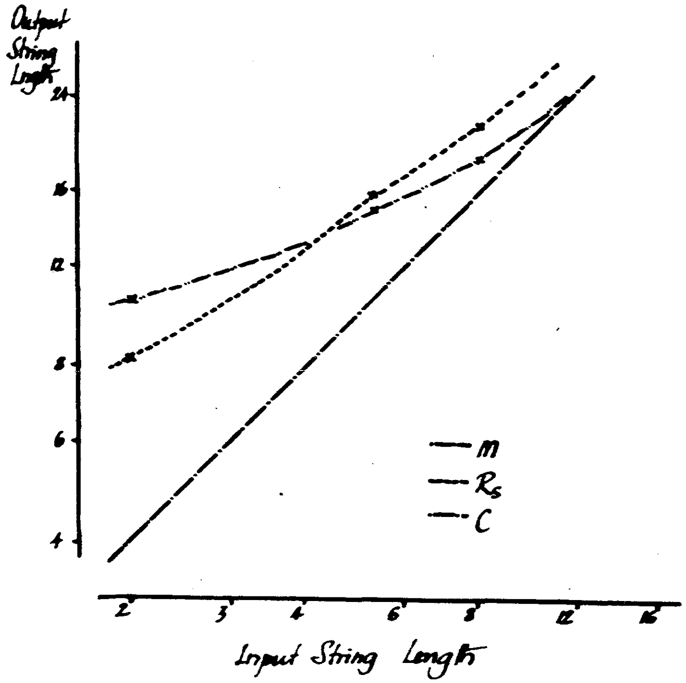
Appendix C3 – Input String Length versus Gain
Stackers used = 5

T	K	G	L_0	N	N_p	N_s	S_p	S_s	S_1	S_2	S_3	S_4	S_5
R	2	2·6	5·2	813	415	398	203	197	77	77			
	3	3·1	6·2	805	415	390	203	195	43	43	43		
	4	3·5	7·3	802	387	415	184	206	28	28	27	27	
	5	3·9	8·1	802	387	415	184	206	20	20	20	20	19
C	2	2·7	5·6	828	415	413	203	205	75	74			
	3	3·4	7·0	828	415	413	203	205	40	39	40		
	4	4·0	8·1	805	415	390	203	195	25	25	25	25	
	5	4·6	9·4	809	415	394	203	195	18	17	17	17	17
m	2	2·0	4·2	825	415	410	203	203	100	99			
R_s	2	2·5	5·1	827	416	411	204	204	81	81			
	5	4·0	8·1	827	416	411	204	204	21	21	20	20	20

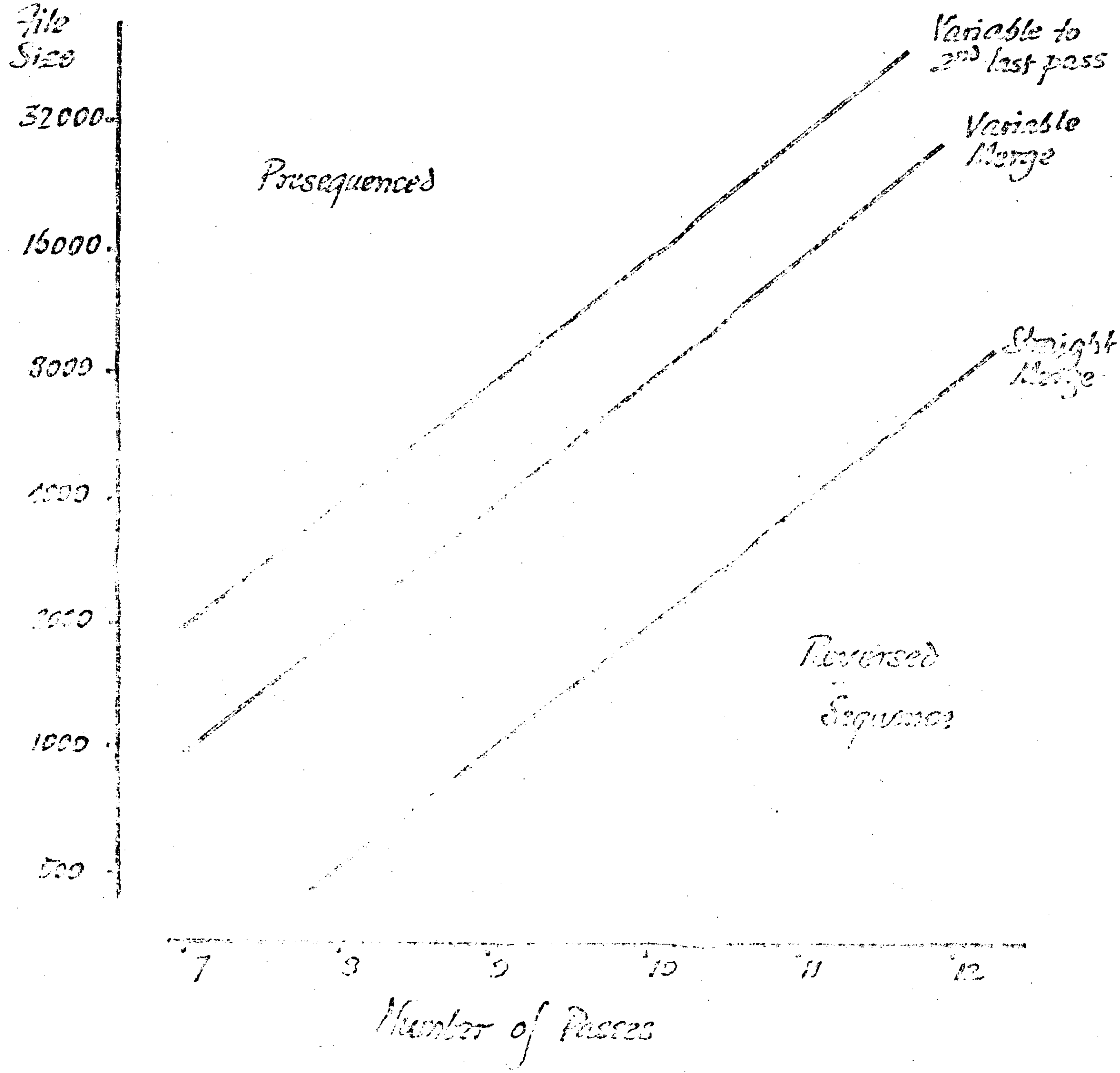
Appendix C4 – Stackers used versus Gain
Input String Length 2.



Appendix D1 – Input String Length versus Gain
Stackers used = 5



Appendix D2 - Input versus Output String Length
Stackers used = 5



Appendix D3 - File Sizes v. Passes

An appropriately drawn vertical line will give the cutover point against card sorting 082.

APPENDIX E

The Sort Simulation Program

On the following five pages, the program used for deriving the results of this paper is listed with interspersed comments.

The program listing is given for interest & checking only, since it can hardly be presented as a model, having grown piecemeal over two or three days from a modest beginning with modest aims. It has no source program since it was punched directly into self-loading format.

The program used 3K of 1401, HLE compare, advanced programming, any 1403 and a 1402 with punch feed read. The loader, a standard one, uses modify address. Primary hopper input is to the read feed, secondary to the punch feed, of the 1402 card reader. A carriage tape with channel 1 punched is necessary, and 12 channel is interrogated for overflow. Input format is 26 three digit numbers per card in c.c. 1–78, the numbers being processed in turn from RIGHT to LEFT.

SORTING PROGRAM

CLEAR STORAGE & BOOTSTRAP CARDS

```
,008015,022026,030034,041,045,053L0721001026bbbbbbbb/0991,001/001118I0?1bbbbbbC1
L096116,105106,110110B101/I99,027‡099029⌑027B00110 0bR026bbbbbbbbbbbbbbbbbbbbbbC2
,008015,022029,036040L071381,3543581001/080,001V00600611M467409,004424MbBOOTLD01
L070412,366367,374381,386390,3973981001M468M003S474005#V358005KM007470bbLOADER02
L069442,406420,424431,439001,0010011001M469000⌑000A473409B443469~B374bbbLOADER03
L071474,447455,462466,468469,471474B350S469B424470~M374420B40607⌑bb0011bLOADER04
```

CONTROL ROUTINE FOR STRAIGHT MERGE

```
50150~bbb~bbb~bbb~b~F0~FJ~/180~C503506~B571T~C509503~B551T~BW01~B519bbbbbbbbbbbb
55155~C509506~BX01T~BW51~B519~C509506~B587T~B563~C509503~BX01T~B543bbbbbbbbbbbbb
```

ROUTINE TO GET NEXT PRIMARY NUMBER

```
70158~H734~M878503~B738503b~M875878~M737~B000~bbb~/080~,001~1~L078878~B705bbbbbb
~PRIMARY INPUT AREA INITIALISATION
87801bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
```

ROUTINE TO GET NEXT SECONDARY NUMBER

```
90130~H934~MS78506~B938506b~MS75S78~M937~bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
93129~Bbbb~bbb~/080~,001~4R~L078S78~B905bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
~SECONDARY INPUT AREA INITIALISATION
S7801bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
```

```
SORTING PROGRAM - PAGE 2
------------------------

I/O STORAGE AREA
T0143~bbb~bbb~bbb~bbb~bbb~bbb~bbb~bbb~bbb~bbb~bbb~bbb~bbb~bbb~bbb~‡bbbbbbbbbbbbbbbbb

SET-UP & PRINT ROUTINE
U0154~HU94~BV01@~H089000~/299~BU59T‡3b~MT‡32‡3~CT‡6T‡3~BU78T~MT‡3T‡6~bbbbbbbbbbbb
U5541~M299~H0890‡6~BU90T‡1‡~BU21~MU892‡4~BU48~⁑~2~B000bbbbbbbbbbbbbbbbbbbbbbbbbb
~PAGE OVERFLOW ROUTINE
V0123~HV23~LV64241~FJ~FA~2~FS~2~Bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
V2450~PRMRYbSECNYbST.1bbST.2bbST.3bbST.4bbST.5bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb

FEED PRIMARY & GET NEXT
W0150~HW37~MW40W18~M503bbb~B701~M503T03~BU01~Pbbb~T15bbbbbbbbbbbbbbbbbbbbbbbbbbb
X5120~M503509~B701~BW23bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb

FEED SECONDARY & GET NEXT
W5150~HW87~MW40W68~M506bbb~B901~M506T09~BU01~Bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
X7120~M506509~B901~PW73bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
00628~N027~HW22X51~HW72X71~B350~bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb

SWAP STACKERS
X0150~HX26~MX32W40~MX29X32~MW40~Bbbb~T15~T21bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
X3314~MX46509~B519~bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
00629~N028~M017X01~N~HX26X33~B350~bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
```

SORTING PROGRAM - PAGE 3

CONTROL ROUTINE FOR RETROSPECTIVE MERGE

```
Y0148mF@mFJm/180mBJ01mC503506mBY57TmBM01mBY41mBW01mBY09mBN01mBP01bbbbbbbbbbbbbb
Y4933mBW51mBY09mBN01mBY69mBY49mBM01mBP01mBY33mbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
```

ROUTINE TO RANK THE STACKERS BY LAST FED NUMBER

```
J0148mHK11mPT13Z13mHZ15T15mHZ21T21mBJ71T25‡mHZ27T27mBJ71T31‡bbbbbbbbbbbbbbbbbbbb
J4927mHZ33T33mBJ71T37‡mHZ39T39mBK01bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
K0149mH089000mBbbbZ/3‡mH0940‡6mBK88ZJ3‡mCZJ8Z/8mBK60TmH0940!6bbbbbbbbbbbbbbbbbbb
K5050mBK23mbbbbbbmMZJ8K59mMmMZ/8ZJ8mMmMK59Z/8mMmBK43mH0890‡6mBK08mbbbbbbbbbbbbbb
mSTORAGE AREA FOR RANKED STACKERS
Z1331mbbbmbbbmbbbmbbbmbbbmbbbmbbbmbbbmbbbmbbbm‡bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
```

LOOK FOR PRIMARY SLOT

```
M0154mH099mH089000mB0?0Z/3‡mCZ/8503mBM43TmMZ/5W40mB0?4mH0890‡6mBM12mbbbbbbbbbbbb
```

LOOK FOR SECONDARY SLOT

```
N0154mH099mH089000mB0?0Z/3‡mCZ/8506mBN43TmMZ/5W40mB0?4mH0890‡6mBN12mbbbbbbbbbbbb
```

FORCE STACKER WHEN NO SLOT

```
P0115mMP14Z18mBK01mbbbmbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
```

SORTING PROGRAM - PAGE 4

```
~CONTROL ROUTINE FOR COMPRESSIVE MERGE
Q0153~F@~FJ~/180~BJ01~M503R03~BR04~M506R03~BR04~CQ88Q82~BQ62T~MC85W40bbbbbbbbbbb
Q5435~BW01~BQ09~MQ79W40~BW51~BQ09~bbb~bbb~bbb~bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
Q8912~BR04~BJ01~BQ24bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
00612~HQ23Q89~B350bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb

ROUTINE CALCULATING GAP BETWEEN NUMBER & STACKER
R0156~bbb~HR67~MQ82Q88~M~H089000~BR68Z/3‡~CZ/8R03~BR93T~MZ/5Q79~SZ/8R03bbbbbbbbb
R5750~MR03Q82~Bbbb~H089000~S?06Z18~S?06Z18~BR43~H0890‡6~BR23~500bbbbbbbbbbbbbbbb


GET NEXT PRIMARY NUMBER - MODIFIED TO SYNCHRONISE THE RETROSPECTIVE MERGE
70153~H753~M503759~M878503~B760503b~M875878~M756~C759503~B‡01T~Bbbbbbbbbbbbbbbbbb
75446~bbb~bbb~/080~,001~1~L078878~B712~BY41b~BY57Tbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
‡0134~H‡26784~B‡23Y24T~H‡26789~MbbbY24~B750~bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb


GET NEXT 2NDRY. NUMBER - MODIFIED TO SYNCHRONISE THE RETROSPECTIVE MERGE
90153~H953~M506759~MS78506~B954506b~MS75S78~M756~C759506~P‡41T~Bbbbbbbbbbbbbbbbb
95426~/080~,001~4R~L078S78~B912~BY69bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
‡4134~H‡66979~B‡63Y24T~H‡66789~MbbbY24~B950~bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
~FORCE SYNCHRONISATION IN CONTROL ROUTINE
00616~HY32P01~HY64~B350bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
```

THE PRECEDING CARDS,CONCERNED WITH SYNCHRONISATION, ARE ONLY INCLUDED WHEN
WHEN THIS VERSION OF THE RETROSPECTIVE MERGE IS REQUIRED!

SORTING PROGRAM - PAGE 5

```
ᵐEND CARD
00652ᵐB511BᵐB047CᵐMT43T37ᵐB047DᵐMT37T31ᵐB047EᵐMT31T25ᵐBY01GᵐBQ01ᵐbbbbbbbbbbbbbbb
```

SENSE SWITCH SETTINGS

```
  B ON - STRAIGHT MERGE & SPLIT
   G ON - RETROSPECTIVE MERGE
  ELSE - COMPRESSIVE MERGE

   C ON - 5 STACKERS
  D ON - 4 STACKERS
  E ON - 3 STACKERS
  ELSE - 2 STACKERS
```