1 SUMMARY

This subroutine compares two files of text (in fixed-length records) to discover where they differ. It finds a sequence of record insertions and deletions that converts the first to the second. The standard output is a listing showing insertions, deletions and common records.

ATTRIBUTES — **Version:** 1.0.0. **Remark:** This is a slightly rewritten version of OE05 and supersedes it. **Types:** OE15A. **Calls:** None. **Original date:** July 1985. **Origin:** S. Marlow and J.K. Reid, Harwell.

2 HOW TO USE THE PACKAGE

2.1 Argument list

```
CALL OE15A(INF1,INF2,LREC)
```

INF1 is an INTEGER variable which must be set by the user to the input stream number for data of the first file. It is not altered.

INF2 is an INTEGER variable which must be set by the user to the input stream number for data of the second file. It is not altered.

LREC is an INTEGER variable which must be set by the user to the number of characters in each record. **Restriction:** LREC≤256.

2.2 Input

Each record is normally read into a CHARACTER string of length LREC from files INF1 and INF2 by the small auxiliary subroutine OE15C, using FORMAT(A). For any other action the user must write another subroutine (see section 2.5).

2.3 Output

Records of the combined file are normally passed one at a time to subroutine OE15D and are printed with a label to indicate their origin. The labels consist of a field for each file; each label indicates the position of the record in the file or blank if it was not there. An example is shown in section 5. Where the number of characters in each record is greater than 120 the label fields are condensed to have length 1 and contain blank, 1 (for the first file) or 2 (for the second).

Normally only the first and last of a sequence of common records is printed, but a full listing is available if the user changes the variable SHORT (see section 2.4). For any other action the user must write a subroutine OEOUT (see section 2.5).

2.4 Use of Common

The subroutine contains the following COMMON areas:

```
COMMON/OE15G/LHASH,KHASH,NRMAX,LCRECR,LIRECR,LPD,LP,SHORT,LSWTCH
COMMON/OE15H/HASH(2047)
COMMON/OE15I/CRECRD(36255)
COMMON/OE15J/IRECRD(5503)
COMMON/OE15K/SPLIT,MCAND,LIMIT
COMMON/OE15L/CAND(3000)
COMMON/OE15M/IDC(5)
```

Most of the scalars are given default values by BLOCK DATA. These can be changed by the user declaring the block and assigning new values.

LHASH is an INTEGER variable, with default value 2047, specifying the range used for hash coding (to ease the recognition of identical records a simple function associates an integer in the range (1,LHASH) with any record). It also specifies the length of array HASH.

KHASH is an INTEGER variable, with default value 2, specifying the number of groups of four characters used to calculate the hash value.

NRMAX an INTEGER variable, with default value 500, which is used with LCRECR to determine the maximum number of distinct source records that can be held at once in main storage.

LCRECR is an INTEGER variable specifying the length of the array CRECRD which is used to determine NR, the maximum number of distinct source records that may be held at once in main storage. NR=MIN(NRMAX,(LCRECR−255)/LREC).

LIRECR is an INTEGER variable specifying the length of the array IRECRD which is used to determine LF, the maximum number of records of either file that may be held at once in main storage. LF=((LIRECR−3)−4*NR)/7.

LPD in an INTEGER variable, with default value 0, specifying the Fortran stream number for messages about finding correspondences (see section 4). If LPD≤0 no such messages are output.

LP is an INTEGER, with default value 6, specifying the FORTRAN stream number for output of the combined files (see section 2.3).

SHORT is a LOGICAL variable, with default value .TRUE., which controls the printing. OE15D prints a full list of the combined files if SHORT has the value .FALSE. and otherwise prints only the first and last of a sequence of common records with a note of the number of unprinted records.

LSWTCH is a LOGICAL variable, with default value .TRUE., specifying the form of the label field indicating the origin of the output record. If LSWTCH is .TRUE. and LREC≤120, the label indicates the position of the record in the file or blank if it was not there; otherwise the label is condensed to 1 (for the first file) or 2 (for the second file).

HASH is an INTEGER array of length LHASH used for workspace.

CRECRD is a CHARACTER*1 array of length LCRECR used to store the source records. The default size of 36 255 is suitable for LREC≤72.

IRECRD is an INTEGER array of length LIRECR used for workspace. The default size of 5503 is suitable for LREC≤72.

SPLIT is a REAL variable with default value 0.5, whose purpose is explained in section 4. It is the proportion of correspondences actually used if the Hunt and Szymanski algorithm is temporarily suspended for any reason.

MCAND is an INTEGER variable with default value 1000, specifying the maximum number of candidate correspondences that can be stored (see section 4).

LIMIT is an INTEGER variable with default value 100 000 specifying the maximal number of inner loop executions before output is performed.

CAND is an INTEGER array of length at least 3*MCAND.

IDC is an INTEGER workspace of length 5 in which IDC(1), IDC(2), IDC(3) will respectively contain the number of insertions to file 1, the number of deletions from file 1 and the records common to both files.

2.5 Modifying the input and output subroutines

To use the subroutine with alternative input and output subroutines:

```
CALL OE15N(INF1,INF2,LREC,OEIN,OEOUT)
```

INF1 is an INTEGER variable which must be set by the user to the input stream number for data of the first file. It is not altered.

INF2 is an INTEGER variable which must be set by the user to the input stream number for data of the second file. It is not altered.

LREC is an INTEGER variable which must be set by the user to the number of characters in each record. **Restriction:** LREC≤256.

OEIN is an input subroutine which must be declared in an EXTERNAL statement in the calling program (see section 2.6).

OEOUT is an output subroutine which must be declared in an EXTERNAL statement in the calling program (see section 2.7).

2.6 Form of alternative input subroutine

The subroutine must read one record and have the following form:

```
SUBROUTINE OEIN(IN,IR,LREC,ENDF)
```

IN is an INTEGER variable specifying the stream number of the file from which a record is required. It must not be altered by OEIN.

IR is a CHARACTER*1 array of length LREC. Unless the whole of file IN has been read, OEIN must place the next record of file IN in IR.

LREC is an INTEGER variable specifying the length of the array IR. Its value will be that of the argument LREC in the call of OE15A. It must not be altered by OEIN.

ENDF is a LOGICAL variable which must be set to .FALSE. unless the whole of the file IN has been read in which case it must be set to .TRUE..

2.7 Form of alternative output subroutine

The subroutine must have the following form:

```
SUBROUTINE OEOUT(IC,IR,IRL,IRLL,LREC)
```

IC is an INTEGER variable whose value must not be altered by OEOUT. Possible values are:

0 at completion of the comparison.

1 for an insertion to file 1.

2 for a deletion from file 1.

3 for a record common to both files.

IR is a CHARACTER*1 array of length LREC containing the record (IC =1,2,3 only).

IRL is a CHARACTER*1 array of length LREC which contains the record that was in IR on the last call. It must be preserved between successive entries.

IRLL is a CHARACTER*1 array of length LREC which contains the record that was in IRL on the last call. It must be preserved between successive entries.

LREC is an INTEGER variable specifying the length of the array IR. Its value will be input by calling OE15A. It must not be altered by OEOUT.

2.8 Auxiliary subroutines

There are six auxiliary subroutines, called OE15B/C/D/E/F/N. OE15B determines whether records are insertions, deletions or common to both files. Subroutines OE15C and OE15D are for input and output and have been explained in sections 2.2 and 2.3. Subroutines `OE15E` and `OE15F` test whether two records are identical and calculate a hash code of a record, and their arguments are explained in comments in the source code. `OE15N` allows the user to call the subroutine using alternative input and output subroutines.

3 GENERAL INFORMATION

Use of common: Common areas `OE15G/H/I/J/K/L/M` are used.

Other routines called directly: `OE15B/C/D/E/F/N.`

Input/output: Input on streams `INF1` and `INF2` and output on streams `LP` and `LPD.`

Restrictions: `LREC`≤256.

4 METHOD

The input files are read just once into a special data structure designed for easy insertion, deletion and recognition of the correspondences using hash coding techniques. As many records as the work-array length permits are read and then if the top records correspond to each other or if either is not present in the stored part of the other file, then `OE15D` is called to record this as a correspondence, insertion or deletion and then the record or records are removed from the data structure. The structure is then refilled and the process repeated as many times as possible. Then a search is made for the longest possible set of records that exist in the stored parts of both files in exactly the same order, using the method of Hunt and McIlroy (Bell Laboratories Computing Science Technical Report 41, 1976) and Hunt and Szymanski (Comm. ACM 20 (1977), 350-353). This search is terminated if the inner loop is executed an unreasonable number of times (`LIMIT` in `COMMON/OE15K/`) or if storage for candidate correspondences (`MCAND` in `COMMON/OE15K/`) runs out. If such a termination occurs or if some of either file remains to be read into the data structure then the first half (or some other factor `SPLIT` in `COMMON/OE15K/`) of the correspondences are used to identify insertions, deletions and common records which are sent to `OE15D` (or one of its alternatives) for output and are then removed from the data structure. The space vacated is used for more records from the files and the same procedure is applied continually until the files are exhausted or the Hunt and Szymanski algorithm terminates naturally with no further records left to be read, in which case all the correspondences are used. The user may monitor the way the correspondences are split by setting `LPD` in `COMMON/OE15G/` to a genuine stream number.

For further details see ‘A Fortran subroutine for comparing two files’, AERE-R.8971, S. Marlow and J.K. Reid.

5 EXAMPLE OF USE

The following main program could be run to compare files of text, on units 12 an 13, where only the first 72 columns are relevant.

```
      CALL OE15A(12,13,72)
      STOP
      END
```

An example of normal (`SHORT=.TRUE.`) output is shown below.

```
   1    1  C######DATE   01 JAN 1984     COPYRIGHT UKAEA, HARWELL.
   2       C######ALIAS MC16A
   3             SUBROUTINE MC16A (A,N,COL)
        2  C######ALIAS MC16AD
        3        SUBROUTINE MC16AD(A,N,COL)
        4        DOUBLE PRECISION A,COL
   4    5        DIMENSION A(1),COL(1)
           ******    17 RECORDS COMMON TO BOTH FILES NOT LISTED ******
  22   23        J=J+1
  23             IF (A(JJ).GT.0.) GO TO 30
```

```
 24            A(IB)=0.
      24       IF (A(JJ).GT.0.D0)GO TO 30
      25       A(IB)=0.D0
 25   26       A(IS)=A(IS)+COL(N-J)**2
 26   27       GO TO 40
 27         30 A(IB)=COL(N-J)/SQRT(A(JJ))
      28    30 A(IB)=COL(N-J)/DSQRT(A(JJ))
 28   29    40 IB=IB-1
         ******     2 RECORDS COMMON TO BOTH FILES NOT LISTED ******
 31   32       END
```