

```

IDENTIFICATION DIVISION.
PROGRAM-ID.          OPPSCAL.
*AUTHOR.            ED FRANEY.
*REMARKS.           CMS.
*****
* 5/8/00 - ADD WINX TO RETURN RECORD *
* 5/10/00 - ADD MSA TO RETURN RECORD *
* 5/11/00 - CHANGE PROVIDER FILE FROM 9999 OCCURS TO 999 *
*          - CHANGE ALL LINE OCCURS FROM 999 TO 450 *
* 5/17/00 - RELOCATE MOVE OF H-WINX1 TO A-WINX TO *
*          0000-PROCESS-MAINLINE *
* 6/08/00 - ADD CODE IN 0550-CALC-STANDARD: *
*          MOVE W-MIN-COIN (W-LP-INDX) TO H-MIN-COIN *
*          - CORRECTED PROBLEM OF COINSURANCE ELECTION *
*          BEING DROPPED AFTER FIRST LINE WAS PROCESSED *
* 6/14/00 - UPDATED WAGE INDEX TABLE *
*          - CREATED NEW COPY BOOK *
* 7/10/00 - CORRECTED 0150-INIT PARAGRAPH *
*          - WILL INCLUDE ALL PACKAGED LINES IN TOTAL *
*          CHARGES FOR OUTLIER CALCULATION *
*          - INCLUDED OPPTS-PKG-FLAG = 0 OR 1 OR 2 *
* 7/18/00 - CORRECTED MAX-COINSURANCE PARAGRAPH 0550-CALC *
*          - CHANGED >> MOVE 776 TO H-NAT-COIN << TO *
*          - MOVE H-MAX-COIN TO H-NAT-COIN *
* 8/02/00 - CHANGED OPPTSAPCS COPYBOOK TO RANK NEW DEVICES *
*          LAST IN DEDUCTIBLE CALCULATION *
*          - OCE SERVICE INDICATOR = H *
*          - PAYMENT AND COINSURANCE = ZERO *
* 8/02/00 - CHANGED OPPTSWINX COPYBOOK TO INCLUDE NEW *
*          MARYLAND CODE *
*          - ' 80' = 008631 000000 *
* 8/03/00 - CHANGED 0300-COIN-DEDUCT *
*          - REMOVE CONDITION OF SERVICE INDICATOR *
*          NOT EQUAL 'T' *
*          - MOVE 1 TO DISCOUNT RATE *
*          - ALWAYS ACCEPT DISCOUNT FROM OCE *
* 8/07/00 - CHANGED 0150-INIT PARAGRAPH *
*          - ADD CONDITION OF OPPTS-PKG-FLAG NOT = ZERO *
*          - WILL NOT PAY PACKAGED PARTIAL *
*          HOSPITALIZATIONS FLAG = 1 OR 2 *
* 8/07/00 - CHANGED 0250-CALC-DISCOUNT *
*          - DISCOUNT INDICATOR OF '8' NOW EQUAL TO 2 *
*          - FORMULA NOW = 2 (DOUBLE) *
*          - FORMULA WAS = 2 / UNITS *
* 8/15/00 - CHANGED 0100-INIT *
*          - ALLOW PROPER PROCESSING OF SPECIAL WAGE *
*          INDEX CONSIDERATIONS *
*          - PERFORM 0220-CHNG-WAGEINDX FIRST *
*          - IF WAGE INDEX = 0 *
*          PERFORM 0200-CALC-WAGEINDX *
* 8/17/00 - CHANGED 0150-INIT *
*          - ALLOW THE PROCESSING OF PARTIAL HOSP. IF *
*          - LINE ITEM DENIAL/REJECT FLAG = 1 *
*          AND APC = 0033,0034,0322-0325,0373,0374 *
* 8/18/00 - CHANGED 0900-END-PRICE-RTN *
*          - PREVENT OUTLIER PROCESSING OF NON-OPPTS *
*          CLAIMS *

```

```

*           - IF TOTAL CLAIM PAYMENT = 0 *
*           DO NOT CALCULATE OUTLIER AMOUNT *
* 8/18/00 - ADD 0125-INIT PARAGRAPH *
*           - SET FLAG IF APC = 0033 ON CLAIM *
* 8/18/00 - CHANGE 0150-INIT *
*           - IF TYPE OF BILL INCLUSION = 0 (OLD) *
*           - (NEW) OR (APC 0033 IS ON THE CLAIM *
*             AND SERVICE INDICATOR = 'P' *
*             OR APC = 0322-0325,0373,0374) *
* 8/22/00 - CHANGE 0400-CALCULATE *
*           - IF PROVIDER REDUCED COINSURANCE IS GREATER *
*             THAN NATIONAL ADJUSTED COINSURANCE *
*           - MOVE NATIONAL COINSURANCE TO THE *
*             PROVIDER REDUCE COINSURANCE *
* 9/14/00 - CHANGE 0550-CALCULATE-STANDARD *
*           - CHANGE MINIMUM COINSURANCE CALCULATION *
*             FOR DRUGS AND DEVICES (SERVICE INDICATORS *
*             'G', 'H', OR 'J') *
* 9/14/00 - CHANGE APC AND WAGE INDEX LOOKUP ROUTINES. *
*           - LOGIC DID NOT ALLOW FOR MULTIPLE RELEASE *
*             DATES *
* 10/12/00- CONTROL THE PROCESSING OF PARAGRAPH 0220- *
*             CHNG-WAGEINDX BY USING THE SERVICE FROM *
*             DATE - ONLY PROCESS IF < 20010101 *
* 10/12/00- CONTROL THE PROCESSING OF PARAGRAPHS 0105 *
*             AND 0110 BY USING THE SERVICE FROM *
*             DATE - RESET FLOOR MSA *
* 10/12/00- CONTROL H-MAX-COIN USING THE SERVICE FROM *
*             DATE - IP MAX = 776 IF < 20010101 *
*             - IP MAX = 792 IF > 20001231 *
*             (PARAGRAPH 0550) *
* 10/12/00- LIMIT THE H-MAX-COIN FOR STATUS INDICATORS *
*             'G' 'J' 'K' (DRUGS) TO $792 PER LINE ITEM *
*             EFFECT 20010101 (PARAGRAPH 0550) *
* 10/12/00- ALLOW FOR A NEW SERVICE INDICATOR 'K' *
* 11/15/00- ADJUST THE COST-TO-CHARGE FOR 2001 BY A *
*             FACTOR OF .981956 *
*           - PARAGRAPHS 0555 AND 0910 *
* 11/16/00- CREATED WAGE INDEX COPY BOOK FOR 20010101 *
* 12/06/00- ALLOW FOR NEW SERVICE INDICATOR 'B' *
*           - NON-ALLOWED ITEM OR SERVICE FOR OPPTS *
* 12/07/00- INSERTED W-APC-ADJ-TABLE TO ADJUST APC *
*             FROM OCE *
*           - RESET SERVICE INDICATOR IF NECESSARY *
*           - RESET PAYMENT INDICATOR IF NECESSARY *
* 12/07/00- INSERTED PARAGRAPH 0160-ADJUST-APC *
* 12/28/00- CORRECTED WAGE INDEX LOOK-UP ROUTINE *
*           - PARAGRAPH 0210-WAGE-LOOKUP ( > INSTEAD OF < ) *
*           - CHECK RECLASS VALUE OF 'Y' ELSE ALLOW ANY *
*             OTHER VALUE FOR NON-RECLASS *
* 02/28/01- ADDED NEW APCS - 'C' CODES DEVICE PASSTHRUS *
*           - 125 CODES *
* 02/28/01- REMOVED OCE/APC PATCH - EFFECTIVE APR 01,2001 *
*           (0160-ADJUST-APC) *
* 03/05/01- SET RETURN CODE TO '30' TO BY-PASS THE PASS *
*           THRU PAYMENTS FOR THE FOLLOWING APCS: *
*           01111 - 01114, 01117, 06300, AND 06600 *

```

```

* 03/06/01- ADD PATCH FOR HCPCS CODE C1050 *
*          CHANGE SERVICE INDICATOR TO 'S' *
*          CHANGE PAYMENT APC TO '0976' *
* 03/07/01- ADDED NEW APCS - 'C' CODES DEVICE PASSTHRUS *
*          - 25 CODES *
* 05/02/01- ADD CODE FOR DAILY COINSURANCE LIMITATION *
*          - UPDATE APC TABLE FOR 7/1/2001 *
*          - NEW PROCESS FOR "DELETED" APC CODES *
* 06/12/01- CORRECT TRUNCATION OF APC RANKING FACTOR *
*          - INCREASE THE SIZE OF THE RANKING VARIABLE *
*          - 9(03) TO 9(05) *
* 07/23/01- ADD NEW PROVIDER SPECIFIC WAGE INDEX LOGIC *
*          - PARAGRAPH 0225-CHNG-WAGEINDX *
* 07/23/01- UPDATE APC TABLE *
* 08/20/01- REMOVE REFERENCE TO L-PSF-GEO-MSA FROM LOGIC *
*          - IN PARAGRAPH 0225-CHNG-WAGEINDX *
* 12/19/01- ADD INPATIENT LIMIT LOGIC FOR CY 2002 *
*          - IN PARAGRAPH 0100-INIT *
* 01/16/02- CREATE COPYBOOK FOR FY 2002 WAGE INDEX *
*          - NEW BASEWINX (EFF. 04/01/2002) *
* 01/16/02- ADD APC 0339 SERVICE UNITS OVERRIDE *
*          - UNITS = 1 (EFF. 04/01/2002) *
* 01/17/02- ADD SECTION 401 AND FLOOR MSA DESIGNATIONS *
*          - PARAGRAPHS 0115-FLOOR-2002 AND *
*            0115-SEC401-2002 (EFF. 04/01/2002) *
* 01/17/02- ADD LOGIC TO PROCESS LINE LEVEL OUTLIER *
*          PAYMENT *
* 01/30/02- ADD NEW COPYBOOK FOR DEVICE OFFSET PROCESS *
*          - SERVICE INDICATOR TYPE H *
*          - EFFECTIVE 04/01/2002 *
*          - TOTAL AND WAGE ADJUST OFFSET AMOUNT AND *
*            SUBTRACT PROPORTIONATELY FROM ANY SERVICE *
*            INDICATOR TYPE 'H' THAT HAVE HCPCS CODE *
*            BEGINNING WITH 'C' (C1713 - C263) *
* 01/31/02- ADD LOGIC FOR PRO RATA REDUCTION FOR ALL *
*          SERVICE INDICATOR TYPES G AND H *
*          - CURRENTLY .689 *
*          - EFFECTIVE 04/01/2002 *
* 02/27/02- CHANGE PRO RATA REDUCTION TO .634 *
* 02/27/02- CHANGE LINE ITEM CALCULATION *
*          FROM 3.0 * LINE PYMT TO 3.5 * LINE PYMT *
* 02/28/02- UPDATE APC RATE TABLE *
* 02/28/02- UPDATE DEVICE OFFSET TABLE *
* 04/24/02- UPDATE APC TABLE *
*          - APC 00034 FOR 20020401 *
* 04/24/02- MOVED LOGIC TO CALCULATE DISCOUNT RATE *
*          BEFORE CALCULATING TOTAL OFFSET AMOUNT *
*          - PERFORM 1250-CALC-DISCOUNT *
* 04/24/02- ADDED DISCOUNTING TO OFFSET AMOUNT *
*          - PARAGRAPH 1160-TOTAL-OFFSET *
* 04/24/02- PARAGRAPH 1150-INIT *
*          - COMPUTE H-TOT-N-CHRG WHEN PACKAGE INDICATOR *
*            = '1' OR '2' *
* 07/23/02- MOVED LOGIC TO RESET SERVICE UNITS TO 1 IF *
*          APC = 0339 BEFORE CALCULATING DISCOUNT *
*          FRACTION *
*          - PERFORM 1150-INIT *

```

```

* 10/31/02- REMOVED SERVICE INDICATOR 'S' FROM 2500-ADJ- *
*          CHRGS LOGIC *
* 12/02/02- ADDED 2180-MOD-CCODE-PYMT TO ADJUST PYMT FOR *
*          C9114 AND C9115 BETWEEN 12/31/2002 AND *
*          04/01/2003 *
* 12/02/02- ADD NEW APC RATE TABLE *
* 02/10/03- ADD NEW PROCESS FOR CALCULATING BLOOD *
*          DEDUCTIBLES *
*          - 2375-BLOOD-DEDUCT *
* 02/10/03- ALLOW FOR NEW INPUT FIELDS TO BE PASSED TO *
*          OPPSCAL *
*          -CLAIM LEVEL: *
*             01 BENE-BLOOD-PINTS          PIC 9(01). *
*          -LINE LEVEL: *
*             05 A-BLOOD-PINTS-USED        PIC 9(01). *
*             05 A-BLOOD-DEDUCT-DUE        PIC 9(05)V9(02). *
* 03/03/03- UPDATE APC TABLE FOR: *
*          APCS: 1348 1607 1814 9111 9202 9203 9204 *
* 04/11/03- CORRECT BLOOD DEDUCTIBLE PROCESS *
* 07/21/03- INCLUDE OVR '4' LOGIC IN PARAGRAPH 2400 *
*          BACK TO 8/1/2000 LOGIC *
* 10/21/03- INCREASED IP DEDUCTIBLE AMOUNT TO $876 *
*          FOR CALENDAR YEAR 2004 *
* 10/30/03- ADD PARAGRAPHS FOR MSA FLOOR AND SECTION 401 *
*          HOSPITALS FOR CALENDAR YEAR 2004 *
*          - 2120-FLOOR-2004 *
*          - 2120-SEC401-2004 *
* 03/02/04- ADD LOGIC IN 2600-ADJ-CHRG-OUTL TO INCLUDE *
*          SPECIFIED "K" INDICATORS IN OUTLIER *
*          CALCULATIO - FOR 04/01/2004 *
* 03/03/04- INCLUDE NEW APC UPDATES FOR 04/01/2004 AND *
*          RETROACTIVE RATES *
* 03/04/04- NEW LOGIC FOR SPECIFIED "H" INDICATORS TO *
*          ALTER COINSURANCE AMOUNT IN 2550-CALC-STANDARD *
* 04/08/04- REMOVE OVERRIDE LOGIC FROM 2150-INIT *
*          ALLOW OCE STATUS INDICATOR FOR A9526 & Q4078 *
*****

```

DATE-COMPILED.

ENVIRONMENT DIVISION.

CONFIGURATION SECTION.

SOURCE-COMPUTER. IBM-370.

OBJECT-COMPUTER. IBM-370.

INPUT-OUTPUT SECTION.

FILE-CONTROL.

DATA DIVISION.

FILE SECTION.

WORKING-STORAGE SECTION.

```

01  W-STORAGE-REF          PIC X(46)  VALUE
    'OPCAL2004.4  - W O R K I N G  S T O R A G E'.
01  CAL-VERSION0          PIC X(07)  VALUE 'C2002.0'.
01  CAL-VERSION1          PIC X(07)  VALUE 'C2002.3'.
01  CAL-VERSION2          PIC X(07)  VALUE 'C2003.1'.
01  CAL-VERSION3          PIC X(07)  VALUE 'C2004.4'.
01  R1                    PIC S9(04) COMP SYNC.
01  R2                    PIC S9(04) COMP SYNC.

```

01 R3 PIC S9(04) COMP SYNC.  
01 R4 PIC S9(04) COMP SYNC.

\*\*\*\*\*  
\* LAYUP TABLE AREA FOR CY2002 APC OFFSET ADJUSTMENTS \*  
\* OPPSOF02 - EFFECTIVE AS OF 04-01-2002 \*  
\* OPPSOF03 - EFFECTIVE AS OF 01-01-2003 \*  
\*\*\*\*\*  
\*\*\*\*\* WOO-INDX \*\*\*\*\*  
COPY OPPSOF02.

\*\*\*\*\* WOO-INDX2 \*\*\*\*\*  
COPY OPPSOF03.

\*\*\*\*\* WOO-INDX3 \*\*\*\*\*  
COPY OPPSOF04.

\*\*\*\*\*  
\* LAYUP TABLE AREA FOR CY2000 APC'S \*  
\*\*\*\*\*  
\*\*\*\*\* WAA-INDX \*\*\*\*\*  
COPY OPPSAPCS.

\*\*\*\*\*  
\* THIS IS THE WAGE-INDEX \*  
\* ASSOCIATED WITH THE BILL BEING PROCESSED \*  
\*\*\*\*\*  
\*\*\*\*\* WWM-INDX \*\*\*\*\*  
COPY OPPSWINX.

\*\*\*\*\*  
\* THIS IS THE BLOOD DEDUCTABLE \*  
\* RANKING TABLE FOR THE BILL BEING PROCESSED \*  
\*\*\*\*\*  
\*\*\*\*\* W-BD-INDX \*\*\*\*\*

01 W-BLOOD-APC-FILLS.  
03 PIC X(42) VALUE  
'P902101P901002P903803P901604C101005C101806'.  
03 PIC X(42) VALUE  
'P902207P903908P904009C102110C101611C102012'.  
01 W-BLOOD-APC-TABLE REDEFINES W-BLOOD-APC-FILLS.  
03 WBD-ENTRY OCCURS 12 TIMES  
INDEXED BY WBD-INDX.  
05 W-BLOOD-HCPCS PIC X(05).  
05 W-BLOOD-RANK PIC 9(02).

\*\*\*\*\*  
\* THIS IS THE MAX-COINSURANCE \*  
\* DETERMINED BY DATE LINE-ITEM PROCESSING \*  
\*\*\*\*\*  
\*\*\*\*\* WMC-INDX \*\*\*\*\*

01 W-MAX-COIN-DATE-FILLS.  
03 PIC X(44) VALUE  
'20000801001200101010022002010100320030101004'.

```

01 W-MAX-COIN-DATE-TABLE REDEFINES W-MAX-COIN-DATE-FILLS.
03 WMC-ENTRY OCCURS 4 TIMES.
    05 WMC-DATE          PIC X(8).
    05 WMC-DTCD          PIC 9(3).

```

```

01 WORK-AREA.
05 H-SUB                PIC S9(07) COMP-3 VALUE ZERO.
05 W-SUB1               PIC S9(07) COMP-3 VALUE ZERO.
05 W-SUB2               PIC S9(07) COMP-3 VALUE ZERO.
05 PS-SUB               PIC S9(07) COMP-3 VALUE ZERO.
05 LN-SUB               PIC S9(07) COMP-3 VALUE ZERO.
05 DISC-FRACTION        PIC 9V9(03) VALUE .500.
05 TERM-PROC-DISC       PIC 9V9(03) VALUE .500.
05 APC33-FLAG           PIC X(01).
05 GJK-FLAG             PIC X(01).
05 ST0-FLAG             PIC X(01).
05 N-FLAG               PIC X(01).
05 C-FLAG               PIC X(01).
05 T-LITEM-PYMT         PIC S9(07)V9(02).
05 W-OFF-APC            PIC X(05).

```

```

*****
*   BELOW ARE THE VARIABLES THAT WILL BE HELD FOR CLAIM           *
*   LEVEL PROCESSING (OUTLIER/DEDUCTIBLE/COINSURANCE)           *
*****

```

```

01 H-ADDITIONAL-VARIABLES.
05 H-OUTLIER-PYMT       PIC S9(07)V9(02).
05 H-PRTL-HOSP-PYMT     PIC 9(07)V9(02).
05 H-TOTAL-CLM-DEDUCT   PIC 9(03)V9(02).
05 H-TOTAL-OFFSET       PIC 9(07)V9(02).
05 H-TOTAL-WAOFF        PIC 9(07)V9(02).
05 H-TOT-CHRG           PIC 9(07)V9(02).
05 H-TOT-PYMT           PIC 9(07)V9(02).
05 H-BENE-DEDUCT        PIC 9(03)V9(02).
05 H-MAX-COIN           PIC 9(05)V9(02).
05 H-IP-LIMIT           PIC 9(05)V9(02).
05 H-NEW-COIN           PIC 9(05)V9(02).
05 H-NEW-WGNAT          PIC 9(05)V9(02).
05 H-BLOOD-DEDUCT-DUE   PIC 9(05)V9(02).
05 H-TOT-ST-CHRG        PIC 9(08)V99.
05 H-TOT-N-CHRG         PIC 9(08)V99.
05 H-TOT-H-CHRG         PIC 9(08)V99.
05 H-TOT-ST-PYMT        PIC 9(08)V99.
05 H-TOT-STVX-PYMT      PIC 9(08)V99.
05 H-TOT-HTD-UNITS      PIC 9(09).
05 H-TOT-OFF-UNITS      PIC 9(09).
05 H-BENE-BLOOD-PINTS   PIC 9(01).
05 H-BENE-PINTS-USED    PIC 9(01).
05 LINE-HOLD-ITEMS.
10 H-COIN-PERCENT       PIC 9(01)V9(04).
10 H-LITEM-PYMT         PIC S9(07)V9(02).
10 H-LITEM-OUTL-PYMT    PIC S9(07)V9(02).
10 H-LITEM-REIM         PIC 9(07)V9(02).
10 H-APC-PYMT           PIC 9(07)V9(02).
10 H-TOTAL-LN-DEDUCT    PIC 9(03)V9(02).
10 H-LN-BLOOD-DEDUCT    PIC 9(05)V9(02).
10 H-LN-BLD-PYMT        PIC 9(05)V9(02).

```

10	H-NAT-COIN	PIC 9(05)V9(02).
10	H-MIN-COIN	PIC 9(05)V9(02).
10	H-PSF-COIN	PIC 9(05)V9(02).
10	H-RED-COIN	PIC 9(05)V9(02).
10	H-RATIO	PIC S9(03)V9(07).
10	H-SHIFT	PIC 9(05)V9(02).
10	H-TOTAL	PIC 9(05)V9(02).
10	H-OUTLIER-FACTOR	PIC 9(01)V9(02).
10	H-OUTLIER-PCT	PIC 9(01)V9(02).
10	H-LN-PTR	PIC 9(03).
10	H-SRVC-UNITS	PIC 9(07).
10	H-RANK	PIC 9(05).
10	H-BLOOD-RANK	PIC 9(02).
10	H-PSF-MSA	PIC X(04).
10	H-DCP-STAGE.	
15	H-DCP-DOS	PIC 9(08).
15	H-DCP-CODE	PIC 9(01).
10	H-PPCT	PIC S9V9(06) COMP-3.
10	H-DISC-RATE	PIC S9V9(08) COMP-3.
10	H-BLOOD-FRACTION	PIC S9V9(08) COMP-3.
10	H-WINX1	PIC S9V9(04) COMP-3.
10	H-SUB-CHRG	PIC 9(08)V99.
10	H-CHRG-RATE	PIC 9(01)V9(8).
10	H-OFF-RATE	PIC 9(01)V9(8).

```
*****
*   BELOW IS THE LAY-UP TABLE TO PROCESS DEDUCTIBLES.   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*****
***** W-LP-INDX *****
```

01	W-LNC-MAX	PIC S9(07) COMP-3 VALUE +0.
01	W-LINE-PTR-TABLE.	
05	W-LP-ENTRY	
	OCCURS 0 TO 450 TIMES	
	DEPENDING ON W-LNC-MAX	
	ASCENDING KEY IS W-RANK	
	INDEXED BY W-LP-INDX.	
10	W-LP-SUB	PIC S9(07) COMP-3.
10	W-APC-PYMT	PIC 9(07)V99.
10	W-NAT-COIN	PIC 9(05)V99.
10	W-MIN-COIN	PIC 9(05)V99.
10	W-RED-COIN	PIC 9(04)V99.
10	W-DISC-RATE	PIC 9(01)V9(08).
10	W-SRVC-UNITS	PIC 9(07).
10	W-RANK	PIC 9(05).
10	W-PPCT	PIC S9V9(06) COMP-3.
10	W-WINX1	PIC S9V9(04) COMP-3.
10	W-SUB-CHRG	PIC 9(08)V99.

```
*****
*   BELOW IS THE LAY-UP TABLE TO PROCESS BLOOD DEDUCTIBLES   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*****
***** W-BD-INDX *****
```

01	W-BLD-MAX	PIC S9(07) COMP-3 VALUE +0.
----	-----------	-----------------------------

```

01 W-BLOOD-PTR-TABLE.
  05 W-BD-ENTRY
      OCCURS 0 TO 450 TIMES
      DEPENDING ON W-BLD-MAX
      INDEXED BY W-BD-INDX.
    10 W-BD-SUB                PIC S9(07)  COMP-3.
    10 W-BD-APC-PYMT          PIC 9(07)V99.
    10 W-BD-NAT-COIN          PIC 9(05)V99.
    10 W-BD-MIN-COIN          PIC 9(05)V99.
    10 W-BD-RED-COIN          PIC 9(04)V99.
    10 W-BD-DISC-RATE         PIC 9(01)V9(08).
    10 W-BD-SRVC-UNITS        PIC 9(07).
    10 W-BD-RANK              PIC 9(05).
    10 W-BD-PPCT              PIC S9V9(06) COMP-3.
    10 W-BD-WINX1             PIC S9V9(04) COMP-3.
    10 W-BD-SUB-CHRG          PIC 9(08)V99.

```

```

*****
*   BELOW IS THE LAY-UP TABLE TO PROCESS DRUG COINSURANCE   *
*   ROLL-UP                                                    *
*****
***** W-DCP-INDX *****

```

```

01 W-DCP-MAX                PIC S9(07)  COMP-3 VALUE +0.
01 W-DOS-COIN-PTR-TABLE.
  05 W-DCP-ENTRY
      OCCURS 0 TO 450 TIMES
      DEPENDING ON W-DCP-MAX
      ASCENDING KEY IS W-DCP-STAGE
      INDEXED BY W-DCP-INDX.
    10 W-DCP-SUB                PIC S9(07).
    10 W-DCP-STAGE.
      15 W-DCP-DOS              PIC 9(08).
      15 W-DCP-CODE            PIC 9(01).
    10 W-DCP-SRVC-IND          PIC X(01).
    10 W-DCP-COIN1            PIC 9(05)V99.
    10 W-DCP-COIN2            PIC 9(05)V99.
    10 W-DCP-WGNAT            PIC 9(05)V99.

```

LINKAGE SECTION.

```

*****
*   LAYUP TABLE AREA FOR CY2000 PROVIDER SPECIFIC RECORD   *
*****

```

```

01 L-PROV-SPEC-AREA.
  05 L-PSF-NPI                PIC X(08).
  05 L-PSF-NPI-FILLER         PIC X(02).
  05 L-PSF-PROV-OSCAR.
    10 L-PSF-PROV-ST          PIC X(02).
    10 L-PSF-PROV-3456        PIC X(04).
  05 L-PSF-EFFDT              PIC 9(08).
  05 L-PSF-FY-BEGIN-DT        PIC 9(08).
  05 L-PSF-REPORT-DT          PIC 9(08).
  05 L-PSF-TERMDT             PIC 9(08).
  05 L-PSF-WAIVE-IND          PIC X(01).

```



```

05 L-PSF-FI-NUM PIC 9(05).
05 L-PSF-PROV-TYPE PIC X(02).
05 L-PSF-SPCL-LOCATION-IND PIC X(01).
05 L-PSF-WGIDX-RECLASS PIC X(01).
05 L-PSF-GEO-MSA PIC X(04).
05 L-PSF-WI-MSA PIC X(04).
05 L-PSF-COLA PIC 9V9(03).
05 L-PSF-BED-SIZE PIC 9(05).
05 L-PSF-OPCOST-RATIO PIC 9V9(03).
05 FILLER PIC X(17).
05 L-PSF-APC-LINE-CNT PIC 9(04).
05 L-PSF-APC-TABLE OCCURS 999 TIMES
    DEPENDING ON L-PSF-APC-LINE-CNT.
    10 L-PSF-APC PIC X(04).
    10 L-PSF-RED-COIN PIC 9(04)V99.

```

```

*****
* INPUT RECORD FROM THE OCE/STANDARD SYSTEM *
* *
*****

```

```

01 OPPTS-LINE-CNT PIC 9(08) COMP.
01 OCE-DATA.
    05 OPPTS-OCE-LINE OCCURS 450 TIMES
        DEPENDING ON OPPTS-LINE-CNT.
    10 OPPTS-HCPCS.
        15 OPPTS-ALPHA PIC X(01).
        15 FILLER PIC X(04).
    10 OPPTS-GRP.
        15 FILLER PIC X(01).
        15 OPPTS-APC PIC X(04).
    10 OPPTS-HCPCS-APC PIC X(05).
    10 OPPTS-SRVC-IND PIC X(01).
    10 OPPTS-PYMT-IND PIC X(01).
    10 OPPTS-DISC-FACT PIC 9(01).
    10 OPPTS-LITEM-DR-FLAG PIC X(01).
    10 OPPTS-PKG-FLAG PIC X(01).
    10 OPPTS-PYMT-ADJ-FLAG PIC X(01).
    10 OPPTS-SITE-SRVC-FLAG PIC X(01).
    10 OPPTS-SRVC-UNITS PIC 9(07).
    10 OPPTS-SUB-CHRG PIC 9(08)V99.
    10 OPPTS-LITEM-ACT-FLAG PIC X(01).
01 L-SERVICE-FROM-DATE PIC 9(08).
01 BENE-DEDUCT PIC 9(03)V9(02).
01 BENE-BLOOD-PINTS PIC 9(01).

```

```

*****
* BELOW ARE THE VARIABLES THAT WILL BE PASSED BACK *
* TO SS ASSOCIATED WITH THE BILL BEING PROCESSED *
* - EFF. 04/01/2002 CALCULATE LINE ITEM OUTIER PAYMENT *
*****

```

```

01 A-ADDITIONAL-VARIABLES.
    05 A-CALC-VERS PIC X(07).
    05 A-TOTAL-CLM-DEDUCT PIC 9(03)V9(02).
    05 A-OUTLIER-PYMT PIC 9(07)V9(02).
    05 A-TOT-CLM-PYMT PIC 9(07)V9(02).
    05 A-TOT-CLM-CHRG PIC 9(07)V9(02).
    05 A-CLM-RTN-CODE PIC 9(02).

```

```

05  A-MSA                PIC X(04).
05  A-WINX                PIC S9V9(04).
05  A-BLOOD-PINTS-USED   PIC 9(01).
05  A-BLOOD-DEDUCT-DUE   PIC 9(05)V9(02).
05  A-LINE-ITEMS OCCURS 450 TIMES
      DEPENDING ON OPPTS-LINE-CNT.
    10  A-LITEM-PYMT       PIC 9(07)V9(02).
    10  A-LITEM-REIM       PIC 9(07)V9(02).
    10  A-TOTAL-LN-DEDUCT  PIC 9(03)V9(02).
    10  A-ADJ-COIN        PIC 9(05)V9(02).
    10  A-RED-COIN        PIC 9(05)V9(02).
    10  A-BLOOD-LN-DEDUCT  PIC 9(05)V9(02).
    10  A-RETURN-CODE     PIC 9(02).

```

```
01  OCE-IN-DATE.
```

```

05  OCE-IN-LINES OCCURS 450 TIMES.
    10  FILLER             PIC X(15).
    10  OPPTS-LITEM-DOS    PIC 9(08).
    10  FILLER             PIC X(21).

```

```

*****
*   PROCESSING: PRIOR TO 20020401                               *
*   A. WILL PROCESS LINE ITEMS BASED ON OCE FLAGS.             *
*   B. INITIALIZE OPPTS HOLD VARIABLES.                        *
*   C. EDIT THE DATA PASSED FROM THE OCE.                     *
*   D. ASSEMBLE PRICING COMPONENTS.                            *
*   E. CALCULATE THE PRICE.                                     *
*   F. RETURN COINSURANCE/REIMBURSEMENT/DEDUCTIBLE/           *
*       PAYMENT/OUTLIER AMOUNT/RETURN CODES                     *
*****

```

```

PROCEDURE DIVISION USING OPPTS-LINE-CNT
                        OCE-DATA
                        A-ADDITIONAL-VARIABLES
                        L-PROV-SPEC-AREA
                        L-SERVICE-FROM-DATE
                        BENE-DEDUCT
                        BENE-BLOOD-PINTS
                        OCE-IN-DATE.

```

```
0000-DATE-CONTROL.
```

```

IF L-SERVICE-FROM-DATE > 20021231
    PERFORM 2000-PROCESS-MAIN-NEW
        THRU 2000-PROCESS-MAIN-NEW-EXIT
ELSE
    IF L-SERVICE-FROM-DATE > 20020331
        PERFORM 1000-PROCESS-MAIN-NEW
            THRU 1000-PROCESS-MAIN-NEW-EXIT
    ELSE
        PERFORM 0000-PROCESS-MAIN-OLD
            THRU 0000-PROCESS-MAIN-OLD-EXIT.

```

```
GOBACK.
```

```
0000-PROCESS-MAIN-OLD.
```

```

PERFORM 0100-INIT
    THRU 0100-INIT-EXIT.

```

```

IF A-CLM-RTN-CODE >= 50
  GOBACK.
MOVE H-WINX1 TO A-WINX.

PERFORM 0125-INIT
  THRU 0125-INIT-EXIT
  VARYING LN-SUB FROM 1 BY 1
  UNTIL LN-SUB > OPPTS-LINE-CNT.

MOVE 0 TO W-DCP-MAX W-BLD-MAX.
PERFORM 0150-INIT
  THRU 0150-INIT-EXIT
  VARYING LN-SUB FROM 1 BY 1
  UNTIL LN-SUB > OPPTS-LINE-CNT.

SET W-BD-INDX TO 1.
MOVE 0 TO W-DCP-MAX.
PERFORM 0400-CALCULATE
  THRU 0400-CALCULATE-EXIT
  VARYING W-LP-INDX FROM 1 BY 1
  UNTIL W-LP-INDX > W-LNC-MAX.

IF GJK-FLAG = 'Y'
  PERFORM 0800-ADJ-STV-REIM
  THRU 0800-ADJ-STV-REIM-EXIT
  VARYING W-DCP-INDX FROM 1 BY 1
  UNTIL W-DCP-INDX > W-DCP-MAX
ELSE
  NEXT SENTENCE.

PERFORM 0900-END-PRICE-RTN
  THRU 0900-END-PRICE-RTN-EXIT.

0000-PROCESS-MAIN-OLD-EXIT.
EXIT.

```

```

*****
*   INITIALIZE WORKING STORAGE HOLD AREAS                               *
*   AND ADDITIONAL VARIABLES TO BE PASSED BACK TO                       *
*   THE STANDARD SYSTEM.                                               *
*   - IF PROVIDER SPECIFIC FILE WAGE INDEX RECLASSIFICATION           *
*   CODE INVALID OR MISSING                                           *
*     - MOVE '52' TO CLAIM LEVEL RETURN CODE                          *
*     - DISCONTINUE CLAIM PROCESSING                                   *
*   - IF SERVICE FROM DATE NOT NUMERIC OR < 20000801                 *
*     - MOVE '53' TO CLAIM LEVEL RETURN CODE                          *
*     - DISCONTINUE CLAIM PROCESSING                                   *
*   - IF SERVICE FROM DATE < PROVIDER EFFECTIVE DATE                 *
*     - MOVE '54' TO CLAIM LEVEL RETURN CODE                          *
*     - DISCONTINUE CLAIM PROCESSING                                   *
*   - IF SERVICE FROM DATE > PROVIDER TERMINATION DATE               *
*     - MOVE '54' TO CLAIM LEVEL RETURN CODE                          *
*     - DISCONTINUE CLAIM PROCESSING                                   *
*****

```

0100-INIT.

MOVE 01 TO A-CLM-RTN-CODE.

```

MOVE 'N' TO APC33-FLAG GJK-FLAG.
IF L-SERVICE-FROM-DATE NOT NUMERIC
  MOVE 53 TO A-CLM-RTN-CODE
  GO TO 0100-INIT-EXIT
ELSE
  IF L-SERVICE-FROM-DATE < 20000801
    MOVE 53 TO A-CLM-RTN-CODE
    GO TO 0100-INIT-EXIT
  ELSE
    IF L-SERVICE-FROM-DATE < L-PSF-EFFDT
      MOVE 54 TO A-CLM-RTN-CODE
      GO TO 0100-INIT-EXIT
    ELSE
      IF L-PSF-TERMDT > 0
        IF L-SERVICE-FROM-DATE > L-PSF-TERMDT
          MOVE 54 TO A-CLM-RTN-CODE
          GO TO 0100-INIT-EXIT
        END-IF
      END-IF.
MOVE SPACE TO A-MSA.
MOVE ZERO TO A-OUTLIER-PYMT
  A-TOTAL-CLM-DEDUCT
  A-TOT-CLM-CHRG
  A-TOT-CLM-PYMT
  A-BLOOD-PINTS-USED
  A-BLOOD-DEDUCT-DUE
  A-WINX
  W-LNC-MAX.
INITIALIZE H-ADDITIONAL-VARIABLES.
INITIALIZE LINE-HOLD-ITEMS.
INITIALIZE T-LITEM-PYMT.
MOVE CAL-VERSION0 TO A-CALC-VERS.
MOVE BENE-DEDUCT TO H-BENE-DEDUCT.
MOVE BENE-BLOOD-PINTS TO H-BENE-BLOOD-PINTS
  H-BENE-PINTS-USED.
MOVE WAD-MAX TO WAD-SUB.
PERFORM UNTIL L-SERVICE-FROM-DATE
  NOT < WAD-DATE (WAD-SUB)
  SUBTRACT 1 FROM WAD-SUB
END-PERFORM.
IF L-PSF-WGIDX-RECLASS = 'Y'
  MOVE L-PSF-WI-MSA TO H-PSF-MSA
ELSE
  IF L-PSF-WGIDX-RECLASS = 'N'
    MOVE L-PSF-GEO-MSA TO H-PSF-MSA
  ELSE
    MOVE 52 TO A-CLM-RTN-CODE
    GO TO 0100-INIT-EXIT.

IF L-SERVICE-FROM-DATE >= 20020101
  MOVE 812 TO H-IP-LIMIT
ELSE
  IF L-SERVICE-FROM-DATE >= 20010101
    MOVE 792 TO H-IP-LIMIT
  ELSE
    IF L-SERVICE-FROM-DATE >= 20000801
      MOVE 776 TO H-IP-LIMIT.

```

```

IF L-SERVICE-FROM-DATE < 20010101
  PERFORM 0105-FLOOR-2000
  THRU 0105-FLOOR-2000-EXIT.

IF L-SERVICE-FROM-DATE > 20001231
  PERFORM 0110-FLOOR-2001
  THRU 0110-FLOOR-2001-EXIT.

MOVE H-PSF-MSA TO A-MSA.

IF L-SERVICE-FROM-DATE >= 20010101
  PERFORM 0225-CHNG-WAGEINDX
  THRU 0225-CHNG-WAGEINDX-EXIT
ELSE
  PERFORM 0220-CHNG-WAGEINDX
  THRU 0220-CHNG-WAGEINDX-EXIT.

IF H-WINX1 = 0 THEN
  PERFORM 0200-CALC-WAGEINDX
  THRU 0200-CALC-WAGEINDX-EXIT.

```

```

0100-INIT-EXIT.
EXIT.

```

```

*****
* RESET FLOOR MSA - 'FROM-DATE' CONTROLLED *
* - YEAR 2000 *
* - YEAR 2001 *
*****

```

```

0105-FLOOR-2000.

```

```

IF (H-PSF-MSA = '6020' OR '9000' OR '8080') AND
  (L-PSF-PROV-ST = '36')
  MOVE ' 36' TO H-PSF-MSA
ELSE
  IF (H-PSF-MSA = '2440') AND
    (L-PSF-PROV-ST = '15')
    MOVE ' 15' TO H-PSF-MSA
  ELSE
    IF (H-PSF-MSA = '2520') AND
      (L-PSF-PROV-ST = '24') AND
      (L-PSF-WGIDX-RECLASS = 'Y')
      MOVE ' 24' TO H-PSF-MSA
    ELSE
      IF (H-PSF-MSA = '1123') AND
        (L-PSF-PROV-ST = '22')
        MOVE ' 22' TO H-PSF-MSA.

```

```

0105-FLOOR-2000-EXIT.
EXIT.

```

```

0110-FLOOR-2001.

```

```

IF (H-PSF-MSA = '6020' OR '9000' OR '8080') AND

```

```

(L-PSF-PROV-ST = '36')
MOVE ' 36' TO H-PSF-MSA
ELSE
IF (H-PSF-MSA = '2440') AND
(L-PSF-PROV-ST = '15')
MOVE ' 15' TO H-PSF-MSA
ELSE
IF (H-PSF-MSA = '9000') AND
(L-PSF-PROV-ST = '51')
MOVE ' 51' TO H-PSF-MSA
ELSE
IF (H-PSF-MSA = '1900') AND
(L-PSF-PROV-ST = '21')
MOVE ' 21' TO H-PSF-MSA
ELSE
IF (H-PSF-MSA = '1123') AND
(L-PSF-PROV-ST = '22')
MOVE ' 22' TO H-PSF-MSA.

```

```

0110-FLOOR-2001-EXIT.
EXIT.

```

```

*****
* SET FLAG IF APC = 0033 *
* - TERMINATE PROCESS IF 0033 LOCATED *
* *
*****

```

```

0125-INIT.

```

```

IF OPSS-APC (LN-SUB) = '0033'
MOVE 'Y' TO APC33-FLAG
MOVE 451 TO LN-SUB.

```

```

0125-INIT-EXIT.
EXIT.

```

```

*****
* 1. VERIFY THE SERVICE INDICATOR PASSED BY THE OCE *
* - IF INVALID SET RETURN CODE TO '40' *
* - DISCONTINUE LINE PROCESSING *
* - IF VALID SET RETURN CODE TO '01' *
* - CONTINUE LINE PROCESSING *
* 2. PROCESS LINES AND LOAD WORK TABLE ACCORDING TO PRICE *
* RANKING *
* 3. CHECK OCE EDIT INDICATORS AND SET RETURN CODES IF *
* INDICATORS ARE INVALID. *
* - VALID RETURN CODES FOR EDIT INDICATORS *
* - '41' - SERVICE INDICATOR INVALID FOR OPSS PRICER *
* - '42' - APC = '00000' OR (PACKAGING FLAG = 1 OR 2) *
* - '43' - PAYMENT INDICATOR NOT = TO 1 OR 5 THRU 9 *
* - '44' - SERVICE INDICATOR = 'H' BUT PAYMENT *
* INDICATOR NOT = TO 6 *
* - '45' - PACKAGING FLAG NOT = TO 0 *
* - '46' - (LINE ITEM DENIAL/REJECT FLAG NOT = TO 0 *
* OR LINE ITEM DENIAL/REJECT FLAG = TO 1 *
* AND (APC NOT = 0033 OR 0034 OR 0322 OR *

```

```

*           0323 OR 0324 OR 0325 OR 0373 OR 0374)) *
*           OR LINE ITEM ACTION FLAG NOT = TO 1 *
* - '47' - LINE ITEM ACTION FLAG = 2 OR 3 *
* - '48' - PAYMENT ADJUSTMENT FLAG NOT VALID *
* - '49' - SITE OF SERVICE FLAG NOT = TO 0 *
*           - OR (APC 0033 IS NOT ON THE CLAIM *
*             AND SERVICE INDICATOR = 'P' *
*             OR APC = 0322-0325,0373,0374) *
* 4. IF OCE INDICATORS ARE VALID, SEARCH APC TABLE *
*   - IF MISSING, DELETED OR INVALID APC *
*   - SET RETURN CODE TO '30' *
*   - DISCONTINUE LINE PROCESSING *
*****

```

0150-INIT.

```

*****
* - DISABLED AS OF 04-01-2001 *
*   PERFORM 0160-ADJUST-APC *
*   THRU 0160-ADJUST-APC-EXIT. *
*****
*   DISABLED AS OF 07-01-2001 *
*****
*   IF (L-SERVICE-FROM-DATE > 20010331) AND *
*     (OPPS-APC (LN-SUB) = '1410') AND *
*     (OPPS-HCPCS (LN-SUB) = 'C1050') *
*     MOVE 'S' TO OPPS-SRVC-IND (LN-SUB) *
*     MOVE '0976' TO OPPS-APC (LN-SUB). *
*****

```

```

SET W-BD-INDX TO LN-SUB.
SET W-LP-INDX TO LN-SUB.
INITIALIZE W-LP-ENTRY (W-LP-INDX).
INITIALIZE W-BD-ENTRY (W-BD-INDX).
MOVE ZERO TO A-LITEM-PYMT (LN-SUB)
             A-LITEM-REIM (LN-SUB)
             A-ADJ-COIN (LN-SUB)
             A-RED-COIN (LN-SUB)
             A-TOTAL-LN-DEDUCT (LN-SUB)
             A-BLOOD-LN-DEDUCT (LN-SUB).

```

```

*   IF OPPS-APC (LN-SUB) = '1111' OR '1112' OR '1113'
*     OR '1114' OR '1117' OR '6300' OR '6600'
*   IF L-SERVICE-FROM-DATE > 20010331
*     MOVE 30 TO A-RETURN-CODE (LN-SUB)
*     GO TO 0150-INIT-EXIT.

```

```

IF NOT (OPPS-SRVC-IND (LN-SUB) = 'A' OR 'B' OR 'C' OR
'E' OR 'F' OR 'G' OR 'H' OR 'J' OR 'K' OR 'L' OR 'N'
OR 'P' OR 'S' OR 'T' OR 'V' OR 'X' OR 'W' OR 'Y' OR
'Z') THEN
  MOVE 40 TO A-RETURN-CODE (LN-SUB)
  GO TO 0150-INIT-EXIT
ELSE
  MOVE 01 TO A-RETURN-CODE (LN-SUB).

```

```

IF OPPS-SRVC-IND (LN-SUB) =

```

```

          'A' OR 'B' OR 'C' OR 'E' OR 'F'
MOVE 41 TO A-RETURN-CODE (LN-SUB)
GO TO 0150-INIT-EXIT.

IF OPPTS-PYMT-IND (LN-SUB) = '1' OR '5' OR '6' OR
                          '7' OR '8' OR '9'
  IF OPPTS-PKG-FLAG (LN-SUB) = '0' OR '1' OR '2'
    IF (OPPTS-LITEM-DR-FLAG (LN-SUB) = '0' OR
        OPPTS-LITEM-DR-FLAG (LN-SUB) = '1' AND
        (OPPTS-APC (LN-SUB) = '0033' OR '0034'
        OR '0322' OR '0323' OR '0324' OR '0325'
        OR '0373' OR '0374')) OR
        (OPPTS-LITEM-ACT-FLAG (LN-SUB) = '1')
      IF NOT (OPPTS-LITEM-ACT-FLAG (LN-SUB)
              EQUAL '2' OR '3')
        IF OPPTS-PYMT-ADJ-FLAG (LN-SUB) = '0' OR '1'
          OR '2' OR '3' OR '4'
          IF (OPPTS-SITE-SRVC-FLAG (LN-SUB) = '0') OR
            ((APC33-FLAG = 'Y') AND
             ((OPPTS-SRVC-IND (LN-SUB) = 'P') OR
              (OPPTS-APC (LN-SUB) = '0322' OR '0323'
              OR '0324' OR '0325' OR '0373'
              OR '0374'))))
        MOVE OPPTS-SRVC-UNITS (LN-SUB) TO H-SRVC-UNITS

MOVE OPPTS-SUB-CHRG (LN-SUB) TO H-SUB-CHRG
COMPUTE H-TOT-CHRG = H-TOT-CHRG +
                  H-SUB-CHRG
  IF (OPPTS-APC (LN-SUB) = '0000') OR
    (OPPTS-PKG-FLAG (LN-SUB) NOT = 0)
    MOVE 42 TO A-RETURN-CODE (LN-SUB)
    GO TO 0150-INIT-EXIT
  END-IF
SEARCH ALL WAA-ENTRY
  AT END
    MOVE 30 TO A-RETURN-CODE (LN-SUB)
    GO TO 0150-INIT-EXIT
  WHEN WAA-APC (WAA-INDX) = OPPTS-GRP (LN-SUB)
    MOVE WAA-PTR (WAA-INDX) TO W-SUB2
    PERFORM 0175-APC-LOOKUP
  ELSE
    MOVE 49 TO A-RETURN-CODE (LN-SUB)
    GO TO 0150-INIT-EXIT
  ELSE
    MOVE 48 TO A-RETURN-CODE (LN-SUB)
    GO TO 0150-INIT-EXIT
  ELSE
    MOVE 47 TO A-RETURN-CODE (LN-SUB)
    GO TO 0150-INIT-EXIT
  ELSE
    MOVE 46 TO A-RETURN-CODE (LN-SUB)
    GO TO 0150-INIT-EXIT
  ELSE
    MOVE 45 TO A-RETURN-CODE (LN-SUB)
    GO TO 0150-INIT-EXIT
  ELSE
    MOVE 43 TO A-RETURN-CODE (LN-SUB)

```



GO TO 0150-INIT-EXIT.

```
IF A-RETURN-CODE (LN-SUB) = 01
  PERFORM 0250-CALC-DISCOUNT
  THRU 0250-CALC-DISCOUNT-EXIT
ELSE
  GO TO 0150-INIT-EXIT.
IF A-RETURN-CODE (LN-SUB) = 01
  PERFORM 0300-COIN-DEDUCT
  THRU 0300-COIN-DEDUCT-EXIT.

IF A-RETURN-CODE (LN-SUB) = 01
  SET WBD-INDX TO 1
  SEARCH WBD-ENTRY VARYING WBD-INDX
  AT END
  GO TO 0150-INIT-EXIT
  WHEN W-BLOOD-HCPCS (WBD-INDX) = OPPTS-HCPCS (LN-SUB)
  MOVE W-BLOOD-RANK (WBD-INDX) TO H-BLOOD-RANK
  PERFORM 0375-BLOOD-DEDUCT
  THRU 0375-BLOOD-DEDUCT-EXIT.
```

0150-INIT-EXIT.  
EXIT.

```
*****
* ADJUST APC FROM 3M USING THE HCPCS CODE AND APC *
* - RESET SERVICE INDICATOR WHEN NECESSARY *
* - DISABLED AS OF 04-01-2001 *
*****
* 0160-ADJUST-APC.
*
* MOVE OPPTS-HCPCS (LN-SUB) TO W-ADJ-HCPCS.
* MOVE OPPTS-GRP (LN-SUB) TO W-ADJ-APC.
* SET WAAJ-INDX TO 1.
* SEARCH WAAJ-ENTRY VARYING WAAJ-INDX
* AT END
* GO TO 0160-ADJUST-APC-EXIT
* WHEN WAAJ-HCPCSAPC (WAAJ-INDX) = W-ADJ-HCPCSAPC
* MOVE WAAJ-FED-APC (WAAJ-INDX) TO
* OPPTS-GRP (LN-SUB)
* MOVE WAAJ-NEW-SRVC-IND (WAAJ-INDX) TO
* OPPTS-SRVC-IND (LN-SUB)
* MOVE WAAJ-NEW-PYMT-IND (WAAJ-INDX) TO
* OPPTS-PYMT-IND (LN-SUB)
* MOVE WAAJ-NEW-PYMT-ADJ (WAAJ-INDX) TO
* OPPTS-PYMT-ADJ-FLAG (LN-SUB).
* 0160-ADJUST-APC-EXIT.
* EXIT.
*****
```

0175-APC-LOOKUP.

```
IF WAR-DTCD (W-SUB2) NOT > WAD-DTCD (WAD-SUB)
  IF WAR-RATEX (W-SUB2) = 'DELETED'
  MOVE 30 TO A-RETURN-CODE (LN-SUB)
  COMPUTE H-TOT-CHRG = H-TOT-CHRG -
```

```

                                H-SUB-CHRG
ELSE
    MOVE WAR-RATE (W-SUB2) TO H-APC-PYMT
    MOVE WAR-RANK (W-SUB2) TO H-RANK
    MOVE WAR-MINC (W-SUB2) TO H-MIN-COIN
    MOVE WAR-COIN (W-SUB2) TO H-NAT-COIN
    MOVE WAR-PPCT (W-SUB2) TO H-PPCT
ELSE
    SUBTRACT 1 FROM W-SUB2
    IF W-SUB2 > WAA-PTR (WAA-INDX - 1)
        GO TO 0175-APC-LOOKUP
    ELSE
        MOVE 0 TO H-APC-PYMT
            H-RANK
            H-MIN-COIN
            H-NAT-COIN
            H-PPCT.

0175-APC-LOOKUP-EXIT.
EXIT.

*****
*   SEARCH WAGE INDEX TABLE AND SELECT THE WAGE INDEX           *
*   WHEN EQUAL TO PROVIDER SPECIFIC WAGE INDEX                   *
*   IF WAGE INDEX NOT LOCATED                                     *
*       - SET CLAIM RETURN CODE TO '50'                          *
*       - DISCONTINUE CLAIM PROCESSING                           *
*   IF WAGE INDEX EQUALS ZERO                                     *
*       - SET CLAIM RETURN CODE TO '51'                          *
*       - DISCONTINUE CLAIM PROCESSING                           *
*****

0200-CALC-WAGEINDX.

MOVE WWD-MAX TO WWD-SUB.
PERFORM UNTIL L-SERVICE-FROM-DATE NOT < WWD-DATE (WWD-SUB)
    SUBTRACT 1 FROM WWD-SUB
END-PERFORM.

SEARCH ALL WWM-ENTRY
    AT END
        MOVE 50 TO A-CLM-RTN-CODE
        GO TO 0200-CALC-WAGEINDX-EXIT
    WHEN WWM-MSA (WWM-INDX) = H-PSF-MSA
        MOVE WWM-PTR (WWM-INDX) TO W-SUB2
        PERFORM 0210-WAGE-LOOKUP.

IF H-WINX1 = 0 THEN
    MOVE 51 TO A-CLM-RTN-CODE.

0200-CALC-WAGEINDX-EXIT.
EXIT.

0210-WAGE-LOOKUP.

IF WWW-DTCD (W-SUB2) NOT > WWD-DTCD (WWD-SUB)
    IF L-PSF-WGIDX-RECLASS = 'Y'
        MOVE WWW-WINX2 (W-SUB2) TO H-WINX1

```

```

ELSE
  MOVE WWW-WINX1 (W-SUB2) TO H-WINX1
END-IF
ELSE
  SUBTRACT 1 FROM W-SUB2
  IF W-SUB2 > WWM-PTR (WWM-INDX - 1)
    GO TO 0210-WAGE-LOOKUP
  ELSE
    MOVE 0 TO H-WINX1.

0210-WAGE-LOOKUP-EXIT.
EXIT.

0220-CHNG-WAGEINDX.

IF (L-PSF-PROV-OSCAR = '140012' OR '150002' OR '150004'
    OR '150008' OR '150034' OR '150090'
    OR '150125' OR '150128' OR '150132')
  AND (L-PSF-GEO-MSA = '1600'
  AND L-PSF-WI-MSA = '1600'
  AND L-PSF-WGIDX-RECLASS = 'Y')
  MOVE 01.0750 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '250078')
  AND (L-PSF-GEO-MSA = '3285'
  AND L-PSF-WI-MSA = '3285'
  AND L-PSF-WGIDX-RECLASS = 'Y')
  MOVE 00.7634 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '330001' OR '330126' OR '330135'
    OR '330205' OR '330209' OR '330264')
  AND (L-PSF-GEO-MSA = '5600'
  AND L-PSF-WI-MSA = '5600'
  AND L-PSF-WGIDX-RECLASS = 'Y')
  MOVE 01.4342 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '340039' OR '340129' OR '340144')
  AND (L-PSF-GEO-MSA = '1520'
  AND L-PSF-WI-MSA = '1520'
  AND L-PSF-WGIDX-RECLASS = 'Y')
  MOVE 00.9434 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '360046' OR '360056' OR '360076'
    OR '360132')
  AND (L-PSF-GEO-MSA = '1640'
  AND L-PSF-WI-MSA = '1640'
  AND L-PSF-WGIDX-RECLASS = 'Y')
  MOVE 00.9419 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '390019' OR '390049' OR '390162'
    OR '390194' OR '390197' OR '390263')
  AND (L-PSF-GEO-MSA = '0240'
  AND L-PSF-WI-MSA = '0240'
  AND L-PSF-WGIDX-RECLASS = 'Y')
  MOVE 01.0228 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '450065' OR '450072' OR '450591')

```

AND (L-PSF-GEO-MSA = '3360'  
AND L-PSF-WI-MSA = '3360'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 00.9388 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '470003')  
AND (L-PSF-GEO-MSA = '1123'  
AND L-PSF-WI-MSA = '1123'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 01.1359 TO H-WINX1.

0220-CHNG-WAGEINDX-EXIT.  
EXIT.

\*\*\*\*\*  
\* FOR FY 2001 NEW LUGAR HOSPITALS ONLY \*  
\*\*\*\*\*

0225-CHNG-WAGEINDX.

IF (L-PSF-PROV-OSCAR = '010043')  
AND (L-PSF-WI-MSA = '1000'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 00.8490 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '010072' OR '010101')  
AND (L-PSF-WI-MSA = '0450'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 00.7871 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '100098')  
AND (L-PSF-WI-MSA = '8960'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 00.9615 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '100232')  
AND (L-PSF-WI-MSA = '2900'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 01.0074 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '110130')  
AND (L-PSF-WI-MSA = '0500'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 00.9739 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '140230')  
AND (L-PSF-WI-MSA = '1400'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 00.9069 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '230027')  
AND (L-PSF-WI-MSA = '3000'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 01.0119 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '340071' OR '340124')  
AND (L-PSF-WI-MSA = '6640')

AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 00.9506 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '390030' OR '390181' OR '390183')  
AND (L-PSF-WI-MSA = '6680'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 00.8992 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '390201')  
AND (L-PSF-WI-MSA = '5640'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 01.0890 TO H-WINX1.

0225-CHNG-WAGEINDX-EXIT.  
EXIT.

\*\*\*\*\*  
\* CALCULATE DISCOUNT FACTOR BASED ON THE DISCOUNT \*  
\* INDICATOR PASSED BY THE OCE: VALUE 1 - 8 \*  
\* IF MISSING OR INVALID DISCOUNT FACTOR \*  
\* - SET RETURN CODE TO '38' \*  
\* - DISCONTINUE LINE PROCESSING \*  
\*\*\*\*\*

0250-CALC-DISCOUNT.

IF OPPTS-DISC-FACT (LN-SUB) = 1 THEN  
MOVE 1 TO H-DISC-RATE  
ELSE  
IF OPPTS-DISC-FACT (LN-SUB) = 2 THEN  
COMPUTE H-DISC-RATE = (1 + DISC-FRACTION \*  
(H-SRVC-UNITS - 1)) / H-SRVC-UNITS  
ELSE  
IF OPPTS-DISC-FACT (LN-SUB) = 3 THEN  
COMPUTE H-DISC-RATE = TERM-PROC-DISC  
/ H-SRVC-UNITS  
ELSE  
IF OPPTS-DISC-FACT (LN-SUB) = 4 THEN  
COMPUTE H-DISC-RATE = (1 + DISC-FRACTION)  
/ H-SRVC-UNITS  
ELSE  
IF OPPTS-DISC-FACT (LN-SUB) = 5 THEN  
COMPUTE H-DISC-RATE = DISC-FRACTION  
ELSE  
IF OPPTS-DISC-FACT (LN-SUB) = 6 THEN  
COMPUTE H-DISC-RATE = (TERM-PROC-DISC \*  
DISC-FRACTION) / H-SRVC-UNITS  
ELSE  
IF OPPTS-DISC-FACT (LN-SUB) = 7 THEN  
COMPUTE H-DISC-RATE = (DISC-FRACTION \*  
(1 + DISC-FRACTION) / H-SRVC-UNITS)  
ELSE  
IF OPPTS-DISC-FACT (LN-SUB) = 8 THEN  
MOVE 2 TO H-DISC-RATE  
ELSE  
MOVE 38 TO A-RETURN-CODE (LN-SUB).

0250-CALC-DISCOUNT-EXIT.

EXIT.

```
*****
*   DETERMINE THE DEDUCTIBLE SEQUENCE. DEDUCTIBLE WILL BE   *
*   TAKEN FROM OPPTS SERVICES FIRST, THEN FROM ANY OTHER   *
*   TYPES OF SERVICES FROM THE CLAIM.                       *
*   - SET POINTERS TO THE HIGHEST RANKED PERCENTAGE       *
*     (LARGEST NATIONAL UNADJUSTED COINSURANCE /          *
*     THE APC PAYMENT)                                     *
*   - MOVE ALL PRICING VARIABLES TO STAGING AREA           *
*****
```

0300-COIN-DEDUCT.

```
ADD 1 TO W-LNC-MAX.
SET W-LP-INDX TO W-LNC-MAX.
PERFORM 0350-STAGE-ENTRY
  THRU 0350-STAGE-ENTRY-EXIT
  UNTIL W-LP-INDX = 1 OR
        H-RANK NOT < W-RANK (W-LP-INDX - 1).
MOVE LN-SUB TO W-LP-SUB (W-LP-INDX).
MOVE H-NAT-COIN TO W-NAT-COIN (W-LP-INDX).
MOVE H-MIN-COIN TO W-MIN-COIN (W-LP-INDX).
MOVE H-APC-PYMT TO W-APC-PYMT (W-LP-INDX).
MOVE H-WINX1 TO W-WINX1 (W-LP-INDX).
MOVE H-RANK TO W-RANK (W-LP-INDX).
MOVE H-PPCT TO W-PPCT (W-LP-INDX).
MOVE H-DISC-RATE TO W-DISC-RATE (W-LP-INDX).

IF OPPTS-SRVC-IND (LN-SUB) = 'P' THEN
  MOVE 1 TO W-SRVC-UNITS (W-LP-INDX)
ELSE
  MOVE H-SRVC-UNITS TO W-SRVC-UNITS (W-LP-INDX).

MOVE 0 TO W-RED-COIN (W-LP-INDX).
PERFORM VARYING PS-SUB FROM 1 BY 1
  UNTIL PS-SUB > L-PSF-APC-LINE-CNT
  IF L-PSF-APC (PS-SUB) = OPPTS-APC (LN-SUB)
    COMPUTE W-RED-COIN (W-LP-INDX) ROUNDED =
      L-PSF-RED-COIN (PS-SUB) * H-SRVC-UNITS * H-DISC-RATE
    MOVE 25 TO A-RETURN-CODE (LN-SUB)
    MOVE L-PSF-APC-LINE-CNT TO PS-SUB
  END-IF
END-PERFORM.
```

0300-COIN-DEDUCT-EXIT.

EXIT.

0350-STAGE-ENTRY.

```
MOVE W-LP-ENTRY (W-LP-INDX - 1) TO
  W-LP-ENTRY (W-LP-INDX).
SET W-LP-INDX DOWN BY 1.
```

0350-STAGE-ENTRY-EXIT.

EXIT.

```
*****
```

```

*   DETERMINE THE BLOOD DEDUCTIBLE SEQUENCE. DEDUCTIBLE   *
*   WILL BE CALCULATED BASED ON RANKING.                   *
*   - SET POINTERS TO THE HIGHEST RANKED APC               *
*   (SMALLEST UNADJUSTED APC PAYMENT)                     *
*   - MOVE ALL PRICING VARIABLES TO STAGING AREA           *
*****

```

0375-BLOOD-DEDUCT.

```

ADD 1 TO W-BLD-MAX.
SET W-BD-INDX TO W-BLD-MAX.
PERFORM 0385-STAGE-ENTRY
  THRU 0385-STAGE-ENTRY-EXIT
  UNTIL W-BD-INDX = 1 OR
    H-BLOOD-RANK NOT < W-BD-RANK (W-BD-INDX - 1).
MOVE LN-SUB TO W-BD-SUB (W-BD-INDX).
MOVE H-NAT-COIN TO W-BD-NAT-COIN (W-BD-INDX).
MOVE H-MIN-COIN TO W-BD-MIN-COIN (W-BD-INDX).
MOVE H-SUB-CHRG TO W-BD-SUB-CHRG (W-BD-INDX).
MOVE H-APC-PYMT TO W-BD-APC-PYMT (W-BD-INDX).
MOVE H-WINX1 TO W-BD-WINX1 (W-BD-INDX).
MOVE H-BLOOD-RANK TO W-BD-RANK (W-BD-INDX).
MOVE H-PPCT TO W-BD-PPCT (W-BD-INDX).
MOVE H-DISC-RATE TO W-BD-DISC-RATE (W-BD-INDX).
MOVE H-SRVC-UNITS TO W-BD-SRVC-UNITS (W-BD-INDX).

```

```

MOVE 0 TO W-BD-RED-COIN (W-BD-INDX).
PERFORM VARYING PS-SUB FROM 1 BY 1
  UNTIL PS-SUB > L-PSF-APC-LINE-CNT
  IF L-PSF-APC (PS-SUB) = OPPS-APC (LN-SUB)
    COMPUTE W-BD-RED-COIN (W-BD-INDX) ROUNDED =
      L-PSF-RED-COIN (PS-SUB) * H-SRVC-UNITS * H-DISC-RATE
    MOVE 25 TO A-RETURN-CODE (LN-SUB)
    MOVE L-PSF-APC-LINE-CNT TO PS-SUB
  END-IF
END-PERFORM.

```

0375-BLOOD-DEDUCT-EXIT.  
EXIT.

0385-STAGE-ENTRY.

```

MOVE W-BD-ENTRY (W-BD-INDX - 1) TO
  W-BD-ENTRY (W-BD-INDX).
SET W-BD-INDX DOWN BY 1.

```

0385-STAGE-ENTRY-EXIT.  
EXIT.

```

*****
*   FIRST STEP IN DETERMINING A LINE ITEM PRICE.           *
*   CHECK ALL APPROPRIATE FLAGS AND INDICATORS PASSED     *
*   BY THE OCE.                                             *
*   - SET RETURN CODES TO INDICATE ERRORS OR STATUS        *
*   - '20' - LINE PROCESSED BUT PAYMENT = 0                *
*   - BENE DEDUCTIBLE => ADJUSTED PAYMENT                  *
*

```

\*\*\*\*\*  
0400-CALCULATE.

MOVE W-LP-SUB (W-LP-INDX) TO LN-SUB.  
IF A-RETURN-CODE (LN-SUB) > 25  
GO TO 0400-CALCULATE-EXIT.

IF OPPTS-PYMT-IND (LN-SUB) = '1' OR '5'  
OR '6' OR '7' OR '8'  
PERFORM 0550-CALC-STANDARD  
THRU 0550-CALC-STANDARD-EXIT  
ELSE  
GO TO 0400-CALCULATE-EXIT.

\*\*\*\*\*  
\* SET STVX AND / OR GJK FLAGS \*  
\* - TEST LINE ITEM DATE OF SERVICE > 20010630 \*  
\* - TOTAL DRUG / DEVICE COINSURANCE \*  
\*\*\*\*\*

IF (A-RETURN-CODE (LN-SUB) < 30) AND  
(OPPTS-LITEM-DOS (LN-SUB) > 20001231)  
PERFORM 0450-ADJ-PROC-COIN  
THRU 0450-ADJ-PROC-COIN-EXIT  
ELSE  
NEXT SENTENCE.

IF A-RETURN-CODE (LN-SUB) < 30  
COMPUTE A-TOTAL-CLM-DEDUCT =  
H-TOTAL-LN-DEDUCT + A-TOTAL-CLM-DEDUCT  
COMPUTE H-TOT-PYMT = H-TOT-PYMT + H-LITEM-PYMT  
COMPUTE A-BLOOD-DEDUCT-DUE =  
A-BLOOD-DEDUCT-DUE + H-LN-BLOOD-DEDUCT  
MOVE H-LITEM-PYMT TO A-LITEM-PYMT (LN-SUB)  
MOVE H-LITEM-REIM TO A-LITEM-REIM (LN-SUB)  
MOVE H-TOTAL-LN-DEDUCT TO A-TOTAL-LN-DEDUCT (LN-SUB)  
MOVE H-LN-BLOOD-DEDUCT TO A-BLOOD-LN-DEDUCT (LN-SUB)  
MOVE H-NAT-COIN TO A-ADJ-COIN (LN-SUB)  
MOVE H-RED-COIN TO A-RED-COIN (LN-SUB)  
IF H-RED-COIN > H-NAT-COIN  
MOVE H-NAT-COIN TO A-RED-COIN (LN-SUB)  
END-IF  
IF OPPTS-LITEM-ACT-FLAG (LN-SUB) = '4'  
MOVE 0 TO A-LITEM-PYMT (LN-SUB)  
MOVE 0 TO A-LITEM-REIM (LN-SUB)  
COMPUTE H-TOT-PYMT = H-TOT-PYMT - H-LITEM-PYMT  
END-IF  
END-IF.  
MOVE ZERO TO LINE-HOLD-ITEMS.

0400-CALCULATE-EXIT.  
EXIT.

\*\*\*\*\*  
\* SET STVX AND / OR GJK FLAGS \*  
\* - STAGE BY SERVICE INDICATOR \*  
\* \*  
\*\*\*\*\*



\*\*\*\*\*

0450-ADJ-PROC-COIN.

```
MOVE OPPS-LITEM-DOS (LN-SUB) TO H-DCP-DOS.
IF OPPS-SRVC-IND (LN-SUB) = 'S' OR 'T' OR 'V'
  COMPUTE H-NEW-WGNAT ROUNDED =
    W-NAT-COIN (W-LP-INDX) *
      (.6 * W-WINX1 (W-LP-INDX) + .4)
  IF H-NEW-WGNAT > H-IP-LIMIT
    MOVE H-IP-LIMIT TO H-NEW-WGNAT
  END-IF
  COMPUTE H-NEW-COIN = H-LITEM-PYMT -
    H-TOTAL-LN-DEDUCT - H-LITEM-REIM - H-LN-BLOOD-DEDUCT
  MOVE 1 TO H-DCP-CODE
  PERFORM 0455-SEARCH-KEY
    THRU 0455-SEARCH-KEY-EXIT
ELSE
  IF OPPS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR 'K'
    COMPUTE H-NEW-COIN = H-LITEM-PYMT -
      H-TOTAL-LN-DEDUCT - H-LITEM-REIM
      - H-LN-BLOOD-DEDUCT
    MOVE 'Y' TO GJK-FLAG
    MOVE 1 TO H-DCP-CODE
    PERFORM 0455-SEARCH-KEY
      THRU 0455-SEARCH-KEY-EXIT
    MOVE 2 TO H-DCP-CODE
    ADD 1 TO W-DCP-MAX
    SET W-DCP-INDX TO W-DCP-MAX
    PERFORM 0475-STAGE-DCP-ENTRY
      THRU 0475-STAGE-DCP-ENTRY-EXIT
      UNTIL W-DCP-INDX = 1 OR
        H-DCP-STAGE NOT < W-DCP-STAGE (W-DCP-INDX - 1)
    MOVE LN-SUB TO W-DCP-SUB (W-DCP-INDX)
    MOVE H-DCP-STAGE TO W-DCP-STAGE (W-DCP-INDX)
    MOVE ZERO TO W-DCP-COIN1 (W-DCP-INDX)
    MOVE ZERO TO W-DCP-WGNAT (W-DCP-INDX)
    MOVE H-NEW-COIN TO W-DCP-COIN2 (W-DCP-INDX)
    MOVE OPPS-SRVC-IND (LN-SUB) TO
      W-DCP-SRVC-IND (W-DCP-INDX).
```

0450-ADJ-PROC-COIN-EXIT.

EXIT.

0455-SEARCH-KEY.

```
SET W-DCP-INDX TO 1.
SEARCH W-DCP-ENTRY VARYING W-DCP-INDX
  AT END
    PERFORM 0460-ADD-ENTRY
      THRU 0460-ADD-ENTRY-EXIT
    WHEN W-DCP-STAGE (W-DCP-INDX) = H-DCP-STAGE
      PERFORM 0465-UPDATE-ENTRY
        THRU 0465-UPDATE-ENTRY-EXIT.
```

0455-SEARCH-KEY-EXIT.

EXIT.

0460-ADD-ENTRY.

```
ADD 1 TO W-DCP-MAX.
SET W-DCP-INDX TO W-DCP-MAX.
PERFORM 0475-STAGE-DCP-ENTRY
  THRU 0475-STAGE-DCP-ENTRY-EXIT
  UNTIL W-DCP-INDX = 1 OR
    H-DCP-STAGE NOT < W-DCP-STAGE (W-DCP-INDX - 1).
IF OPPTS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR 'K'
  MOVE ZERO TO W-DCP-SUB (W-DCP-INDX)
  MOVE H-DCP-STAGE TO W-DCP-STAGE (W-DCP-INDX)
  MOVE ZERO TO W-DCP-COIN1 (W-DCP-INDX)
  MOVE ZERO TO W-DCP-WGNAT (W-DCP-INDX)
  MOVE H-NEW-COIN TO W-DCP-COIN2 (W-DCP-INDX)
  MOVE 'X' TO W-DCP-SRVC-IND (W-DCP-INDX)
ELSE
  MOVE LN-SUB TO W-DCP-SUB (W-DCP-INDX)
  MOVE H-DCP-STAGE TO W-DCP-STAGE (W-DCP-INDX)
  MOVE H-NEW-COIN TO W-DCP-COIN1 (W-DCP-INDX)
  MOVE H-NEW-WGNAT TO W-DCP-WGNAT (W-DCP-INDX)
  MOVE ZERO TO W-DCP-COIN2 (W-DCP-INDX)
  MOVE OPPTS-SRVC-IND (LN-SUB) TO
    W-DCP-SRVC-IND (W-DCP-INDX).
```

0460-ADD-ENTRY-EXIT.

EXIT.

0465-UPDATE-ENTRY.

```
IF OPPTS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR 'K'
  ADD H-NEW-COIN TO W-DCP-COIN2 (W-DCP-INDX)
ELSE
  IF W-DCP-SRVC-IND (W-DCP-INDX) = 'X'
    PERFORM 0485-REPLACE-TYPE1
    THRU 0485-REPLACE-TYPE1-EXIT
  ELSE
    PERFORM 0480-RANK-COIN
    THRU 0480-RANK-COIN-EXIT.
```

0465-UPDATE-ENTRY-EXIT.

EXIT.

0475-STAGE-DCP-ENTRY.

```
MOVE W-DCP-ENTRY (W-DCP-INDX - 1) TO
  W-DCP-ENTRY (W-DCP-INDX).
SET W-DCP-INDX DOWN BY 1.
```

0475-STAGE-DCP-ENTRY-EXIT.

EXIT.

0480-RANK-COIN.

```
IF H-NEW-WGNAT > W-DCP-WGNAT (W-DCP-INDX)
  MOVE LN-SUB TO W-DCP-SUB (W-DCP-INDX)
  MOVE H-NEW-COIN TO W-DCP-COIN1 (W-DCP-INDX)
  MOVE H-NEW-WGNAT TO W-DCP-WGNAT (W-DCP-INDX)
```

MOVE OPSS-SRVC-IND (LN-SUB)  
TO W-DCP-SRVC-IND (W-DCP-INDX).

0480-RANK-COIN-EXIT.  
EXIT.

0485-REPLACE-TYPE1.

MOVE LN-SUB TO W-DCP-SUB (W-DCP-INDX)  
MOVE H-NEW-COIN TO W-DCP-COIN1 (W-DCP-INDX)  
MOVE H-NEW-WGNAT TO W-DCP-WGNAT (W-DCP-INDX)  
MOVE OPSS-SRVC-IND (LN-SUB)  
TO W-DCP-SRVC-IND (W-DCP-INDX).

0485-REPLACE-TYPE1-EXIT.  
EXIT.

```
*****
* 1. APPLY DEDUCTIBLE TO HIGHEST NATIONAL COINSURANCE AMOUNT *
*   - DESCENDING UNTIL DEDUCTIBLE = 0. *
* 2. CALCULATE THE STANDARD LINE PRICE *
*   (APC PAYMENT * WAGE INDEX * SERVICE UNITS *
*     * DISCOUNT FACTOR) *
*   - WAGE ADJUST 60% OF THE APC PAYMENT ONLY *
* 3. ADD LINE PRICE TO OUTLIER HOLD AREA. *
* *
* 4. CALCULATE THE LINE PRICE FOR DESIGNATED DEVICES *
*   AND DRUGS *
*****
```

0550-CALC-STANDARD.

COMPUTE H-MAX-COIN ROUNDED = (H-IP-LIMIT \*  
W-SRVC-UNITS (W-LP-INDX) \* W-DISC-RATE (W-LP-INDX)).  
MOVE 0 TO H-BLOOD-FRACTION.  
IF (OPSS-SRVC-IND (LN-SUB) = 'S' OR 'X') AND  
(OPSS-HCPCS (LN-SUB) = 'P9010' OR 'P9016' OR 'P9021'  
OR 'P9022' OR 'P9038' OR 'P9039' OR 'P9040'  
OR 'C1010' OR 'C1018')  
PERFORM 0550-SET-BLOOD-FRACTION  
THRU 0550-SET-BLOOD-FRACTION-EXIT  
PERFORM 0550-CALC-FY00-BLOOD-DED  
THRU 0550-CALC-FY00-BLOOD-DED-EXIT  
GO TO 0550-CALC-STANDARD-EXIT.

IF OPSS-SRVC-IND (LN-SUB) = 'S' OR 'V'  
OR 'T' OR 'P' OR 'X' THEN  
COMPUTE H-LITEM-PYMT ROUNDED =  
(((W-APC-PYMT (W-LP-INDX) \* .60) \*  
W-WINX1 (W-LP-INDX))  
+ (W-APC-PYMT (W-LP-INDX) \* .40)) \*  
W-SRVC-UNITS (W-LP-INDX) \* W-DISC-RATE (W-LP-INDX)  
PERFORM 0560-CALC-BENE-DEDUCT  
THRU 0560-CALC-BENE-DEDUCT-EXIT  
ELSE  
IF OPSS-SRVC-IND (LN-SUB) = 'H' THEN  
IF OPSS-PYMT-IND (LN-SUB) = '6' THEN  
PERFORM 0555-CALC-H-STANDARD

```

        THRU 0555-CALC-H-STANDARD-EXIT
        PERFORM 0560-CALC-BENE-DEDUCT
        THRU 0560-CALC-BENE-DEDUCT-EXIT
    ELSE
        MOVE 44 TO A-RETURN-CODE (LN-SUB)
        GO TO 0550-CALC-STANDARD-EXIT
    END-IF
ELSE
    IF OPPTS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR 'K' THEN
    IF OPPTS-PYMT-IND (LN-SUB) = '1' OR '5' OR '7' THEN
        PERFORM 0550-CALC-GJK
            THRU 0550-CALC-GJK-EXIT
        PERFORM 0560-CALC-BENE-DEDUCT
            THRU 0560-CALC-BENE-DEDUCT-EXIT
    ELSE
        MOVE 41 TO A-RETURN-CODE (LN-SUB)
        GO TO 0550-CALC-STANDARD-EXIT
    END-IF
END-IF.

IF H-LITEM-PYMT > 0
    COMPUTE H-LITEM-REIM ROUNDED =
        ((H-LITEM-PYMT - H-TOTAL-LN-DEDUCT) -
         H-LN-BLOOD-DEDUCT) * W-PPCT (W-LP-INDX)
    COMPUTE H-NAT-COIN = H-LITEM-PYMT -
        H-TOTAL-LN-DEDUCT - H-LITEM-REIM - H-LN-BLOOD-DEDUCT.
    MOVE W-MIN-COIN (W-LP-INDX) TO H-MIN-COIN.
    IF H-MIN-COIN > 0
        IF OPPTS-SRVC-IND (LN-SUB) = 'G' OR 'H' OR 'J' OR 'K'
            COMPUTE H-MIN-COIN ROUNDED = H-MIN-COIN *
                (W-SRVC-UNITS (W-LP-INDX) -
                 (W-SRVC-UNITS (W-LP-INDX) * H-BLOOD-FRACTION))
                * W-DISC-RATE (W-LP-INDX)
        ELSE
            IF OPPTS-APC (LN-SUB) = '0158' OR '0159'
                COMPUTE H-MIN-COIN ROUNDED = H-LITEM-PYMT * .25
            ELSE
                COMPUTE H-MIN-COIN ROUNDED = H-LITEM-PYMT * .2
            END-IF
        END-IF
    END-IF.

    MOVE W-RED-COIN (W-LP-INDX) TO H-RED-COIN.
    IF (H-RED-COIN > 0 AND < H-MIN-COIN)
        MOVE H-MIN-COIN TO H-RED-COIN
    ELSE
        IF H-RED-COIN < H-NAT-COIN AND > H-MIN-COIN
            AND > H-MAX-COIN
                COMPUTE H-LITEM-REIM = H-LITEM-REIM +
                    (H-RED-COIN - H-MAX-COIN)
            END-IF
        END-IF.

    IF H-NAT-COIN > H-MAX-COIN AND H-RED-COIN = 0 THEN
        COMPUTE H-LITEM-REIM = H-LITEM-REIM +
            (H-NAT-COIN - H-MAX-COIN)
        MOVE H-MAX-COIN TO H-NAT-COIN.

```

0550-CALC-STANDARD-EXIT.  
EXIT.

0550-CALC-FY00-BLOOD-DED.

COMPUTE H-LITEM-PYMT ROUNDED =  
(((W-BD-APC-PYMT (W-BD-INDX) \* .60) \*  
W-BD-WINX1 (W-BD-INDX))  
+ (W-BD-APC-PYMT (W-BD-INDX) \* .40)) \*  
W-BD-SRVC-UNITS (W-BD-INDX) \*  
W-BD-DISC-RATE (W-BD-INDX)  
COMPUTE H-LN-BLOOD-DEDUCT ROUNDED =  
H-LITEM-PYMT \* H-BLOOD-FRACTION  
SET W-BD-INDX UP BY 1  
PERFORM 0560-CALC-BENE-DEDUCT  
THRU 0560-CALC-BENE-DEDUCT-EXIT.

IF H-LITEM-PYMT > 0  
COMPUTE H-LITEM-REIM ROUNDED =  
((H-LITEM-PYMT - H-TOTAL-LN-DEDUCT) -  
H-LN-BLOOD-DEDUCT) \* W-PPCT (W-LP-INDX)  
COMPUTE H-NAT-COIN = H-LITEM-PYMT -  
H-TOTAL-LN-DEDUCT - H-LITEM-REIM - H-LN-BLOOD-DEDUCT.  
MOVE W-MIN-COIN (W-LP-INDX) TO H-MIN-COIN.  
IF H-MIN-COIN > 0  
COMPUTE H-MIN-COIN ROUNDED = H-MIN-COIN \*  
(W-SRVC-UNITS (W-LP-INDX) -  
(W-SRVC-UNITS (W-LP-INDX) \* H-BLOOD-FRACTION))  
\* W-DISC-RATE (W-LP-INDX).

MOVE W-RED-COIN (W-LP-INDX) TO H-RED-COIN.  
IF (H-RED-COIN > 0 AND < H-MIN-COIN)  
MOVE H-MIN-COIN TO H-RED-COIN  
ELSE  
IF H-RED-COIN < H-NAT-COIN AND > H-MIN-COIN  
AND > H-MAX-COIN  
COMPUTE H-LITEM-REIM = H-LITEM-REIM +  
(H-RED-COIN - H-MAX-COIN)  
END-IF  
END-IF.

IF H-NAT-COIN > H-MAX-COIN AND H-RED-COIN = 0 THEN  
COMPUTE H-LITEM-REIM = H-LITEM-REIM +  
(H-NAT-COIN - H-MAX-COIN)  
MOVE H-MAX-COIN TO H-NAT-COIN.

IF H-LITEM-PYMT = H-LN-BLOOD-DEDUCT  
MOVE 0 TO H-LITEM-REIM  
MOVE 0 TO H-NAT-COIN.

0550-CALC-FY00-BLOOD-DED-EXIT.  
EXIT.

\*\*\*\*\*  
\* 1. CALCULATE LINE ITEM PAYMENT FOR SERVICE INDICATOR \*  
\* TYPES G , J , OR K. \*  
\* 2. EFFECTIVE 01/01/2003 REMOVE PRO RATA REDUCTION FOR \*

\* ALL SERVICE INDICATOR 'G' PAYMENTS \*

0550-CALC-GJK.

IF OPPTS-HCPCS(LN-SUB) = 'P9021' OR 'P9010' OR 'P9038'  
OR 'P9016' OR 'C1010' OR 'C1018' OR 'P9022'  
OR 'P9039' OR 'P9040' OR 'C1016'

PERFORM 0550-SET-BLOOD-FRACTION  
THRU 0550-SET-BLOOD-FRACTION-EXIT

ELSE

COMPUTE H-LITEM-PYMT ROUNDED =  
W-APC-PYMT (W-LP-INDX) \* W-SRVC-UNITS (W-LP-INDX)  
\* W-DISC-RATE (W-LP-INDX)  
GO TO 0550-CALC-GJK-EXIT.

COMPUTE H-LITEM-PYMT ROUNDED =  
W-BD-APC-PYMT (W-BD-INDX) \* W-BD-SRVC-UNITS (W-BD-INDX)  
\* W-BD-DISC-RATE (W-BD-INDX).

COMPUTE H-LN-BLOOD-DEDUCT ROUNDED =  
H-LITEM-PYMT \* H-BLOOD-FRACTION.

SET W-BD-INDX UP BY 1.

0550-CALC-GJK-EXIT.

EXIT.

0550-SET-BLOOD-FRACTION.

MOVE W-BD-SUB (W-BD-INDX) TO LN-SUB.

IF H-BENE-PINTS-USED > 0

IF W-BD-SRVC-UNITS (W-BD-INDX) <= H-BENE-PINTS-USED

MOVE 1 TO H-BLOOD-FRACTION

COMPUTE H-BENE-PINTS-USED =

H-BENE-PINTS-USED - W-BD-SRVC-UNITS (W-BD-INDX)

ELSE

IF W-BD-SRVC-UNITS (W-BD-INDX) > H-BENE-PINTS-USED

COMPUTE H-BLOOD-FRACTION =

H-BENE-PINTS-USED / W-BD-SRVC-UNITS (W-BD-INDX)

MOVE 0 TO H-BENE-PINTS-USED

ELSE

MOVE 0 TO H-BLOOD-FRACTION

ELSE

MOVE 0 TO H-BLOOD-FRACTION.

0550-SET-BLOOD-FRACTION-EXIT.

EXIT.

0555-CALC-H-STANDARD.

MOVE OPPTS-SUB-CHRG (LN-SUB) TO H-SUB-CHRG

IF L-SERVICE-FROM-DATE > 20001231

COMPUTE T-LITEM-PYMT ROUNDED =

(H-SUB-CHRG \* (L-PSF-OPCOST-RATIO \* .981956)) -

((W-APC-PYMT (W-LP-INDX) \* .60) \*

```

        W-WINX1 (W-LP-INDX))
        + (W-APC-PYMT (W-LP-INDX) * .40))
ELSE
    COMPUTE T-LITEM-PYMT ROUNDED =
        (H-SUB-CHRG * L-PSF-OPCOST-RATIO) -
        (((W-APC-PYMT (W-LP-INDX) * .60) *
            W-WINX1 (W-LP-INDX))
            + (W-APC-PYMT (W-LP-INDX) * .40)).

    IF T-LITEM-PYMT < 0 THEN
        MOVE 0 TO H-LITEM-PYMT
    ELSE
        MOVE T-LITEM-PYMT TO H-LITEM-PYMT.

0555-CALC-H-STANDARD-EXIT.
EXIT.

0560-CALC-BENE-DEDUCT.

IF OPSS-PYMT-ADJ-FLAG (LN-SUB) = '4'
GO TO 0560-CALC-BENE-DEDUCT-EXIT.
IF H-BENE-DEDUCT > 0 THEN
    COMPUTE H-LN-BLD-PYMT =
        H-LITEM-PYMT - H-LN-BLOOD-DEDUCT
    IF H-BENE-DEDUCT <= H-LN-BLD-PYMT THEN
        MOVE H-BENE-DEDUCT TO H-TOTAL-LN-DEDUCT
        MOVE 0 TO H-BENE-DEDUCT
    ELSE
        COMPUTE H-BENE-DEDUCT =
            H-BENE-DEDUCT - H-LN-BLD-PYMT
        MOVE H-LN-BLD-PYMT TO H-TOTAL-LN-DEDUCT
        MOVE 20 TO A-RETURN-CODE (LN-SUB)
    END-IF
END-IF.

0560-CALC-BENE-DEDUCT-EXIT.
EXIT.

0800-ADJ-STV-REIM.

IF W-DCP-CODE (W-DCP-INDX) = 1
    PERFORM 0810-PROCESS-TYPE1
        THRU 0810-PROCESS-TYPE1-EXIT
ELSE
    PERFORM 0840-PROCESS-TYPE2
        THRU 0840-PROCESS-TYPE2-EXIT.

0800-ADJ-STV-REIM-EXIT.
EXIT.

0810-PROCESS-TYPE1.

IF W-DCP-COIN2 (W-DCP-INDX) > 0
    MOVE W-DCP-DOS (W-DCP-INDX) TO H-DCP-DOS
    MOVE W-DCP-WGNAT (W-DCP-INDX) TO H-TOTAL
    COMPUTE H-RATIO =
        (H-IP-LIMIT - W-DCP-WGNAT (W-DCP-INDX)) /

```

```

                                W-DCP-COIN2 (W-DCP-INDX)
IF H-RATIO < 0
    MOVE 0 TO H-RATIO.
IF H-RATIO > 1
    MOVE 1 TO H-RATIO.

0810-PROCESS-TYPE1-EXIT.
    EXIT.

0840-PROCESS-TYPE2.

IF W-DCP-DOS (W-DCP-INDX) >= 20010701
    IF W-DCP-DOS (W-DCP-INDX) = H-DCP-DOS
        MOVE W-DCP-SUB (W-DCP-INDX) TO LN-SUB
        COMPUTE H-SHIFT =
            W-DCP-COIN2 (W-DCP-INDX) * (1 - H-RATIO)
        COMPUTE H-TOTAL =
            A-ADJ-COIN (LN-SUB) + H-TOTAL - H-SHIFT
        IF H-TOTAL > H-IP-LIMIT
            COMPUTE H-SHIFT = H-SHIFT + H-TOTAL
                - H-IP-LIMIT
        END-IF
        COMPUTE A-ADJ-COIN (LN-SUB) =
            A-ADJ-COIN (LN-SUB) - H-SHIFT
        COMPUTE A-LITEM-REIM (LN-SUB) =
            A-LITEM-REIM (LN-SUB) + H-SHIFT
        MOVE 22 TO A-RETURN-CODE (LN-SUB)
    END-IF
ELSE
    IF W-DCP-DOS (W-DCP-INDX) >= 20010101
        IF W-DCP-COIN2 (W-DCP-INDX) > H-IP-LIMIT
            MOVE W-DCP-SUB (W-DCP-INDX) TO LN-SUB
            COMPUTE H-SHIFT =
                W-DCP-COIN2 (W-DCP-INDX) - H-IP-LIMIT
            COMPUTE A-ADJ-COIN (LN-SUB) =
                A-ADJ-COIN (LN-SUB) - H-SHIFT
            COMPUTE A-LITEM-REIM (LN-SUB) =
                A-LITEM-REIM (LN-SUB) + H-SHIFT
            MOVE 22 TO A-RETURN-CODE (LN-SUB)
        END-IF
    END-IF
END-IF.

0840-PROCESS-TYPE2-EXIT.
    EXIT.

```

```

*****
* 1. MOVE TOTAL CLAIM CHARGE AMOUNT. *
* 2. MOVE TOTAL CLAIM PAYMENT AMOUNT. *
* 3. MOVE TOTAL CLAIM BLOOD PINTS USED. *
* 4. CALCULATE CLAIM LEVEL OUTLIER AMOUNT. *
*****

```

```

0900-END-PRICE-RTN.

```

```

    MOVE H-TOT-CHRG TO A-TOT-CLM-CHRG.

```



```

MOVE H-TOT-PYMT TO A-TOT-CLM-PYMT.
COMPUTE A-BLOOD-PINTS-USED =
      H-BENE-BLOOD-PINTS - H-BENE-PINTS-USED.
IF A-TOT-CLM-PYMT > 0
  PERFORM 0910-CALC-OUTLIER
  THRU 0910-CALC-OUTLIER-EXIT.
IF H-OUTLIER-PYMT > 0
  MOVE H-OUTLIER-PYMT TO A-OUTLIER-PYMT.

```

```

0900-END-PRICE-RTN-EXIT.
EXIT.

```

```

0910-CALC-OUTLIER.

```

```

IF L-SERVICE-FROM-DATE > 20001231
  COMPUTE H-OUTLIER-PYMT ROUNDED =
    ((H-TOT-CHRG * (L-PSF-OPCOST-RATIO * .981956))
     - (2.5 * H-TOT-PYMT)) * .75
ELSE
  COMPUTE H-OUTLIER-PYMT ROUNDED =
    ((H-TOT-CHRG * L-PSF-OPCOST-RATIO)
     - (2.5 * H-TOT-PYMT)) * .75.

```

```

0910-CALC-OUTLIER-EXIT.
EXIT.

```

```

*****
*   PROCESSING: AFTER 20020331   *
*   A. WILL PROCESS LINE ITEMS BASED ON OCE FLAGS.   *
*   B. INITIALIZE OPPTS HOLD VARIABLES.   *
*   C. EDIT THE DATA PASSED FROM THE OCE.   *
*   D. ASSEMBLE PRICING COMPONENTS.   *
*   E. CALCULATE THE PRICE.   *
*   F. RETURN COINSURANCE/REIMBURSEMENT/DEDUCTIBLE/   *
*       PAYMENT/OUTLIER AMOUNT/RETURN CODES   *
*****

```

```

1000-PROCESS-MAIN-NEW.

```

```

PERFORM 1100-INIT
  THRU 1100-INIT-EXIT.

```

```

IF A-CLM-RTN-CODE >= 50
  GOBACK.

```

```

MOVE H-WINX1 TO A-WINX.
PERFORM 1125-INIT
  THRU 1125-INIT-EXIT
  VARYING LN-SUB FROM 1 BY 1
  UNTIL LN-SUB > OPPTS-LINE-CNT.

```

```

MOVE 0 TO W-DCP-MAX W-BLD-MAX.
PERFORM 1150-INIT
  THRU 1150-INIT-EXIT
  VARYING LN-SUB FROM 1 BY 1
  UNTIL LN-SUB > OPPTS-LINE-CNT.

```

```

MOVE 0 TO W-DCP-MAX.

```

```
IF L-SERVICE-FROM-DATE > 20020331
  PERFORM 1555-CALC-H-TOT
  THRU 1555-CALC-H-TOT-EXIT
  VARYING W-LP-INDX FROM 1 BY 1
  UNTIL W-LP-INDX > W-LNC-MAX.
```

```
SET W-BD-INDX TO 1.
MOVE 0 TO W-DCP-MAX.
PERFORM 1400-CALCULATE
  THRU 1400-CALCULATE-EXIT
  VARYING W-LP-INDX FROM 1 BY 1
  UNTIL W-LP-INDX > W-LNC-MAX.
```

```
IF L-SERVICE-FROM-DATE > 20020331
  PERFORM 1600-ADJ-CHRG-OUTL
  THRU 1600-ADJ-CHRG-OUTL-EXIT
  VARYING W-LP-INDX FROM 1 BY 1
  UNTIL W-LP-INDX > W-LNC-MAX
ELSE
  NEXT SENTENCE.
```

```
IF GJK-FLAG = 'Y'
  PERFORM 1800-ADJ-STV-REIM
  THRU 1800-ADJ-STV-REIM-EXIT
  VARYING W-DCP-INDX FROM 1 BY 1
  UNTIL W-DCP-INDX > W-DCP-MAX
ELSE
  NEXT SENTENCE.
```

```
PERFORM 1900-END-PRICE-RTN
  THRU 1900-END-PRICE-RTN-EXIT.
```

```
1000-PROCESS-MAIN-NEW-EXIT.
EXIT.
```

```
*****
*   INITIALIZE WORKING STORAGE HOLD AREAS                               *
*   AND ADDITIONAL VARIABLES TO BE PASSED BACK TO                       *
*   THE STANDARD SYSTEM.                                               *
*   - IF PROVIDER SPECIFIC FILE WAGE INDEX RECLASSIFICATION           *
*   CODE INVALID OR MISSING                                           *
*     - MOVE '52' TO CLAIM LEVEL RETURN CODE                           *
*     - DISCONTINUE CLAIM PROCESSING                                    *
*   - IF SERVICE FROM DATE NOT NUMERIC OR < 20000801                   *
*     - MOVE '53' TO CLAIM LEVEL RETURN CODE                           *
*     - DISCONTINUE CLAIM PROCESSING                                    *
*   - IF SERVICE FROM DATE < PROVIDER EFFECTIVE DATE                   *
*     - MOVE '54' TO CLAIM LEVEL RETURN CODE                           *
*     - DISCONTINUE CLAIM PROCESSING                                    *
*   - IF SERVICE FROM DATE > PROVIDER TERMINATION DATE                 *
*     - MOVE '54' TO CLAIM LEVEL RETURN CODE                           *
*     - DISCONTINUE CLAIM PROCESSING                                    *
*****
```

```
1100-INIT.
```

```
MOVE 01 TO A-CLM-RTN-CODE.
```

```

MOVE 'N' TO APC33-FLAG GJK-FLAG ST0-FLAG
      N-FLAG C-FLAG.
IF L-SERVICE-FROM-DATE NOT NUMERIC
  MOVE 53 TO A-CLM-RTN-CODE
  GO TO 1100-INIT-EXIT
ELSE
  IF L-SERVICE-FROM-DATE < 20000801
    MOVE 53 TO A-CLM-RTN-CODE
    GO TO 1100-INIT-EXIT
  ELSE
    IF L-SERVICE-FROM-DATE < L-PSF-EFFDT
      MOVE 54 TO A-CLM-RTN-CODE
      GO TO 1100-INIT-EXIT
    ELSE
      IF L-PSF-TERMDT > 0
        IF L-SERVICE-FROM-DATE > L-PSF-TERMDT
          MOVE 54 TO A-CLM-RTN-CODE
          GO TO 1100-INIT-EXIT
        END-IF
      END-IF.
MOVE SPACE TO A-MSA.
MOVE ZERO TO A-OUTLIER-PYMT
      A-TOTAL-CLM-DEDUCT
      A-TOT-CLM-CHRG
      A-TOT-CLM-PYMT
      A-BLOOD-PINTS-USED
      A-BLOOD-DEDUCT-DUE
      A-WINX
      W-LNC-MAX.
INITIALIZE H-ADDITIONAL-VARIABLES.
INITIALIZE LINE-HOLD-ITEMS.
INITIALIZE T-LITEM-PYMT.
MOVE CAL-VERSION1 TO A-CALC-VERS.
MOVE BENE-DEDUCT TO H-BENE-DEDUCT.
MOVE BENE-BLOOD-PINTS TO H-BENE-BLOOD-PINTS
      H-BENE-PINTS-USED.
MOVE WAD-MAX TO WAD-SUB.
PERFORM UNTIL L-SERVICE-FROM-DATE
      NOT < WAD-DATE (WAD-SUB)
  SUBTRACT 1 FROM WAD-SUB
END-PERFORM.
IF L-PSF-WGIDX-RECLASS = 'Y'
  MOVE L-PSF-WI-MSA TO H-PSF-MSA
ELSE
  IF L-PSF-WGIDX-RECLASS = 'N'
    MOVE L-PSF-GEO-MSA TO H-PSF-MSA
  ELSE
    MOVE 52 TO A-CLM-RTN-CODE
    GO TO 1100-INIT-EXIT.

IF L-SERVICE-FROM-DATE >= 20020101
  MOVE 812 TO H-IP-LIMIT
ELSE
  IF L-SERVICE-FROM-DATE >= 20010101
    MOVE 792 TO H-IP-LIMIT
  ELSE
    IF L-SERVICE-FROM-DATE >= 20000801

```

MOVE 776 TO H-IP-LIMIT.

```
IF L-SERVICE-FROM-DATE > 20020331
  PERFORM 1115-FLOOR-2002
  THRU 1115-FLOOR-2002-EXIT
  PERFORM 1115-SEC401-2002
  THRU 1115-SEC401-2002-EXIT
ELSE
  IF L-SERVICE-FROM-DATE > 20001231
    PERFORM 1110-FLOOR-2001
    THRU 1110-FLOOR-2001-EXIT
  ELSE
    IF L-SERVICE-FROM-DATE > 20000731
      PERFORM 1105-FLOOR-2000
      THRU 1105-FLOOR-2000-EXIT.
```

MOVE H-PSF-MSA TO A-MSA.

```
IF L-SERVICE-FROM-DATE >= 20010101
  PERFORM 1225-CHNG-WAGEINDX
  THRU 1225-CHNG-WAGEINDX-EXIT
ELSE
  PERFORM 1220-CHNG-WAGEINDX
  THRU 1220-CHNG-WAGEINDX-EXIT.

IF H-WINX1 = 0 THEN
  PERFORM 1200-CALC-WAGEINDX
  THRU 1200-CALC-WAGEINDX-EXIT.
```

1100-INIT-EXIT.  
EXIT.

```
*****
*   RESET FLOOR MSA - 'FROM-DATE' CONTROLLED   *
*   - YEAR 2000                                 *
*   - YEAR 2001                                 *
*   - YEAR 2002                                 *
*****
```

1105-FLOOR-2000.

```
IF (H-PSF-MSA = '6020' OR '9000' OR '8080') AND
  (L-PSF-PROV-ST = '36')
  MOVE ' 36' TO H-PSF-MSA
ELSE
  IF (H-PSF-MSA = '2440') AND
    (L-PSF-PROV-ST = '15')
    MOVE ' 15' TO H-PSF-MSA
  ELSE
    IF (H-PSF-MSA = '2520') AND
      (L-PSF-PROV-ST = '24') AND
      (L-PSF-WGIDX-RECLASS = 'Y')
      MOVE ' 24' TO H-PSF-MSA
    ELSE
      IF (H-PSF-MSA = '1123') AND
        (L-PSF-PROV-ST = '22')
```

MOVE ' 22' TO H-PSF-MSA.

1105-FLOOR-2000-EXIT.

EXIT.

1110-FLOOR-2001.

IF (H-PSF-MSA = '6020' OR '9000' OR '8080') AND  
(L-PSF-PROV-ST = '36')

MOVE ' 36' TO H-PSF-MSA

ELSE

IF (H-PSF-MSA = '2440') AND  
(L-PSF-PROV-ST = '15')

MOVE ' 15' TO H-PSF-MSA

ELSE

IF (H-PSF-MSA = '9000') AND  
(L-PSF-PROV-ST = '51')

MOVE ' 51' TO H-PSF-MSA

ELSE

IF (H-PSF-MSA = '1900') AND  
(L-PSF-PROV-ST = '21')

MOVE ' 21' TO H-PSF-MSA

ELSE

IF (H-PSF-MSA = '1123') AND  
(L-PSF-PROV-ST = '22')

MOVE ' 22' TO H-PSF-MSA.

1110-FLOOR-2001-EXIT.

EXIT.

1115-FLOOR-2002.

IF (H-PSF-MSA = '6020' OR '9000' OR '8080') AND  
(L-PSF-PROV-ST = '36')

MOVE ' 36' TO H-PSF-MSA

ELSE

IF (H-PSF-MSA = '2440') AND  
(L-PSF-PROV-ST = '15')

MOVE ' 15' TO H-PSF-MSA

ELSE

IF (H-PSF-MSA = '1303') AND  
(L-PSF-PROV-ST = '47') AND  
(L-PSF-WGIDX-RECLASS = 'Y')

MOVE ' 47' TO H-PSF-MSA

MOVE 'N' TO L-PSF-WGIDX-RECLASS

ELSE

IF (H-PSF-MSA = '1900') AND  
(L-PSF-PROV-ST = '21')

MOVE ' 21' TO H-PSF-MSA

ELSE

IF (H-PSF-MSA = '1123') AND  
(L-PSF-PROV-ST = '22')

MOVE ' 22' TO H-PSF-MSA

ELSE

IF (H-PSF-MSA = ' 14') AND

(L-PSF-PROV-ST = '16') AND

```
(L-PSF-WGIDX-RECLASS = 'Y')
  MOVE 'N' TO L-PSF-WGIDX-RECLASS
  MOVE ' 16' TO H-PSF-MSA.
```

```
1115-FLOOR-2002-EXIT.
  EXIT.
```

```
1115-SEC401-2002.
```

```
  IF (L-PSF-PROV-OSCAR = '050192' OR '050286' OR
      '050446' OR '050469' OR
      '050528' OR '050542')
    MOVE ' 05' TO H-PSF-MSA
  ELSE
    IF (L-PSF-PROV-OSCAR = '100048' OR '100118')
      MOVE ' 10' TO H-PSF-MSA
    ELSE
      IF (L-PSF-PROV-OSCAR = '170137')
        MOVE ' 17' TO H-PSF-MSA
      ELSE
        IF (L-PSF-PROV-OSCAR = '190048' OR '190110')
          MOVE ' 19' TO H-PSF-MSA
        ELSE
          IF (L-PSF-PROV-OSCAR = '230078')
            MOVE ' 23' TO H-PSF-MSA
          ELSE
            IF (L-PSF-PROV-OSCAR = '260006')
              MOVE ' 26' TO H-PSF-MSA
            ELSE
              IF (L-PSF-PROV-OSCAR = '290038')
                MOVE ' 29' TO H-PSF-MSA
              ELSE
                IF (L-PSF-PROV-OSCAR = '300009')
                  MOVE ' 30' TO H-PSF-MSA
                ELSE
                  IF (L-PSF-PROV-OSCAR = '390106')
                    MOVE ' 39' TO H-PSF-MSA
                  ELSE
                    IF (L-PSF-PROV-OSCAR = '520007' OR
                        '520153')
                      MOVE ' 52' TO H-PSF-MSA.
```

```
1115-SEC401-2002-EXIT.
  EXIT.
```

```
*****
* SET FLAG IF APC = 0033 *
* - TERMINATE PROCESS IF 0033 LOCATED *
* *
*****
```

```
1125-INIT.
```

```
  IF OPPTS-APC (LN-SUB) = '0033'
    MOVE 'Y' TO APC33-FLAG
    MOVE 451 TO LN-SUB.
```

1125-INIT-EXIT.  
EXIT.

```
*****
* 1. VERIFY THE SERVICE INDICATOR PASSED BY THE OCE *
* - IF INVALID SET RETURN CODE TO '40' *
* - DISCONTINUE LINE PROCESSING *
* - IF VALID SET RETURN CODE TO '01' *
* - CONTINUE LINE PROCESSING *
* 2. PROCESS LINES AND LOAD WORK TABLE ACCORDING TO PRICE *
* RANKING *
* 3. CHECK OCE EDIT INDICATORS AND SET RETURN CODES IF *
* INDICATORS ARE INVALID. *
* - VALID RETURN CODES FOR EDIT INDICATORS *
* - '41' - SERVICE INDICATOR INVALID FOR OPPTS PRICER *
* - '42' - APC = '00000' OR (PACKAGING FLAG = 1 OR 2) *
* - '43' - PAYMENT INDICATOR NOT = TO 1 OR 5 THRU 9 *
* - '44' - SERVICE INDICATOR = 'H' BUT PAYMENT *
* INDICATOR NOT = TO 6 *
* - '45' - PACKAGING FLAG NOT = TO 0 *
* - '46' - (LINE ITEM DENIAL/REJECT FLAG NOT = TO 0 *
* OR LINE ITEM DENIAL/REJECT FLAG = TO 1 *
* AND (APC NOT = 0033 OR 0034 OR 0322 OR *
* 0323 OR 0324 OR 0325 OR 0373 OR 0374)) *
* OR LINE ITEM ACTION FLAG NOT = TO 1 *
* - '47' - LINE ITEM ACTION FLAG = 2 OR 3 *
* - '48' - PAYMENT ADJUSTMENT FLAG NOT VALID *
* - '49' - SITE OF SERVICE FLAG NOT = TO 0 *
* - OR (APC 0033 IS NOT ON THE CLAIM *
* AND SERVICE INDICATOR = 'P' *
* OR APC = 0322-0325,0373,0374) *
* 4. IF OCE INDICATORS ARE VALID, SEARCH APC TABLE *
* - IF MISSING, DELETED OR INVALID APC *
* - SET RETURN CODE TO '30' *
* - DISCONTINUE LINE PROCESSING *
```

\*\*\*\*\*  
1150-INIT.

MOVE 01 TO A-RETURN-CODE (LN-SUB).

```
*****
* OVER-RIDE SERVICE UNITS FOR APC 0339 - EFFECTIVE 04/01/2002 *
* - CHANGE UNIT VALUE TO 1 *
```

```
IF (L-SERVICE-FROM-DATE > 20020331) AND
(OPPS-APC (LN-SUB) = '0339')
MOVE 1 TO OPPTS-SRVC-UNITS (LN-SUB).
```

```
MOVE OPPTS-SRVC-UNITS (LN-SUB) TO H-SRVC-UNITS.
PERFORM 1250-CALC-DISCOUNT
THRU 1250-CALC-DISCOUNT-EXIT.
```

```
IF A-RETURN-CODE (LN-SUB) = 42
GO TO 1150-INIT-EXIT.
```

\*\*\*\*\*

\* EFFECTIVE AS OF 04-01-2002 \*  
\* - TOTAL DEVICE OFFSET \*  
\*\*\*\*\*

IF L-SERVICE-FROM-DATE > 20020331 AND  
OPPS-SRVC-IND (LN-SUB) = 'H'  
MOVE 'Y' TO C-FLAG.  
IF L-SERVICE-FROM-DATE > 20020331  
PERFORM 1160-TOTAL-OFFSET  
THRU 1160-TOTAL-OFFSET-EXIT.

SET W-LP-INDX TO LN-SUB.  
SET W-BD-INDX TO LN-SUB.  
INITIALIZE W-LP-ENTRY (W-LP-INDX).  
INITIALIZE W-BD-ENTRY (W-BD-INDX).  
MOVE ZERO TO A-LITEM-PYMT (LN-SUB)  
A-LITEM-REIM (LN-SUB)  
A-ADJ-COIN (LN-SUB)  
A-RED-COIN (LN-SUB)  
A-TOTAL-LN-DEDUCT (LN-SUB)  
A-BLOOD-LN-DEDUCT (LN-SUB).

IF NOT (OPPS-SRVC-IND (LN-SUB) = 'A' OR 'B' OR 'C' OR  
'E' OR 'F' OR 'G' OR 'H' OR 'J' OR 'K' OR 'L' OR 'N'  
OR 'P' OR 'S' OR 'T' OR 'V' OR 'X' OR 'W' OR 'Y' OR  
'Z') THEN  
MOVE 40 TO A-RETURN-CODE (LN-SUB)  
GO TO 1150-INIT-EXIT  
ELSE  
MOVE 01 TO A-RETURN-CODE (LN-SUB).

IF OPPS-SRVC-IND (LN-SUB) =  
'A' OR 'B' OR 'C' OR 'E' OR 'F'  
MOVE 41 TO A-RETURN-CODE (LN-SUB)  
GO TO 1150-INIT-EXIT.

IF OPPS-PYMT-IND (LN-SUB) = '1' OR '5' OR '6' OR  
'7' OR '8' OR '9'  
IF OPPS-PKG-FLAG (LN-SUB) = '0' OR '1' OR '2'  
IF (OPPS-LITEM-DR-FLAG (LN-SUB) = '0' OR  
OPPS-LITEM-DR-FLAG (LN-SUB) = '1' AND  
(OPPS-APC (LN-SUB) = '0033' OR '0034'  
OR '0322' OR '0323' OR '0324' OR '0325'  
OR '0373' OR '0374')) OR  
(OPPS-LITEM-ACT-FLAG (LN-SUB) = '1')  
IF NOT (OPPS-LITEM-ACT-FLAG (LN-SUB)  
EQUAL '2' OR '3')  
IF OPPS-PYMT-ADJ-FLAG (LN-SUB) = '0' OR '1'  
OR '2' OR '3' OR '4'  
IF (OPPS-SITE-SRVC-FLAG (LN-SUB) = '0') OR  
(APC33-FLAG = 'Y') AND  
((OPPS-SRVC-IND (LN-SUB) = 'P') OR  
(OPPS-APC (LN-SUB) = '0322' OR '0323'  
OR '0324' OR '0325' OR '0373'  
OR '0374'))))  
MOVE OPPS-SUB-CHRG (LN-SUB) TO H-SUB-CHRG  
COMPUTE H-TOT-CHRG = H-TOT-CHRG +



H-SUB-CHRG

```
IF OPPTS-PKG-FLAG (LN-SUB) = '1' OR '2'
  COMPUTE H-TOT-N-CHRG = H-SUB-CHRG
    + H-TOT-N-CHRG
  MOVE 'Y' TO N-FLAG
END-IF
IF (OPPTS-APC (LN-SUB) = '0000') OR
  (OPPTS-PKG-FLAG (LN-SUB) NOT = 0)
  MOVE 42 TO A-RETURN-CODE (LN-SUB)
  GO TO 1150-INIT-EXIT
END-IF
SEARCH ALL WAA-ENTRY
AT END
  MOVE 30 TO A-RETURN-CODE (LN-SUB)
  GO TO 1150-INIT-EXIT
WHEN WAA-APC (WAA-INDX) = OPPTS-GRP (LN-SUB)
  MOVE WAA-PTR (WAA-INDX) TO W-SUB2
  PERFORM 1175-APC-LOOKUP
  ELSE
    MOVE 49 TO A-RETURN-CODE (LN-SUB)
    GO TO 1150-INIT-EXIT
  ELSE
    MOVE 48 TO A-RETURN-CODE (LN-SUB)
    GO TO 1150-INIT-EXIT
  ELSE
    MOVE 47 TO A-RETURN-CODE (LN-SUB)
    GO TO 1150-INIT-EXIT
  ELSE
    MOVE 46 TO A-RETURN-CODE (LN-SUB)
    GO TO 1150-INIT-EXIT
  ELSE
    MOVE 45 TO A-RETURN-CODE (LN-SUB)
    GO TO 1150-INIT-EXIT
  ELSE
    MOVE 43 TO A-RETURN-CODE (LN-SUB)
    GO TO 1150-INIT-EXIT.

IF A-RETURN-CODE (LN-SUB) = 01
  PERFORM 1300-COIN-DEDUCT
  THRU 1300-COIN-DEDUCT-EXIT.

IF A-RETURN-CODE (LN-SUB) = 01
  SET WBD-INDX TO 1
  SEARCH WBD-ENTRY VARYING WBD-INDX
  AT END
    GO TO 1150-INIT-EXIT
  WHEN W-BLOOD-HCPCS (WBD-INDX) = OPPTS-HCPCS (LN-SUB)
    MOVE W-BLOOD-RANK (WBD-INDX) TO H-BLOOD-RANK
    PERFORM 1375-BLOOD-DEDUCT
    THRU 1375-BLOOD-DEDUCT-EXIT.

1150-INIT-EXIT.
EXIT.
```

```
*****
*   COMPUTE TOTAL OFFSET FROM TABLE 5 FOR DEVICES   *
```

```

*           - EFFECTIVE OF 04-01-2002           *
*****
1160-TOTAL-OFFSET.

      MOVE OPPTS-GRP (LN-SUB) TO W-OFF-APC.
      SEARCH ALL WOO-ENTRY
      AT END
      GO TO 1160-TOTAL-OFFSET-EXIT
      WHEN WOO-APC (WOO-INDX) = W-OFF-APC
      COMPUTE H-TOTAL-OFFSET = H-TOTAL-OFFSET
        + (WOO-OFFSET (WOO-INDX) * H-DISC-RATE).

1160-TOTAL-OFFSET-EXIT.
EXIT.
*****

1175-APC-LOOKUP.

      IF WAR-DTCD (W-SUB2) NOT > WAD-DTCD (WAD-SUB)
      IF WAR-RATEX (W-SUB2) = 'DELETED'
      MOVE 30 TO A-RETURN-CODE (LN-SUB)
      COMPUTE H-TOT-CHRG = H-TOT-CHRG -
        H-SUB-CHRG
      ELSE
      MOVE WAR-RATE (W-SUB2) TO H-APC-PYMT
      MOVE WAR-RANK (W-SUB2) TO H-RANK
      MOVE WAR-MINC (W-SUB2) TO H-MIN-COIN
      MOVE WAR-COIN (W-SUB2) TO H-NAT-COIN
      MOVE WAR-PPCT (W-SUB2) TO H-PPCT
      ELSE
      SUBTRACT 1 FROM W-SUB2
      IF W-SUB2 > WAA-PTR (WAA-INDX - 1)
      GO TO 1175-APC-LOOKUP
      ELSE
      MOVE 0 TO H-APC-PYMT
      H-RANK
      H-MIN-COIN
      H-NAT-COIN
      H-PPCT.

1175-APC-LOOKUP-EXIT.
EXIT.
*****
*   SEARCH WAGE INDEX TABLE AND SELECT THE WAGE INDEX   *
*   WHEN EQUAL TO PROVIDER SPECIFIC WAGE INDEX           *
*   IF WAGE INDEX NOT LOCATED                             *
*   - SET CLAIM RETURN CODE TO '50'                       *
*   - DISCONTINUE CLAIM PROCESSING                        *
*   IF WAGE INDEX EQUALS ZERO                             *
*   - SET CLAIM RETURN CODE TO '51'                       *
*   - DISCONTINUE CLAIM PROCESSING                        *
*****
1200-CALC-WAGEINDX.

      MOVE WWD-MAX TO WWD-SUB.
      PERFORM UNTIL L-SERVICE-FROM-DATE NOT < WWD-DATE (WWD-SUB)

```

SUBTRACT 1 FROM WWD-SUB  
END-PERFORM.

SEARCH ALL WWM-ENTRY

AT END  
MOVE 50 TO A-CLM-RTN-CODE  
GO TO 1200-CALC-WAGEINDX-EXIT  
WHEN WWM-MSA (WWM-INDX) = H-PSF-MSA  
MOVE WWM-PTR (WWM-INDX) TO W-SUB2  
PERFORM 1210-WAGE-LOOKUP.

IF H-WINX1 = 0 THEN  
MOVE 51 TO A-CLM-RTN-CODE.

1200-CALC-WAGEINDX-EXIT.  
EXIT.

1210-WAGE-LOOKUP.

IF WWW-DTCD (W-SUB2) NOT > WWD-DTCD (WWD-SUB)  
IF L-PSF-WGIDX-RECLASS = 'Y'  
MOVE WWW-WINX2 (W-SUB2) TO H-WINX1  
ELSE  
MOVE WWW-WINX1 (W-SUB2) TO H-WINX1  
END-IF  
ELSE  
SUBTRACT 1 FROM W-SUB2  
IF W-SUB2 > WWM-PTR (WWM-INDX - 1)  
GO TO 1210-WAGE-LOOKUP  
ELSE  
MOVE 0 TO H-WINX1.

1210-WAGE-LOOKUP-EXIT.  
EXIT.

1220-CHNG-WAGEINDX.

IF (L-PSF-PROV-OSCAR = '140012' OR '150002' OR '150004'  
OR '150008' OR '150034' OR '150090'  
OR '150125' OR '150128' OR '150132')  
AND (L-PSF-GEO-MSA = '1600'  
AND L-PSF-WI-MSA = '1600'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 01.0750 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '250078')  
AND (L-PSF-GEO-MSA = '3285'  
AND L-PSF-WI-MSA = '3285'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 00.7634 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '330001' OR '330126' OR '330135'  
OR '330205' OR '330209' OR '330264')  
AND (L-PSF-GEO-MSA = '5600'  
AND L-PSF-WI-MSA = '5600'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 01.4342 TO H-WINX1.



```

IF (L-PSF-PROV-OSCAR = '100232')
    AND (L-PSF-WI-MSA = '2900'
    AND L-PSF-WGIDX-RECLASS = 'Y')
    MOVE 01.0074 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '110130')
    AND (L-PSF-WI-MSA = '0500'
    AND L-PSF-WGIDX-RECLASS = 'Y')
    MOVE 00.9739 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '140230')
    AND (L-PSF-WI-MSA = '1400'
    AND L-PSF-WGIDX-RECLASS = 'Y')
    MOVE 00.9069 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '230027')
    AND (L-PSF-WI-MSA = '3000'
    AND L-PSF-WGIDX-RECLASS = 'Y')
    MOVE 01.0119 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '340071' OR '340124')
    AND (L-PSF-WI-MSA = '6640'
    AND L-PSF-WGIDX-RECLASS = 'Y')
    MOVE 00.9506 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '390030' OR '390181' OR '390183')
    AND (L-PSF-WI-MSA = '6680'
    AND L-PSF-WGIDX-RECLASS = 'Y')
    MOVE 00.8992 TO H-WINX1.

IF (L-PSF-PROV-OSCAR = '390201')
    AND (L-PSF-WI-MSA = '5640'
    AND L-PSF-WGIDX-RECLASS = 'Y')
    MOVE 01.0890 TO H-WINX1.

```

```

1225-CHNG-WAGEINDX-EXIT.
EXIT.

```

```

*****
*   CALCULATE DISCOUNT FACTOR BASED ON THE DISCOUNT   *
*   INDICATOR PASSED BY THE OCE: VALUE 1 - 8             *
*   IF MISSING OR INVALID DISCOUNT FACTOR              *
*   - SET RETURN CODE TO '38'                            *
*   - DISCONTINUE LINE PROCESSING                        *
*****

```

```

1250-CALC-DISCOUNT.

```

```

IF (H-SRVC-UNITS = 0 AND
    OPPTS-APC (LN-SUB) = '0000')
    MOVE 42 TO A-RETURN-CODE (LN-SUB)
    GO TO 1250-CALC-DISCOUNT-EXIT
ELSE
    IF (H-SRVC-UNITS = 0 AND
        OPPTS-APC (LN-SUB) > '0000')
        MOVE 1 TO H-SRVC-UNITS.

```

```

IF OPPTS-DISC-FACT (LN-SUB) = 1 THEN

```

```

        MOVE 1 TO H-DISC-RATE
ELSE
    IF OPPTS-DISC-FACT (LN-SUB) = 2 THEN
        COMPUTE H-DISC-RATE = (1 + DISC-FRACTION *
            (H-SRVC-UNITS - 1)) / H-SRVC-UNITS
    ELSE
        IF OPPTS-DISC-FACT (LN-SUB) = 3 THEN
            COMPUTE H-DISC-RATE = TERM-PROC-DISC
                / H-SRVC-UNITS
        ELSE
            IF OPPTS-DISC-FACT (LN-SUB) = 4 THEN
                COMPUTE H-DISC-RATE = (1 + DISC-FRACTION)
                    / H-SRVC-UNITS
            ELSE
                IF OPPTS-DISC-FACT (LN-SUB) = 5 THEN
                    COMPUTE H-DISC-RATE = DISC-FRACTION
                ELSE
                    IF OPPTS-DISC-FACT (LN-SUB) = 6 THEN
                        COMPUTE H-DISC-RATE = (TERM-PROC-DISC *
                            DISC-FRACTION) / H-SRVC-UNITS
                    ELSE
                        IF OPPTS-DISC-FACT (LN-SUB) = 7 THEN
                            COMPUTE H-DISC-RATE = (DISC-FRACTION *
                                (1 + DISC-FRACTION)) / H-SRVC-UNITS
                        ELSE
                            IF OPPTS-DISC-FACT (LN-SUB) = 8 THEN
                                MOVE 2 TO H-DISC-RATE
                            ELSE
                                MOVE 38 TO A-RETURN-CODE (LN-SUB).

```

```

1250-CALC-DISCOUNT-EXIT.
EXIT.

```

```

*****
*   DETERMINE THE DEDUCTIBLE SEQUENCE. DEDUCTIBLE WILL BE   *
*   TAKEN FROM OPPTS SERVICES FIRST, THEN FROM ANY OTHER   *
*   TYPES OF SERVICES FROM THE CLAIM.                       *
*   - SET POINTERS TO THE HIGHEST RANKED PERCENTAGE        *
*   (LARGEST NATIONAL UNADJUSTED COINSURANCE /             *
*   THE APC PAYMENT)                                        *
*   - MOVE ALL PRICING VARIABLES TO STAGING AREA           *
*****

```

```

1300-COIN-DEDUCT.

```

```

ADD 1 TO W-LNC-MAX.
SET W-LP-INDX TO W-LNC-MAX.
PERFORM 1350-STAGE-ENTRY
    THRU 1350-STAGE-ENTRY-EXIT
        UNTIL W-LP-INDX = 1 OR
            H-RANK NOT < W-RANK (W-LP-INDX - 1).
MOVE LN-SUB TO W-LP-SUB (W-LP-INDX).
MOVE H-NAT-COIN TO W-NAT-COIN (W-LP-INDX).
MOVE H-MIN-COIN TO W-MIN-COIN (W-LP-INDX).
MOVE H-SUB-CHRG TO W-SUB-CHRG (W-LP-INDX).
MOVE H-APC-PYMT TO W-APC-PYMT (W-LP-INDX).
MOVE H-WINX1 TO W-WINX1 (W-LP-INDX).
MOVE H-RANK TO W-RANK (W-LP-INDX).

```

```

MOVE H-PPCT      TO W-PPCT (W-LP-INDX).
MOVE H-DISC-RATE TO W-DISC-RATE (W-LP-INDX).

IF OPPTS-SRVC-IND (LN-SUB) = 'P' THEN
    MOVE 1 TO W-SRVC-UNITS (W-LP-INDX)
ELSE
    MOVE H-SRVC-UNITS TO W-SRVC-UNITS (W-LP-INDX).

MOVE 0 TO W-RED-COIN (W-LP-INDX).
PERFORM VARYING PS-SUB FROM 1 BY 1
UNTIL PS-SUB > L-PSF-APC-LINE-CNT
IF L-PSF-APC (PS-SUB) = OPPTS-APC (LN-SUB)
    COMPUTE W-RED-COIN (W-LP-INDX) ROUNDED =
        L-PSF-RED-COIN (PS-SUB) * H-SRVC-UNITS * H-DISC-RATE
    MOVE 25 TO A-RETURN-CODE (LN-SUB)
    MOVE L-PSF-APC-LINE-CNT TO PS-SUB
END-IF
END-PERFORM.

```

```

1300-COIN-DEDUCT-EXIT.
EXIT.

```

```

1350-STAGE-ENTRY.

```

```

MOVE W-LP-ENTRY (W-LP-INDX - 1) TO
    W-LP-ENTRY (W-LP-INDX).
SET W-LP-INDX DOWN BY 1.

```

```

1350-STAGE-ENTRY-EXIT.
EXIT.

```

```

*****
*   DETERMINE THE BLOOD DEDUCTIBLE SEQUENCE. DEDUCTIBLE   *
*   WILL BE CALCULATED BASED ON RANKING.                   *
*   - SET POINTERS TO THE HIGHEST RANKED APC               *
*   (SMALLEST UNADJUSTED APC PAYMENT)                     *
*   - MOVE ALL PRICING VARIABLES TO STAGING AREA          *
*****

```

```

1375-BLOOD-DEDUCT.

```

```

ADD 1 TO W-BLD-MAX.
SET W-BD-INDX TO W-BLD-MAX.
PERFORM 1385-STAGE-ENTRY
    THRU 1385-STAGE-ENTRY-EXIT
    UNTIL W-BD-INDX = 1 OR
        H-BLOOD-RANK NOT < W-BD-RANK (W-BD-INDX - 1).
MOVE LN-SUB TO W-BD-SUB (W-BD-INDX).
MOVE H-NAT-COIN TO W-BD-NAT-COIN (W-BD-INDX).
MOVE H-MIN-COIN TO W-BD-MIN-COIN (W-BD-INDX).
MOVE H-SUB-CHRG TO W-BD-SUB-CHRG (W-BD-INDX).
MOVE H-APC-PYMT TO W-BD-APC-PYMT (W-BD-INDX).
MOVE H-WINX1    TO W-BD-WINX1 (W-BD-INDX).
MOVE H-BLOOD-RANK TO W-BD-RANK (W-BD-INDX).
MOVE H-PPCT      TO W-BD-PPCT (W-BD-INDX).
MOVE H-DISC-RATE TO W-BD-DISC-RATE (W-BD-INDX).
MOVE H-SRVC-UNITS TO W-BD-SRVC-UNITS (W-BD-INDX).

```

```
MOVE 0 TO W-BD-RED-COIN (W-BD-INDX).
PERFORM VARYING PS-SUB FROM 1 BY 1
  UNTIL PS-SUB > L-PSF-APC-LINE-CNT
  IF L-PSF-APC (PS-SUB) = OPPTS-APC (LN-SUB)
    COMPUTE W-BD-RED-COIN (W-BD-INDX) ROUNDED =
      L-PSF-RED-COIN (PS-SUB) * H-SRVC-UNITS * H-DISC-RATE
    MOVE 25 TO A-RETURN-CODE (LN-SUB)
    MOVE L-PSF-APC-LINE-CNT TO PS-SUB
  END-IF
END-PERFORM.
```

```
1375-BLOOD-DEDUCT-EXIT.
  EXIT.
```

```
1385-STAGE-ENTRY.
```

```
  MOVE W-BD-ENTRY (W-BD-INDX - 1) TO
    W-BD-ENTRY (W-BD-INDX).
  SET W-BD-INDX DOWN BY 1.
```

```
1385-STAGE-ENTRY-EXIT.
  EXIT.
```

```
*****
*   FIRST STEP IN DETERMINING A LINE ITEM PRICE.           *
*   CHECK ALL APPROPRIATE FLAGS AND INDICATORS PASSED     *
*   BY THE OCE.                                           *
*   - SET RETURN CODES TO INDICATE ERRORS OR STATUS       *
*   - '20' - LINE PROCESSED BUT PAYMENT = 0              *
*   - BENE DEDUCTIBLE => ADJUSTED PAYMENT                 *
*                                                         *
*****
```

```
1400-CALCULATE.
```

```
  MOVE W-LP-SUB (W-LP-INDX) TO LN-SUB.
  IF A-RETURN-CODE (LN-SUB) > 25
    GO TO 1400-CALCULATE-EXIT.

  IF OPPTS-PYMT-IND (LN-SUB) = '1' OR '5'
    OR '6' OR '7' OR '8'
    PERFORM 1550-CALC-STANDARD
      THRU 1550-CALC-STANDARD-EXIT
  ELSE
    GO TO 1400-CALCULATE-EXIT.
```

```
*****
*   SET GJK FLAG                                           *
*   - TEST LINE ITEM DATE OF SERVICE > 20010630          *
*   - TOTAL DRUG / DEVICE COINSURANCE                    *
*****
```

```
  IF (A-RETURN-CODE (LN-SUB) < 30) AND
    (OPPTS-LITEM-DOS (LN-SUB) > 20001231)
    PERFORM 1450-ADJ-PROC-COIN
      THRU 1450-ADJ-PROC-COIN-EXIT
  ELSE
    NEXT SENTENCE.
```



```

*****
*   SET ST0 AND STVX FLAGS                               *
*   - TEST LINE ITEM DATE OF SERVICE > 20020331         *
*   - TOTAL PROCEDURE CHARGES AND BUNDLED CHARGES       *
*****

```

```

IF (OPPS-LITEM-DOS (LN-SUB) > 20020331)
  PERFORM 1500-ADJ-CHRGs
  THRU 1500-ADJ-CHRGs-EXIT
ELSE
  NEXT SENTENCE.

```

```

IF A-RETURN-CODE (LN-SUB) < 30
  COMPUTE A-TOTAL-CLM-DEDUCT =
    H-TOTAL-LN-DEDUCT + A-TOTAL-CLM-DEDUCT
  COMPUTE H-TOT-PYMT = H-TOT-PYMT + H-LITEM-PYMT
  COMPUTE A-BLOOD-DEDUCT-DUE =
    A-BLOOD-DEDUCT-DUE + H-LN-BLOOD-DEDUCT
  MOVE H-LITEM-PYMT TO A-LITEM-PYMT (LN-SUB)
  MOVE H-LITEM-REIM TO A-LITEM-REIM (LN-SUB)
  MOVE H-TOTAL-LN-DEDUCT TO A-TOTAL-LN-DEDUCT (LN-SUB)
  MOVE H-LN-BLOOD-DEDUCT TO A-BLOOD-LN-DEDUCT (LN-SUB)
  MOVE H-NAT-COIN TO A-ADJ-COIN (LN-SUB)
  MOVE H-RED-COIN TO A-RED-COIN (LN-SUB)
  IF H-RED-COIN > H-NAT-COIN
    MOVE H-NAT-COIN TO A-RED-COIN (LN-SUB)
  END-IF
  IF OPPS-LITEM-ACT-FLAG (LN-SUB) = '4'
    MOVE 0 TO A-LITEM-PYMT (LN-SUB)
    MOVE 0 TO A-LITEM-REIM (LN-SUB)
    COMPUTE H-TOT-PYMT = H-TOT-PYMT - H-LITEM-PYMT
  END-IF
END-IF.
MOVE ZERO TO LINE-HOLD-ITEMS.

```

```

1400-CALCULATE-EXIT.
EXIT.

```

```

*****
*   SET GJK FLAG                                         *
*   - STAGE BY SERVICE INDICATOR                         *
*                                                         *
*****

```

```

1450-ADJ-PROC-COIN.

```

```

MOVE OPPS-LITEM-DOS (LN-SUB) TO H-DCP-DOS.
IF OPPS-SRVC-IND (LN-SUB) = 'S' OR 'T' OR 'V'
  COMPUTE H-NEW-WGNAT ROUNDED =
    W-NAT-COIN (W-LP-INDX) *
    (.6 * W-WINX1 (W-LP-INDX) + .4)
  IF H-NEW-WGNAT > H-IP-LIMIT
    MOVE H-IP-LIMIT TO H-NEW-WGNAT
  END-IF
  COMPUTE H-NEW-COIN = H-LITEM-PYMT -
    H-TOTAL-LN-DEDUCT - H-LITEM-REIM - H-LN-BLOOD-DEDUCT
  MOVE 1 TO H-DCP-CODE

```

```

        PERFORM 1455-SEARCH-KEY
            THRU 1455-SEARCH-KEY-EXIT
ELSE
    IF OPPTS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR 'K'
        COMPUTE H-NEW-COIN = H-LITEM-PYMT -
            H-TOTAL-LN-DEDUCT - H-LITEM-REIM
            - H-LN-BLOOD-DEDUCT
        MOVE 'Y' TO GJK-FLAG
        MOVE 1 TO H-DCP-CODE
        PERFORM 1455-SEARCH-KEY
            THRU 1455-SEARCH-KEY-EXIT
        MOVE 2 TO H-DCP-CODE
        ADD 1 TO W-DCP-MAX
        SET W-DCP-INDX TO W-DCP-MAX
        PERFORM 1475-STAGE-DCP-ENTRY
            THRU 1475-STAGE-DCP-ENTRY-EXIT
            UNTIL W-DCP-INDX = 1 OR
                H-DCP-STAGE NOT < W-DCP-STAGE (W-DCP-INDX - 1)
        MOVE LN-SUB TO W-DCP-SUB (W-DCP-INDX)
        MOVE H-DCP-STAGE TO W-DCP-STAGE (W-DCP-INDX)
        MOVE ZERO TO W-DCP-COIN1 (W-DCP-INDX)
        MOVE ZERO TO W-DCP-WGNAT (W-DCP-INDX)
        MOVE H-NEW-COIN TO W-DCP-COIN2 (W-DCP-INDX)
        MOVE OPPTS-SRVC-IND (LN-SUB) TO
            W-DCP-SRVC-IND (W-DCP-INDX).

1450-ADJ-PROC-COIN-EXIT.
    EXIT.

1455-SEARCH-KEY.

    SET W-DCP-INDX TO 1.
    SEARCH W-DCP-ENTRY VARYING W-DCP-INDX
        AT END
        PERFORM 1460-ADD-ENTRY
            THRU 1460-ADD-ENTRY-EXIT
        WHEN W-DCP-STAGE (W-DCP-INDX) = H-DCP-STAGE
            PERFORM 1465-UPDATE-ENTRY
                THRU 1465-UPDATE-ENTRY-EXIT.

1455-SEARCH-KEY-EXIT.
    EXIT.

1460-ADD-ENTRY.

    ADD 1 TO W-DCP-MAX.
    SET W-DCP-INDX TO W-DCP-MAX.
    PERFORM 1475-STAGE-DCP-ENTRY
        THRU 1475-STAGE-DCP-ENTRY-EXIT
        UNTIL W-DCP-INDX = 1 OR
            H-DCP-STAGE NOT < W-DCP-STAGE (W-DCP-INDX - 1).
    IF OPPTS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR 'K'
        MOVE ZERO TO W-DCP-SUB (W-DCP-INDX)
        MOVE H-DCP-STAGE TO W-DCP-STAGE (W-DCP-INDX)
        MOVE ZERO TO W-DCP-COIN1 (W-DCP-INDX)
        MOVE ZERO TO W-DCP-WGNAT (W-DCP-INDX)
        MOVE H-NEW-COIN TO W-DCP-COIN2 (W-DCP-INDX)

```

```
        MOVE 'X' TO W-DCP-SRVC-IND (W-DCP-INDX)
ELSE
    MOVE LN-SUB TO W-DCP-SUB (W-DCP-INDX)
    MOVE H-DCP-STAGE TO W-DCP-STAGE (W-DCP-INDX)
    MOVE H-NEW-COIN TO W-DCP-COIN1 (W-DCP-INDX)
    MOVE H-NEW-WGNAT TO W-DCP-WGNAT (W-DCP-INDX)
    MOVE ZERO TO W-DCP-COIN2 (W-DCP-INDX)
    MOVE OPPTS-SRVC-IND (LN-SUB) TO
        W-DCP-SRVC-IND (W-DCP-INDX).
```

```
1460-ADD-ENTRY-EXIT.
    EXIT.
```

```
1465-UPDATE-ENTRY.
```

```
    IF OPPTS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR 'K'
        ADD H-NEW-COIN TO W-DCP-COIN2 (W-DCP-INDX)
    ELSE
        IF W-DCP-SRVC-IND (W-DCP-INDX) = 'X'
            PERFORM 1485-REPLACE-TYPE1
                THRU 1485-REPLACE-TYPE1-EXIT
        ELSE
            PERFORM 1480-RANK-COIN
                THRU 1480-RANK-COIN-EXIT.
```

```
1465-UPDATE-ENTRY-EXIT.
    EXIT.
```

```
1475-STAGE-DCP-ENTRY.
```

```
    MOVE W-DCP-ENTRY (W-DCP-INDX - 1) TO
        W-DCP-ENTRY (W-DCP-INDX).
    SET W-DCP-INDX DOWN BY 1.
```

```
1475-STAGE-DCP-ENTRY-EXIT.
    EXIT.
```

```
1480-RANK-COIN.
```

```
    IF H-NEW-WGNAT > W-DCP-WGNAT (W-DCP-INDX)
        MOVE LN-SUB TO W-DCP-SUB (W-DCP-INDX)
        MOVE H-NEW-COIN TO W-DCP-COIN1 (W-DCP-INDX)
        MOVE H-NEW-WGNAT TO W-DCP-WGNAT (W-DCP-INDX)
        MOVE OPPTS-SRVC-IND (LN-SUB)
            TO W-DCP-SRVC-IND (W-DCP-INDX).
```

```
1480-RANK-COIN-EXIT.
    EXIT.
```

```
1485-REPLACE-TYPE1.
```

```
    MOVE LN-SUB TO W-DCP-SUB (W-DCP-INDX)
    MOVE H-NEW-COIN TO W-DCP-COIN1 (W-DCP-INDX)
    MOVE H-NEW-WGNAT TO W-DCP-WGNAT (W-DCP-INDX)
    MOVE OPPTS-SRVC-IND (LN-SUB)
        TO W-DCP-SRVC-IND (W-DCP-INDX).
```

1485-REPLACE-TYPE1-EXIT.  
EXIT.

1500-ADJ-CHRG.

IF (OPPS-SRVC-IND (LN-SUB) = 'S' OR 'T') AND  
(W-SUB-CHRG (W-LP-INDX) < 1.01)  
MOVE 'Y' TO ST0-FLAG.

IF (OPPS-SRVC-IND (LN-SUB) = 'S' OR 'T') AND  
(OPPS-PKG-FLAG (LN-SUB) = '0')  
COMPUTE H-TOT-ST-CHRG = W-SUB-CHRG (W-LP-INDX)  
+ H-TOT-ST-CHRG  
COMPUTE H-TOT-ST-PYMT = H-LITEM-PYMT  
+ H-TOT-ST-PYMT.

IF (OPPS-SRVC-IND (LN-SUB) = 'S' OR 'T' OR 'V'  
OR 'X' OR 'P') AND (OPPS-PKG-FLAG (LN-SUB) = '0')  
COMPUTE H-TOT-STVX-PYMT = H-LITEM-PYMT  
+ H-TOT-STVX-PYMT.

1500-ADJ-CHRG-EXIT.  
EXIT.

\*\*\*\*\*  
\* 1. APPLY DEDUCTIBLE TO HIGHEST NATIONAL COINSURANCE AMOUNT \*  
\* - DESCENDING UNTIL DEDUCTIBLE = 0. \*  
\* 2. CALCULATE THE STANDARD LINE PRICE \*  
\* (APC PAYMENT \* WAGE INDEX \* SERVICE UNITS \*  
\* \* DISCOUNT FACTOR) \*  
\* - WAGE ADJUST 60% OF THE APC PAYMENT ONLY \*  
\* 3. ADD LINE PRICE TO OUTLIER HOLD AREA. \*  
\* \*  
\* 4. CALCULATE THE LINE PRICE FOR DESIGNATED DEVICES \*  
\* AND DRUGS \*  
\*\*\*\*\*

1550-CALC-STANDARD.

COMPUTE H-MAX-COIN ROUNDED = (H-IP-LIMIT \*  
W-SRVC-UNITS (W-LP-INDX) \* W-DISC-RATE (W-LP-INDX)).  
IF OPPS-SRVC-IND (LN-SUB) = 'S' OR 'V'  
OR 'T' OR 'P' OR 'X' THEN  
COMPUTE H-LITEM-PYMT ROUNDED =  
(((W-APC-PYMT (W-LP-INDX) \* .60) \*  
W-WINX1 (W-LP-INDX))  
+ (W-APC-PYMT (W-LP-INDX) \* .40)) \*  
W-SRVC-UNITS (W-LP-INDX) \* W-DISC-RATE (W-LP-INDX)  
PERFORM 1560-CALC-BENE-DEDUCT  
THRU 1560-CALC-BENE-DEDUCT-EXIT

ELSE

IF OPPS-SRVC-IND (LN-SUB) = 'H' THEN  
IF OPPS-PYMT-IND (LN-SUB) = '6' THEN  
PERFORM 1555-CALC-H-STANDARD  
THRU 1555-CALC-H-STANDARD-EXIT  
PERFORM 1560-CALC-BENE-DEDUCT  
THRU 1560-CALC-BENE-DEDUCT-EXIT

ELSE

MOVE 44 TO A-RETURN-CODE (LN-SUB)

```

        GO TO 1550-CALC-STANDARD-EXIT
    END-IF
ELSE
    IF OPPTS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR 'K' THEN
        IF OPPTS-PYMT-IND (LN-SUB) = '1' OR '5' OR '7' THEN
            MOVE 0 TO H-BLOOD-FRACTION
            PERFORM 1550-CALC-GJK
                THRU 1550-CALC-GJK-EXIT
            PERFORM 1560-CALC-BENE-DEDUCT
                THRU 1560-CALC-BENE-DEDUCT-EXIT
        ELSE
            MOVE 41 TO A-RETURN-CODE (LN-SUB)
            GO TO 1550-CALC-STANDARD-EXIT
        END-IF
    END-IF.

IF H-LITEM-PYMT > 0
    COMPUTE H-LITEM-REIM ROUNDED =
        ((H-LITEM-PYMT - H-TOTAL-LN-DEDUCT) -
            H-LN-BLOOD-DEDUCT) * W-PPCT (W-LP-INDX)
    COMPUTE H-NAT-COIN = H-LITEM-PYMT -
        H-TOTAL-LN-DEDUCT - H-LITEM-REIM - H-LN-BLOOD-DEDUCT.
    MOVE W-MIN-COIN (W-LP-INDX) TO H-MIN-COIN.
    IF H-MIN-COIN > 0
        IF OPPTS-SRVC-IND (LN-SUB) = 'G' OR 'H' OR 'J' OR 'K'
            COMPUTE H-MIN-COIN ROUNDED = H-MIN-COIN *
                (W-SRVC-UNITS (W-LP-INDX) -
                    (W-SRVC-UNITS (W-LP-INDX) * H-BLOOD-FRACTION))
                * W-DISC-RATE (W-LP-INDX)
        ELSE
            IF OPPTS-APC (LN-SUB) = '0158' OR '0159'
                COMPUTE H-MIN-COIN ROUNDED = H-LITEM-PYMT * .25
            ELSE
                COMPUTE H-MIN-COIN ROUNDED = H-LITEM-PYMT * .2
            END-IF
        END-IF
    END-IF.

    MOVE W-RED-COIN (W-LP-INDX) TO H-RED-COIN.
    IF (H-RED-COIN > 0 AND < H-MIN-COIN)
        MOVE H-MIN-COIN TO H-RED-COIN
    ELSE
        IF H-RED-COIN < H-NAT-COIN AND > H-MIN-COIN
            AND > H-MAX-COIN
            COMPUTE H-LITEM-REIM = H-LITEM-REIM +
                (H-RED-COIN - H-MAX-COIN)
        END-IF
    END-IF.

    IF H-NAT-COIN > H-MAX-COIN AND H-RED-COIN = 0 THEN
        COMPUTE H-LITEM-REIM = H-LITEM-REIM +
            (H-NAT-COIN - H-MAX-COIN)
        MOVE H-MAX-COIN TO H-NAT-COIN.

1550-CALC-STANDARD-EXIT.
EXIT.

```

```

*****
* 1. CALCULATE LINE ITEM PAYMENT FOR SERVICE INDICATOR      *
*   TYPES G , J , OR K.                                     *
* 2. EFFECTIVE 04/01/2002 A PRO RATA REDUCTION APPLIES TO  *
*   ALL SERVICE INDICATOR 'G' PAYMENTS (CURRENTLY .636)    *
*****

```

1550-CALC-GJK.

```

IF OPSS-HCPCS(LN-SUB) = 'P9021' OR 'P9010' OR 'P9038'
    OR 'P9016' OR 'C1010' OR 'C1018' OR 'P9022'
    OR 'P9039' OR 'P9040' OR 'C1016'

```

```

    PERFORM 1550-SET-BLOOD-FRACTION
    THRU 1550-SET-BLOOD-FRACTION-EXIT

```

ELSE

```

IF (OPPS-SRVC-IND (LN-SUB) = 'G')
    COMPUTE H-LITEM-PYMT ROUNDED =
        (((W-APC-PYMT (W-LP-INDX) -
            (5 * W-MIN-COIN (W-LP-INDX))) * .364)
        + (5 * W-MIN-COIN (W-LP-INDX)))
        * W-SRVC-UNITS (W-LP-INDX)
        * W-DISC-RATE (W-LP-INDX)

```

```

    COMPUTE W-PPCT (W-LP-INDX) =
        (H-LITEM-PYMT - (W-NAT-COIN (W-LP-INDX) *
        W-SRVC-UNITS (W-LP-INDX) *
        W-DISC-RATE (W-LP-INDX))) / H-LITEM-PYMT
    GO TO 1550-CALC-GJK-EXIT

```

ELSE

```

    COMPUTE H-LITEM-PYMT ROUNDED =
        (W-APC-PYMT (W-LP-INDX) * W-SRVC-UNITS (W-LP-INDX))
        * W-DISC-RATE (W-LP-INDX)
    GO TO 1550-CALC-GJK-EXIT.

```

```

    COMPUTE H-LITEM-PYMT ROUNDED =
    W-BD-APC-PYMT (W-BD-INDX) * W-BD-SRVC-UNITS (W-BD-INDX)
        * W-BD-DISC-RATE (W-BD-INDX).

```

```

    COMPUTE H-LN-BLOOD-DEDUCT ROUNDED =
        H-LITEM-PYMT * H-BLOOD-FRACTION.

```

SET W-BD-INDX UP BY 1.

1550-CALC-GJK-EXIT.

EXIT.

1550-SET-BLOOD-FRACTION.

MOVE W-BD-SUB (W-BD-INDX) TO LN-SUB.

IF H-BENE-PINTS-USED > 0

IF W-BD-SRVC-UNITS (W-BD-INDX) <= H-BENE-PINTS-USED

MOVE 1 TO H-BLOOD-FRACTION

COMPUTE H-BENE-PINTS-USED =

H-BENE-PINTS-USED - W-BD-SRVC-UNITS (W-BD-INDX)

ELSE

IF W-BD-SRVC-UNITS (W-BD-INDX) > H-BENE-PINTS-USED

COMPUTE H-BLOOD-FRACTION =

H-BENE-PINTS-USED / W-BD-SRVC-UNITS (W-BD-INDX)

```

                MOVE 0 TO H-BENE-PINTS-USED
            ELSE
                MOVE 0 TO H-BLOOD-FRACTION
        ELSE
            MOVE 0 TO H-BLOOD-FRACTION.

1550-SET-BLOOD-FRACTION-EXIT.
EXIT.

*****
* 1. CALCULATE LINE ITEM PAYMENT FOR SERVICE INDICATOR      *
*   TYPE H.                                                  *
* 2. EFFECTIVE 04/01/2002 A PRO RATA REDUCTION APPLIES TO  *
*   ALL SERVICE INDICATOR 'H' PAYMENTS (CURRENTLY .689)    *
*****

1555-CALC-H-TOT.

MOVE W-LP-SUB (W-LP-INDX) TO LN-SUB.
MOVE OPSS-SUB-CHRG (LN-SUB) TO H-SUB-CHRG.
IF OPSS-SRVC-IND (LN-SUB) = 'H'
    IF OPSS-PYMT-IND (LN-SUB) = '6'
        COMPUTE H-TOT-H-CHRG =
            (H-TOT-H-CHRG + H-SUB-CHRG)
    ELSE
        NEXT SENTENCE
ELSE
    NEXT SENTENCE.

1555-CALC-H-TOT-EXIT.
EXIT.

1555-CALC-H-STANDARD.

MOVE OPSS-SUB-CHRG (LN-SUB) TO H-SUB-CHRG.

IF L-SERVICE-FROM-DATE > 20001231
    COMPUTE T-LITEM-PYMT ROUNDED =
        (H-SUB-CHRG * (L-PSF-OPCOST-RATIO * .981956))
        - (((W-APC-PYMT (W-LP-INDX) * .60) *
            W-WINX1 (W-LP-INDX))
            + (W-APC-PYMT (W-LP-INDX) * .40))
ELSE
    COMPUTE T-LITEM-PYMT ROUNDED =
        (H-SUB-CHRG * L-PSF-OPCOST-RATIO)
        - (((W-APC-PYMT (W-LP-INDX) * .60) *
            W-WINX1 (W-LP-INDX))
            + (W-APC-PYMT (W-LP-INDX) * .40)).

IF C-FLAG = 'Y' AND L-SERVICE-FROM-DATE > 20020331
    COMPUTE H-TOTAL-WAOFF ROUNDED =
        ((H-TOTAL-OFFSET * .60) * A-WINX)
        + (H-TOTAL-OFFSET * .40)
    PERFORM 1700-CALC-H-OFFSET
    THRU 1700-CALC-H-OFFSET-EXIT
ELSE

```

```

NEXT SENTENCE.

IF T-LITEM-PYMT < 0 THEN
    MOVE 0 TO H-LITEM-PYMT
ELSE
    MOVE T-LITEM-PYMT TO H-LITEM-PYMT.

1555-CALC-H-STANDARD-EXIT.
EXIT.

1560-CALC-BENE-DEDUCT.

IF OPSS-PYMT-ADJ-FLAG (LN-SUB) = '4'
    GO TO 1560-CALC-BENE-DEDUCT-EXIT.
IF H-BENE-DEDUCT > 0 THEN
    COMPUTE H-LN-BLD-PYMT =
        H-LITEM-PYMT - H-LN-BLOOD-DEDUCT
    IF H-BENE-DEDUCT <= H-LN-BLD-PYMT THEN
        MOVE H-BENE-DEDUCT TO H-TOTAL-LN-DEDUCT
        MOVE 0 TO H-BENE-DEDUCT
    ELSE
        COMPUTE H-BENE-DEDUCT =
            H-BENE-DEDUCT - H-LN-BLD-PYMT
        MOVE H-LN-BLD-PYMT TO H-TOTAL-LN-DEDUCT
        MOVE 20 TO A-RETURN-CODE (LN-SUB)
    END-IF
END-IF.

1560-CALC-BENE-DEDUCT-EXIT.
EXIT.

1600-ADJ-CHRG-OUTL.

MOVE W-LP-SUB (W-LP-INDX) TO LN-SUB.
IF OPSS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR
    'H' OR 'N'
    GO TO 1600-ADJ-CHRG-OUTL-EXIT.

IF (ST0-FLAG = 'Y') AND (OPSS-SRVC-IND (LN-SUB)
    = 'S' OR 'T') AND (H-TOT-ST-PYMT > 0)
    COMPUTE H-CHRG-RATE ROUNDED =
        (A-LITEM-PYMT (LN-SUB) / H-TOT-ST-PYMT)
    COMPUTE W-SUB-CHRG (W-LP-INDX) ROUNDED =
        (H-CHRG-RATE * H-TOT-ST-CHRG)
ELSE
    IF (N-FLAG = 'Y' AND ST0-FLAG = 'N') AND
        ((OPSS-SRVC-IND (LN-SUB) = 'S' OR 'T' OR 'V' OR
            'X' OR 'P') AND (OPSS-PKG-FLAG (LN-SUB) = '0'))
        AND (H-TOT-STVX-PYMT > 0)
        COMPUTE H-CHRG-RATE ROUNDED =
            (A-LITEM-PYMT (LN-SUB) / H-TOT-STVX-PYMT)
        COMPUTE H-SUB-CHRG ROUNDED =
            (H-CHRG-RATE * H-TOT-N-CHRG)
        COMPUTE W-SUB-CHRG (W-LP-INDX) ROUNDED =
            W-SUB-CHRG (W-LP-INDX) + H-SUB-CHRG.

```



```

IF (N-FLAG = 'Y' AND ST0-FLAG = 'Y') AND
  ((OPPS-SRVC-IND (LN-SUB) = 'S' OR 'T' OR 'V' OR
   'X' OR 'P') AND (OPPS-PKG-FLAG (LN-SUB) = '0'))
  AND (H-TOT-STVX-PYMT > 0)
  COMPUTE H-CHRG-RATE ROUNDED =
    (A-LITEM-PYMT (LN-SUB) / H-TOT-STVX-PYMT)
  COMPUTE H-SUB-CHRG ROUNDED =
    (H-CHRG-RATE * H-TOT-N-CHRG)
  COMPUTE W-SUB-CHRG (W-LP-INDX) ROUNDED =
    W-SUB-CHRG (W-LP-INDX) + H-SUB-CHRG.

```

```

COMPUTE H-LITEM-OUTL-PYMT ROUNDED =
  ((W-SUB-CHRG (W-LP-INDX) *
   (L-PSF-OPCOST-RATIO * .981956)) -
   (3.5 * A-LITEM-PYMT (LN-SUB))) * .50.

```

```

IF H-LITEM-OUTL-PYMT > 0
  COMPUTE H-OUTLIER-PYMT = H-OUTLIER-PYMT +
    H-LITEM-OUTL-PYMT.

```

```

1600-ADJ-CHRG-OUTL-EXIT.
EXIT.

```

```

*****
* 1. RE-CALCULATE LINE ITEM PAYMENT FOR SERVICE INDICATOR      *
*   TYPE H.                                                    *
* 2. SUBTRACT WAGE ADJUSTED OFFSET AMOUNT FROM SI TYPE 'H'   *
*   WITH HCPCS CODE BEGINNING WITH 'C'                        *
* 2. EFFECTIVE 04/01/2002                                     *
*****

```

```

1700-CALC-H-OFFSET.

```

```

IF H-TOT-H-CHRG > 0
  COMPUTE H-OFF-RATE ROUNDED =
    H-SUB-CHRG / H-TOT-H-CHRG
  COMPUTE T-LITEM-PYMT ROUNDED = (T-LITEM-PYMT -
    (H-TOTAL-WAOFF * H-OFF-RATE)) * .364
ELSE
  NEXT SENTENCE.

```

```

1700-CALC-H-OFFSET-EXIT.
EXIT.

```

```

1800-ADJ-STV-REIM.

```

```

IF W-DCP-CODE (W-DCP-INDX) = 1
  PERFORM 1810-PROCESS-TYPE1
    THRU 1810-PROCESS-TYPE1-EXIT
ELSE
  PERFORM 1840-PROCESS-TYPE2
    THRU 1840-PROCESS-TYPE2-EXIT.

```

```

1800-ADJ-STV-REIM-EXIT.
EXIT.

```

```

1810-PROCESS-TYPE1.

```

```

IF W-DCP-COIN2 (W-DCP-INDX) > 0
MOVE W-DCP-DOS (W-DCP-INDX) TO H-DCP-DOS
MOVE W-DCP-WGNAT (W-DCP-INDX) TO H-TOTAL
COMPUTE H-RATIO =
      (H-IP-LIMIT - W-DCP-WGNAT (W-DCP-INDX)) /
      W-DCP-COIN2 (W-DCP-INDX)
IF H-RATIO < 0
MOVE 0 TO H-RATIO.
IF H-RATIO > 1
MOVE 1 TO H-RATIO.

1810-PROCESS-TYPE1-EXIT.
EXIT.

1840-PROCESS-TYPE2.

IF W-DCP-DOS (W-DCP-INDX) >= 20010701
IF W-DCP-DOS (W-DCP-INDX) = H-DCP-DOS
MOVE W-DCP-SUB (W-DCP-INDX) TO LN-SUB
COMPUTE H-SHIFT =
      W-DCP-COIN2 (W-DCP-INDX) * (1 - H-RATIO)
COMPUTE H-TOTAL =
      A-ADJ-COIN (LN-SUB) + H-TOTAL - H-SHIFT
IF H-TOTAL > H-IP-LIMIT
COMPUTE H-SHIFT = H-SHIFT + H-TOTAL
      - H-IP-LIMIT
END-IF
COMPUTE A-ADJ-COIN (LN-SUB) =
      A-ADJ-COIN (LN-SUB) - H-SHIFT
COMPUTE A-LITEM-REIM (LN-SUB) =
      A-LITEM-REIM (LN-SUB) + H-SHIFT
MOVE 22 TO A-RETURN-CODE (LN-SUB)
END-IF
ELSE
IF W-DCP-DOS (W-DCP-INDX) >= 20010101
IF W-DCP-COIN2 (W-DCP-INDX) > H-IP-LIMIT
MOVE W-DCP-SUB (W-DCP-INDX) TO LN-SUB
COMPUTE H-SHIFT =
      W-DCP-COIN2 (W-DCP-INDX) - H-IP-LIMIT
COMPUTE A-ADJ-COIN (LN-SUB) =
      A-ADJ-COIN (LN-SUB) - H-SHIFT
COMPUTE A-LITEM-REIM (LN-SUB) =
      A-LITEM-REIM (LN-SUB) + H-SHIFT
MOVE 22 TO A-RETURN-CODE (LN-SUB)
END-IF
END-IF
END-IF.

1840-PROCESS-TYPE2-EXIT.
EXIT.

*****
* 1. CALCULATE CLAIM LEVEL OUTLIER AMOUNT *
*****
1900-END-PRICE-RTN.

```

```

MOVE H-TOT-CHRG TO A-TOT-CLM-CHRG.
MOVE H-TOT-PYMT TO A-TOT-CLM-PYMT.
COMPUTE A-BLOOD-PINTS-USED =
      H-BENE-BLOOD-PINTS - H-BENE-PINTS-USED.
*
* IF (A-TOT-CLM-PYMT > 0) AND
*   (L-SERVICE-FROM-DATE < 20020401)
*   PERFORM 1910-CALC-OUTLIER
*     THRU 1910-CALC-OUTLIER-EXIT.
IF H-OUTLIER-PYMT > 0
  MOVE H-OUTLIER-PYMT TO A-OUTLIER-PYMT.

1900-END-PRICE-RTN-EXIT.
EXIT.

* 1910-CALC-OUTLIER.
*
* IF L-SERVICE-FROM-DATE > 20001231
*   COMPUTE H-OUTLIER-PYMT ROUNDED =
*     ((H-TOT-CHRG * (L-PSF-OPCOST-RATIO * .981956))
*      - (2.5 * H-TOT-PYMT)) * .75
*   ELSE
*     COMPUTE H-OUTLIER-PYMT ROUNDED =
*       ((H-TOT-CHRG * L-PSF-OPCOST-RATIO)
*        - (2.5 * H-TOT-PYMT)) * .75.
*
* 1910-CALC-OUTLIER-EXIT.
*   EXIT.

*****
*   PROCESSING: AFTER 20021231
*   A. WILL PROCESS LINE ITEMS BASED ON OCE FLAGS.
*   B. INITIALIZE OPPTS HOLD VARIABLES.
*   C. EDIT THE DATA PASSED FROM THE OCE.
*   D. ASSEMBLE PRICING COMPONENTS.
*   E. CALCULATE THE PRICE.
*   F. RETURN COINSURANCE/REIMBURSEMENT/DEDUCTIBLE/
*       PAYMENT/OUTLIER AMOUNT/RETURN CODES
*****

2000-PROCESS-MAIN-NEW.

PERFORM 2100-INIT
      THRU 2100-INIT-EXIT.

IF A-CLM-RTN-CODE >= 50
  GOBACK.

MOVE H-WINX1 TO A-WINX.
PERFORM 2125-INIT
      THRU 2125-INIT-EXIT
      VARYING LN-SUB FROM 1 BY 1
      UNTIL LN-SUB > OPPTS-LINE-CNT.

MOVE 0 TO W-DCP-MAX W-BLD-MAX.
PERFORM 2150-INIT
      THRU 2150-INIT-EXIT
      VARYING LN-SUB FROM 1 BY 1

```

UNTIL LN-SUB > OPPS-LINE-CNT.

MOVE 0 TO W-DCP-MAX.  
PERFORM 2555-CALC-H-TOT  
THRU 2555-CALC-H-TOT-EXIT  
VARYING W-LP-INDX FROM 1 BY 1  
UNTIL W-LP-INDX > W-LNC-MAX.

SET W-BD-INDX TO 1.  
MOVE 0 TO W-DCP-MAX.  
PERFORM 2400-CALCULATE  
THRU 2400-CALCULATE-EXIT  
VARYING W-LP-INDX FROM 1 BY 1  
UNTIL W-LP-INDX > W-LNC-MAX.

PERFORM 2600-ADJ-CHRG-OUTL  
THRU 2600-ADJ-CHRG-OUTL-EXIT  
VARYING W-LP-INDX FROM 1 BY 1  
UNTIL W-LP-INDX > W-LNC-MAX

IF GJK-FLAG = 'Y'  
PERFORM 2800-ADJ-STV-REIM  
THRU 2800-ADJ-STV-REIM-EXIT  
VARYING W-DCP-INDX FROM 1 BY 1  
UNTIL W-DCP-INDX > W-DCP-MAX

ELSE  
NEXT SENTENCE.

PERFORM 2900-END-PRICE-RTN  
THRU 2900-END-PRICE-RTN-EXIT.

2000-PROCESS-MAIN-NEW-EXIT.  
EXIT.

```
*****
*   INITIALIZE WORKING STORAGE HOLD AREAS                               *
*   AND ADDITIONAL VARIABLES TO BE PASSED BACK TO                       *
*   THE STANDARD SYSTEM.                                                *
*   - IF PROVIDER SPECIFIC FILE WAGE INDEX RECLASSIFICATION            *
*   CODE INVALID OR MISSING                                             *
*     - MOVE '52' TO CLAIM LEVEL RETURN CODE                           *
*     - DISCONTINUE CLAIM PROCESSING                                    *
*   - IF SERVICE FROM DATE NOT NUMERIC OR < 20000801                   *
*     - MOVE '53' TO CLAIM LEVEL RETURN CODE                           *
*     - DISCONTINUE CLAIM PROCESSING                                    *
*   - IF SERVICE FROM DATE < PROVIDER EFFECTIVE DATE                   *
*     - MOVE '54' TO CLAIM LEVEL RETURN CODE                           *
*     - DISCONTINUE CLAIM PROCESSING                                    *
*   - IF SERVICE FROM DATE > PROVIDER TERMINATION DATE                 *
*     - MOVE '54' TO CLAIM LEVEL RETURN CODE                           *
*     - DISCONTINUE CLAIM PROCESSING                                    *
*****
```

2100-INIT.

MOVE 01 TO A-CLM-RTN-CODE.  
MOVE 'N' TO APC33-FLAG GJK-FLAG ST0-FLAG

```

                N-FLAG C-FLAG.
IF L-SERVICE-FROM-DATE NOT NUMERIC
    MOVE 53 TO A-CLM-RTN-CODE
    GO TO 2100-INIT-EXIT
ELSE
    IF L-SERVICE-FROM-DATE < L-PSF-EFFDT
        MOVE 54 TO A-CLM-RTN-CODE
        GO TO 2100-INIT-EXIT
    ELSE
        IF L-PSF-TERMDT > 0
            IF L-SERVICE-FROM-DATE > L-PSF-TERMDT
                MOVE 54 TO A-CLM-RTN-CODE
                GO TO 2100-INIT-EXIT
            END-IF
        END-IF.
MOVE SPACE TO A-MSA.
MOVE ZERO TO A-OUTLIER-PYMT
        A-TOTAL-CLM-DEDUCT
        A-TOT-CLM-CHRG
        A-TOT-CLM-PYMT
        A-BLOOD-DEDUCT-DUE
        A-BLOOD-PINTS-USED
        A-WINX
        W-LNC-MAX.
INITIALIZE H-ADDITIONAL-VARIABLES.
INITIALIZE LINE-HOLD-ITEMS.
INITIALIZE T-LITEM-PYMT.
IF L-SERVICE-FROM-DATE >= 20040101
    MOVE CAL-VERSION3 TO A-CALC-VERS
ELSE
    MOVE CAL-VERSION2 TO A-CALC-VERS.
MOVE BENE-DEDUCT TO H-BENE-DEDUCT.
MOVE BENE-BLOOD-PINTS TO H-BENE-BLOOD-PINTS
        H-BENE-PINTS-USED.
MOVE WAD-MAX TO WAD-SUB.
PERFORM UNTIL L-SERVICE-FROM-DATE
    NOT < WAD-DATE (WAD-SUB)
    SUBTRACT 1 FROM WAD-SUB
END-PERFORM.
IF L-PSF-WGIDX-RECLASS = 'Y'
    MOVE L-PSF-WI-MSA TO H-PSF-MSA
ELSE
    IF L-PSF-WGIDX-RECLASS = 'N'
        MOVE L-PSF-GEO-MSA TO H-PSF-MSA
    ELSE
        MOVE 52 TO A-CLM-RTN-CODE
        GO TO 2100-INIT-EXIT.

IF L-SERVICE-FROM-DATE >= 20040101
    MOVE 876 TO H-IP-LIMIT
ELSE
    IF L-SERVICE-FROM-DATE >= 20030101
        MOVE 840 TO H-IP-LIMIT.

IF L-SERVICE-FROM-DATE >= 20040101
    PERFORM 2120-FLOOR-2004
        THRU 2120-FLOOR-2004-EXIT

```

```

PERFORM 2120-SEC401-2004
  THRU 2120-SEC401-2004-EXIT
ELSE
  IF L-SERVICE-FROM-DATE >= 20030101
    PERFORM 2120-FLOOR-2003
      THRU 2120-FLOOR-2003-EXIT
    PERFORM 2120-SEC401-2003
      THRU 2120-SEC401-2003-EXIT.

  MOVE H-PSF-MSA TO A-MSA.

PERFORM 2230-CHNG-WAGEINDX
  THRU 2230-CHNG-WAGEINDX-EXIT.

  IF H-WINX1 = 0 THEN
    PERFORM 2200-CALC-WAGEINDX
      THRU 2200-CALC-WAGEINDX-EXIT.

2100-INIT-EXIT.
EXIT.

*****
*   RESET FLOOR MSA - 'FROM-DATE' CONTROLLED                               *
*   - YEAR 2003                                                                 *
*   - YEAR 2004                                                                 *
*****

2120-FLOOR-2003.

  IF H-PSF-MSA = ' 14'
    AND L-PSF-WGIDX-RECLASS = 'Y'
    AND L-PSF-PROV-ST = '16'
      MOVE 'N' TO L-PSF-WGIDX-RECLASS
      MOVE ' 16' TO H-PSF-MSA.

  IF H-PSF-MSA = '1123'
    AND L-PSF-PROV-ST = '22'
      MOVE ' 22' TO H-PSF-MSA.

  IF H-PSF-MSA = '1800'
    AND L-PSF-WGIDX-RECLASS = 'Y'
    AND L-PSF-PROV-ST = '11'
      MOVE 'N' TO L-PSF-WGIDX-RECLASS
      MOVE ' 11' TO H-PSF-MSA.

  IF H-PSF-MSA = '1900'
    AND L-PSF-PROV-ST = '21'
      MOVE ' 21' TO H-PSF-MSA.

  IF H-PSF-MSA = '2440'
    AND L-PSF-PROV-ST = '15'
      MOVE ' 15' TO H-PSF-MSA.

  IF H-PSF-MSA = '3660'
    AND L-PSF-PROV-ST = '49'
      MOVE 'N' TO L-PSF-WGIDX-RECLASS

```

```

        MOVE ' 49' TO H-PSF-MSA.

IF H-PSF-MSA = '3700'
  AND L-PSF-WGIDX-RECLASS = 'Y'
  AND L-PSF-PROV-ST = '26'
  MOVE 'N' TO L-PSF-WGIDX-RECLASS
  MOVE ' 26' TO H-PSF-MSA.

IF H-PSF-MSA = '6020'
  AND L-PSF-PROV-ST = '36'
  MOVE ' 36' TO H-PSF-MSA.

IF H-PSF-MSA = '9000'
  AND L-PSF-PROV-ST = '36'
  MOVE ' 36' TO H-PSF-MSA.

2120-FLOOR-2003-EXIT.
EXIT.

2120-SEC401-2003.

IF (L-PSF-PROV-OSCAR = '050192' OR '050286' OR
    '050446' OR '050469' OR
    '050528')
  MOVE ' 05' TO H-PSF-MSA
ELSE
  IF (L-PSF-PROV-OSCAR = '100048' OR '100118')
    MOVE ' 10' TO H-PSF-MSA
  ELSE
    IF (L-PSF-PROV-OSCAR = '170137')
      MOVE ' 17' TO H-PSF-MSA
    ELSE
      IF (L-PSF-PROV-OSCAR = '190048' OR '190110')
        MOVE ' 19' TO H-PSF-MSA
      ELSE
        IF (L-PSF-PROV-OSCAR = '230078')
          MOVE ' 23' TO H-PSF-MSA
        ELSE
          IF (L-PSF-PROV-OSCAR = '260006')
            MOVE ' 26' TO H-PSF-MSA
          ELSE
            IF (L-PSF-PROV-OSCAR = '300009')
              MOVE ' 30' TO H-PSF-MSA
            ELSE
              IF (L-PSF-PROV-OSCAR = '380084')
                MOVE ' 38' TO H-PSF-MSA.

2120-SEC401-2003-EXIT.
EXIT.

2120-FLOOR-2004.

IF H-PSF-MSA = ' 14'
  AND L-PSF-WGIDX-RECLASS = 'Y'
  AND L-PSF-PROV-ST = '16'
  MOVE 'N' TO L-PSF-WGIDX-RECLASS
  MOVE ' 16' TO H-PSF-MSA.

```

```
IF H-PSF-MSA = '0200'  
  AND L-PSF-WGIDX-RECLASS = 'Y'  
  AND L-PSF-PROV-ST = '06'  
    MOVE 'N' TO L-PSF-WGIDX-RECLASS  
    MOVE ' 06' TO H-PSF-MSA.  
  
IF H-PSF-MSA = '1480'  
  AND L-PSF-WGIDX-RECLASS = 'Y'  
  AND L-PSF-PROV-ST = '36'  
    MOVE 'N' TO L-PSF-WGIDX-RECLASS  
    MOVE ' 36' TO H-PSF-MSA.  
  
IF H-PSF-MSA = '1900'  
  AND L-PSF-PROV-ST = '21'  
    MOVE ' 21' TO H-PSF-MSA.  
  
IF H-PSF-MSA = '2440'  
  AND L-PSF-PROV-ST = '15'  
    MOVE ' 15' TO H-PSF-MSA.  
  
IF H-PSF-MSA = '2985'  
  AND L-PSF-PROV-ST = '24'  
    MOVE ' 24' TO H-PSF-MSA.  
  
IF H-PSF-MSA = '3660'  
  AND L-PSF-WGIDX-RECLASS = 'Y'  
  AND L-PSF-PROV-ST = '49'  
    MOVE 'N' TO L-PSF-WGIDX-RECLASS  
    MOVE ' 49' TO H-PSF-MSA.  
  
IF H-PSF-MSA = '3660'  
  AND L-PSF-PROV-ST = '49'  
    MOVE ' 49' TO H-PSF-MSA.  
  
IF H-PSF-MSA = '3700'  
  AND L-PSF-WGIDX-RECLASS = 'Y'  
  AND L-PSF-PROV-ST = '26'  
    MOVE 'N' TO L-PSF-WGIDX-RECLASS  
    MOVE ' 26' TO H-PSF-MSA.  
  
IF H-PSF-MSA = '6020'  
  AND L-PSF-PROV-ST = '36'  
    MOVE ' 36' TO H-PSF-MSA.  
  
IF H-PSF-MSA = '6740'  
  AND L-PSF-WGIDX-RECLASS = 'Y'  
  AND L-PSF-PROV-ST = '50'  
    MOVE 'N' TO L-PSF-WGIDX-RECLASS  
    MOVE ' 50' TO H-PSF-MSA.  
  
IF H-PSF-MSA = '7720'  
  AND L-PSF-WGIDX-RECLASS = 'Y'  
  AND L-PSF-PROV-ST = '28'  
    MOVE 'N' TO L-PSF-WGIDX-RECLASS  
    MOVE ' 28' TO H-PSF-MSA.
```



```
IF H-PSF-MSA = '8080'  
  AND L-PSF-PROV-ST = '36'  
    MOVE ' 36' TO H-PSF-MSA.
```

```
IF H-PSF-MSA = '9000'  
  AND L-PSF-PROV-ST = '36'  
    MOVE ' 36' TO H-PSF-MSA.
```

```
2120-FLOOR-2004-EXIT.  
EXIT.
```

```
2120-SEC401-2004.
```

```
IF (L-PSF-PROV-OSCAR = '050192' OR '050286' OR  
    '050469' OR '050528' OR  
    '050618')  
  MOVE ' 05' TO H-PSF-MSA  
ELSE  
  IF (L-PSF-PROV-OSCAR = '100048' OR '100118')  
    MOVE ' 10' TO H-PSF-MSA  
  ELSE  
    IF (L-PSF-PROV-OSCAR = '170137')  
      MOVE ' 17' TO H-PSF-MSA  
    ELSE  
      IF (L-PSF-PROV-OSCAR = '190048' OR '190110')  
        MOVE ' 19' TO H-PSF-MSA  
      ELSE  
        IF (L-PSF-PROV-OSCAR = '230078')  
          MOVE ' 23' TO H-PSF-MSA  
        ELSE  
          IF (L-PSF-PROV-OSCAR = '260006')  
            MOVE ' 26' TO H-PSF-MSA  
          ELSE  
            IF (L-PSF-PROV-OSCAR = '300009')  
              MOVE ' 30' TO H-PSF-MSA  
            ELSE  
              IF (L-PSF-PROV-OSCAR = '380084')  
                MOVE ' 38' TO H-PSF-MSA  
              ELSE  
                IF (L-PSF-PROV-OSCAR = '390106')  
                  MOVE ' 39' TO H-PSF-MSA.
```

```
2120-SEC401-2004-EXIT.  
EXIT.
```

```
*****  
* SET FLAG IF APC = 0033 *  
* - TERMINATE PROCESS IF 0033 LOCATED *  
* *  
*****
```

```
2125-INIT.
```

```
IF OPPTS-APC (LN-SUB) = '0033'  
  MOVE 'Y' TO APC33-FLAG  
  MOVE 451 TO LN-SUB.
```

2125-INIT-EXIT.  
EXIT.

```
*****
* 1. VERIFY THE SERVICE INDICATOR PASSED BY THE OCE *
* - IF INVALID SET RETURN CODE TO '40' *
* - DISCONTINUE LINE PROCESSING *
* - IF VALID SET RETURN CODE TO '01' *
* - CONTINUE LINE PROCESSING *
* 2. PROCESS LINES AND LOAD WORK TABLE ACCORDING TO PRICE *
* RANKING *
* 3. CHECK OCE EDIT INDICATORS AND SET RETURN CODES IF *
* INDICATORS ARE INVALID. *
* - VALID RETURN CODES FOR EDIT INDICATORS *
* - '41' - SERVICE INDICATOR INVALID FOR OPPTS PRICER *
* - '42' - APC = '00000' OR (PACKAGING FLAG = 1 OR 2) *
* - '43' - PAYMENT INDICATOR NOT = TO 1 OR 5 THRU 9 *
* - '44' - SERVICE INDICATOR = 'H' BUT PAYMENT *
* INDICATOR NOT = TO 6 *
* - '45' - PACKAGING FLAG NOT = TO 0 OR 1 OR 2 *
* - '46' - (LINE ITEM DENIAL/REJECT FLAG NOT = TO 0 *
* OR LINE ITEM DENIAL/REJECT FLAG = TO 1 *
* AND (APC NOT = 0033 OR 0034 OR 0322 OR *
* 0323 OR 0324 OR 0325 OR 0373 OR 0374)) *
* OR LINE ITEM ACTION FLAG NOT = TO 1 *
* - '47' - LINE ITEM ACTION FLAG = 2 OR 3 *
* - '48' - PAYMENT ADJUSTMENT FLAG NOT VALID *
* - '49' - SITE OF SERVICE FLAG NOT = TO 0 *
* - OR (APC 0033 IS NOT ON THE CLAIM *
* AND SERVICE INDICATOR = 'P' *
* OR APC = 0322-0325,0373,0374) *
* 4. IF OCE INDICATORS ARE VALID, SEARCH APC TABLE *
* - IF MISSING, DELETED OR INVALID APC *
* - SET RETURN CODE TO '30' *
* - DISCONTINUE LINE PROCESSING *
```

\*\*\*\*\*  
2150-INIT.

MOVE 01 TO A-RETURN-CODE (LN-SUB).

```
*****
* OVER-RIDE SERVICE UNITS FOR APC 0339 - EFFECTIVE 04/01/2002 *
* - CHANGE UNIT VALUE TO 1 *
```

```
*****
IF OPPTS-APC (LN-SUB) = '0339'
MOVE 1 TO OPPTS-SRVC-UNITS (LN-SUB).
```

```
*****
* OVER-RIDE SERVICE INDICATOR FOR - EFFECTIVE 01/01/2004 *
* - CHANGE INDICATOR TO 'N' AND MOVE SPACES TO APC *
* - OCE CORRECTION WILL REPLACE THIS OVERRIDE 04/01/2004 *
*****
* IF ((L-SERVICE-FROM-DATE > 20031231) AND *
* (L-SERVICE-FROM-DATE < 20040401)) AND *
* (OPPTS-HCPCS (LN-SUB) = 'Q4078' OR 'A9526') *
* MOVE 'N' TO OPPTS-SRVC-IND (LN-SUB)
```

```

*          MOVE '00000' TO OPPTS-GRP (LN-SUB)
*          MOVE '00000' TO OPPTS-HCPCS-APC (LN-SUB).
*****

*****
* OVER-RIDE SERVICE INDICATOR FOR          - EFFECTIVE 01/01/2004 *
* - CHANGE AS OF 04/01/2004 *
* - CHANGE INDICATOR TO 'N' AND MOVE ZEROS TO APC 0738 *
* - CHANGE PACKAGE FLAG TO '1' *
* - OCE CORRECTION WILL REPLACE THIS OVERRIDE 07/01/2004 *
*****
      IF ((L-SERVICE-FROM-DATE > 20031231) AND
          (L-SERVICE-FROM-DATE < 20040401)) AND
          (OPPTS-APC (LN-SUB) = '0738')
          MOVE 'N' TO OPPTS-SRVC-IND (LN-SUB)
          MOVE '00000' TO OPPTS-GRP (LN-SUB)
          MOVE '00000' TO OPPTS-HCPCS-APC (LN-SUB)
          MOVE '1' TO OPPTS-PKG-FLAG (LN-SUB).
*****

      MOVE OPPTS-SRVC-UNITS (LN-SUB) TO H-SRVC-UNITS.
      PERFORM 2250-CALC-DISCOUNT
              THRU 2250-CALC-DISCOUNT-EXIT.

      IF A-RETURN-CODE (LN-SUB) = 42
          GO TO 2150-INIT-EXIT.

*****
* EFFECTIVE AS OF 04-01-2002 *
* - TOTAL DEVICE OFFSET *
*****

      IF OPPTS-SRVC-IND (LN-SUB) = 'H'
          MOVE 'Y' TO C-FLAG
          COMPUTE H-TOT-HTD-UNITS = H-TOT-HTD-UNITS
              + H-SRVC-UNITS.

      PERFORM 2160-TOTAL-OFFSET
              THRU 2160-TOTAL-OFFSET-EXIT.

      SET W-LP-INDX TO LN-SUB.
      SET W-BD-INDX TO LN-SUB.
      INITIALIZE W-LP-ENTRY (W-LP-INDX).
      INITIALIZE W-BD-ENTRY (W-BD-INDX).
      MOVE ZERO TO A-LITEM-PYMT (LN-SUB)
          A-LITEM-REIM (LN-SUB)
          A-ADJ-COIN (LN-SUB)
          A-RED-COIN (LN-SUB)
          A-TOTAL-LN-DEDUCT (LN-SUB)
          A-BLOOD-LN-DEDUCT (LN-SUB).

      IF NOT (OPPTS-SRVC-IND (LN-SUB) = 'A' OR 'B' OR 'C' OR
          'E' OR 'F' OR 'G' OR 'H' OR 'J' OR 'K' OR 'L' OR 'N'
          OR 'P' OR 'S' OR 'T' OR 'V' OR 'X' OR 'W' OR 'Y' OR
          'Z') THEN
          MOVE 40 TO A-RETURN-CODE (LN-SUB)
          GO TO 2150-INIT-EXIT

```

```

ELSE
  MOVE 01 TO A-RETURN-CODE (LN-SUB).

IF OPSS-SRVC-IND (LN-SUB) = 'A' OR 'B' OR 'C' OR 'E'
  OR 'F' OR 'L' OR 'W' OR 'Y' OR 'Z'
  MOVE 41 TO A-RETURN-CODE (LN-SUB)
  GO TO 2150-INIT-EXIT.

IF OPSS-PYMT-IND (LN-SUB) = '1' OR '5' OR '6' OR
  '7' OR '8' OR '9'
  IF OPSS-PKG-FLAG (LN-SUB) = '0' OR '1' OR '2'
    OR '3'
    IF (OPSS-LITEM-DR-FLAG (LN-SUB) = '0' OR
      OPSS-LITEM-DR-FLAG (LN-SUB) = '1' AND
      (OPSS-APC (LN-SUB) = '0033' OR '0034'
      OR '0322' OR '0323' OR '0324' OR '0325'
      OR '0373' OR '0374')) OR
      (OPSS-LITEM-ACT-FLAG (LN-SUB) = '1')
      IF NOT (OPSS-LITEM-ACT-FLAG (LN-SUB)
        EQUAL '2' OR '3')
        IF OPSS-PYMT-ADJ-FLAG (LN-SUB) = '0' OR '1'
          OR '2' OR '3' OR '4'
          IF (OPSS-SITE-SRVC-FLAG (LN-SUB) = '0') OR
            ((APC33-FLAG = 'Y') AND
              ((OPSS-SRVC-IND (LN-SUB) = 'P') OR
                (OPSS-APC (LN-SUB) = '0322' OR '0323'
                OR '0324' OR '0325' OR '0373'
                OR '0374'))))
            MOVE OPSS-SUB-CHRG (LN-SUB) TO H-SUB-CHRG
            COMPUTE H-TOT-CHRG = H-TOT-CHRG +
              H-SUB-CHRG

            IF OPSS-LITEM-ACT-FLAG (LN-SUB) = '4'
              COMPUTE H-TOT-CHRG = H-TOT-CHRG -
                H-SUB-CHRG
            END-IF
            IF OPSS-PKG-FLAG (LN-SUB) = '1' OR '2'
              COMPUTE H-TOT-N-CHRG = H-SUB-CHRG
                + H-TOT-N-CHRG
              MOVE 'Y' TO N-FLAG
            END-IF
            IF (OPSS-APC (LN-SUB) = '0000') OR
              (OPSS-PKG-FLAG (LN-SUB) = '1' OR '2')
              MOVE 42 TO A-RETURN-CODE (LN-SUB)
              GO TO 2150-INIT-EXIT
            END-IF
          SEARCH ALL WAA-ENTRY
          AT END
            MOVE 30 TO A-RETURN-CODE (LN-SUB)
            GO TO 2150-INIT-EXIT
          WHEN WAA-APC (WAA-INDX) = OPSS-GRP (LN-SUB)
            MOVE WAA-PTR (WAA-INDX) TO W-SUB2
            PERFORM 2175-APC-LOOKUP
          ELSE
            MOVE 49 TO A-RETURN-CODE (LN-SUB)
            GO TO 2150-INIT-EXIT
          ELSE

```

```

                MOVE 48 TO A-RETURN-CODE (LN-SUB)
                GO TO 2150-INIT-EXIT
            ELSE
                MOVE 47 TO A-RETURN-CODE (LN-SUB)
                GO TO 2150-INIT-EXIT
            ELSE
                MOVE 46 TO A-RETURN-CODE (LN-SUB)
                GO TO 2150-INIT-EXIT
            ELSE
                MOVE 45 TO A-RETURN-CODE (LN-SUB)
                GO TO 2150-INIT-EXIT
            ELSE
                MOVE 43 TO A-RETURN-CODE (LN-SUB)
                GO TO 2150-INIT-EXIT.

IF (A-RETURN-CODE (LN-SUB) = 01) AND
(OPPS-HCPCS (LN-SUB) = 'C9114' OR 'C9115')
PERFORM 2180-MOD-CCODE-PYMT
THRU 2180-MOD-CCODE-PYMT-EXIT.

IF A-RETURN-CODE (LN-SUB) = 01
PERFORM 2300-COIN-DEDUCT
THRU 2300-COIN-DEDUCT-EXIT.

IF A-RETURN-CODE (LN-SUB) = 01
SET WBD-INDX TO 1
SEARCH WBD-ENTRY VARYING WBD-INDX
AT END
GO TO 2150-INIT-EXIT
WHEN W-BLOOD-HCPCS (WBD-INDX) = OPPS-HCPCS (LN-SUB)
MOVE W-BLOOD-RANK (WBD-INDX) TO H-BLOOD-RANK
PERFORM 2375-BLOOD-DEDUCT
THRU 2375-BLOOD-DEDUCT-EXIT.

2150-INIT-EXIT.
EXIT.

*****
*   COMPUTE TOTAL OFFSET FROM TABLE 5 FOR DEVICES           *
*   - EFFECTIVE AS OF 01-01-2003                             *
*   - EFFECTIVE AS OF 01-01-2004                             *
*   - SEARCH TABLE OPPSOF04                                 *
*   - WHERE ALL OFFSET VALUES EQUAL ZERO                   *
*****
2160-TOTAL-OFFSET.

MOVE OPPS-GRP (LN-SUB) TO W-OFF-APC.
IF L-SERVICE-FROM-DATE >= 20040101
SEARCH ALL WOO-ENTRY3
AT END
GO TO 2160-TOTAL-OFFSET-EXIT
WHEN WOO-APC3 (WOO-INDX3) = W-OFF-APC
COMPUTE H-TOTAL-OFFSET = H-TOTAL-OFFSET
+ (WOO-OFFSET3 (WOO-INDX3) * H-DISC-RATE
* H-SRVC-UNITS)
COMPUTE H-TOT-OFF-UNITS = H-TOT-OFF-UNITS
+ H-SRVC-UNITS

```

```

ELSE
  SEARCH ALL WOO-ENTRY2
  AT END
  GO TO 2160-TOTAL-OFFSET-EXIT
  WHEN WOO-APC2 (WOO-INDX2) = W-OFF-APC
  COMPUTE H-TOTAL-OFFSET = H-TOTAL-OFFSET
    + (WOO-OFFSET2 (WOO-INDX2) * H-DISC-RATE
      * H-SRVC-UNITS)
  COMPUTE H-TOT-OFF-UNITS = H-TOT-OFF-UNITS
    + H-SRVC-UNITS.

```

```

2160-TOTAL-OFFSET-EXIT.
EXIT.

```

```

*****
* SEARCH APC TABLE - MOVE VALUES TO HOLD AREA FOR PROCESSING *
* - ADJUST TOTAL CHARGE FOR DELETED APC'S *
*****
2175-APC-LOOKUP.

```

```

IF WAR-DTCD (W-SUB2) NOT > WAD-DTCD (WAD-SUB)
IF WAR-RATEX (W-SUB2) = 'DELETED'
  MOVE 30 TO A-RETURN-CODE (LN-SUB)
  COMPUTE H-TOT-CHRG = H-TOT-CHRG -
    H-SUB-CHRG

```

```

ELSE
  MOVE WAR-RATE (W-SUB2) TO H-APC-PYMT
  MOVE WAR-RANK (W-SUB2) TO H-RANK
  MOVE WAR-MINC (W-SUB2) TO H-MIN-COIN
  MOVE WAR-COIN (W-SUB2) TO H-NAT-COIN
  MOVE WAR-PPCT (W-SUB2) TO H-PPCT

```

```

ELSE
  SUBTRACT 1 FROM W-SUB2
  IF W-SUB2 > WAA-PTR (WAA-INDX - 1)
  GO TO 2175-APC-LOOKUP
  ELSE
  MOVE 0 TO H-APC-PYMT
    H-RANK
    H-MIN-COIN
    H-NAT-COIN
    H-PPCT.

```

```

2175-APC-LOOKUP-EXIT.
EXIT.

```

```

*****
* IF DATE OF SERVICE BETWEEN 12/31/2002 AND 04/01/2003 *
* IF HCPCS CODE = C9114 OR C9115 ADJUST PAYMENT AND *
* COINSURANCE AMOUNTS *
*****
2180-MOD-CCODE-PYMT.

```

```

IF L-SERVICE-FROM-DATE > 20021231 AND
L-SERVICE-FROM-DATE < 20030401
IF OPPTS-HCPCS (LN-SUB) = 'C9114'
  COMPUTE H-APC-PYMT = H-APC-PYMT * 3
  COMPUTE H-NAT-COIN = H-NAT-COIN * 3

```

```

        COMPUTE H-MIN-COIN = H-MIN-COIN * 3
    ELSE
        IF OPPTS-HCPCS (LN-SUB) = 'C9115'
            COMPUTE H-APC-PYMT = H-APC-PYMT * 2
            COMPUTE H-NAT-COIN = H-NAT-COIN * 2
            COMPUTE H-MIN-COIN = H-MIN-COIN * 2
        ELSE
            NEXT SENTENCE.

2180-MOD-CCODE-PYMT-EXIT.
    EXIT.

*****
*   SEARCH WAGE INDEX TABLE AND SELECT THE WAGE INDEX           *
*   WHEN EQUAL TO PROVIDER SPECIFIC WAGE INDEX                   *
*   IF WAGE INDEX NOT LOCATED                                     *
*       - SET CLAIM RETURN CODE TO '50'                          *
*       - DISCONTINUE CLAIM PROCESSING                           *
*   IF WAGE INDEX EQUALS ZERO                                     *
*       - SET CLAIM RETURN CODE TO '51'                          *
*       - DISCONTINUE CLAIM PROCESSING                           *
*****
2200-CALC-WAGEINDX.

    MOVE WWD-MAX TO WWD-SUB.
    PERFORM UNTIL L-SERVICE-FROM-DATE NOT < WWD-DATE (WWD-SUB)
        SUBTRACT 1 FROM WWD-SUB
    END-PERFORM.

    SEARCH ALL WWM-ENTRY
    AT END
        MOVE 50 TO A-CLM-RTN-CODE
        GO TO 2200-CALC-WAGEINDX-EXIT
    WHEN WWM-MSA (WWM-INDX) = H-PSF-MSA
        MOVE WWM-PTR (WWM-INDX) TO W-SUB2
        PERFORM 2210-WAGE-LOOKUP.

    IF H-WINX1 = 0 THEN
        MOVE 51 TO A-CLM-RTN-CODE.

2200-CALC-WAGEINDX-EXIT.
    EXIT.

2210-WAGE-LOOKUP.

    IF WWW-DTCD (W-SUB2) NOT > WWD-DTCD (WWD-SUB)
        IF L-PSF-WGIDX-RECLASS = 'Y'
            MOVE WWW-WINX2 (W-SUB2) TO H-WINX1
        ELSE
            MOVE WWW-WINX1 (W-SUB2) TO H-WINX1
        END-IF
    ELSE
        SUBTRACT 1 FROM W-SUB2
        IF W-SUB2 > WWM-PTR (WWM-INDX - 1)
            GO TO 2210-WAGE-LOOKUP
        ELSE
            MOVE 0 TO H-WINX1.

```

2210-WAGE-LOOKUP-EXIT.  
EXIT.

\*\*\*\*\*  
\* FOR FY 2003 NEW LUGAR HOSPITALS ONLY \*  
\*\*\*\*\*

2230-CHNG-WAGEINDX.

IF (L-SERVICE-FROM-DATE > 20021231 AND  
L-SERVICE-FROM-DATE < 20040101)  
IF (L-PSF-PROV-OSCAR = '110130')  
AND (L-PSF-WI-MSA = ' 11'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 00.8230 TO H-WINX1.

IF (L-SERVICE-FROM-DATE > 20031231 AND  
L-SERVICE-FROM-DATE < 20050101)  
IF (L-PSF-PROV-OSCAR = '330001' OR '330126' OR  
'330135' OR '330205' OR '330209' OR '330264')  
AND (L-PSF-WI-MSA = '5600'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 01.3892 TO H-WINX1.

IF (L-SERVICE-FROM-DATE > 20031231 AND  
L-SERVICE-FROM-DATE < 20050101)  
IF (L-PSF-PROV-OSCAR = '470003')  
AND (L-PSF-WI-MSA = '1123'  
AND L-PSF-WGIDX-RECLASS = 'Y')  
MOVE 01.1120 TO H-WINX1.

2230-CHNG-WAGEINDX-EXIT.  
EXIT.

\*\*\*\*\*  
\* CALCULATE DISCOUNT FACTOR BASED ON THE DISCOUNT \*  
\* INDICATOR PASSED BY THE OCE: VALUE 1 - 8 \*  
\* IF MISSING OR INVALID DISCOUNT FACTOR \*  
\* - SET RETURN CODE TO '38' \*  
\* - DISCONTINUE LINE PROCESSING \*  
\*\*\*\*\*

2250-CALC-DISCOUNT.

IF (H-SRVC-UNITS = 0 AND  
OPPS-APC (LN-SUB) = '0000')  
MOVE 42 TO A-RETURN-CODE (LN-SUB)  
GO TO 2250-CALC-DISCOUNT-EXIT  
ELSE  
IF (H-SRVC-UNITS = 0 AND  
OPPS-APC (LN-SUB) > '0000')  
MOVE 1 TO H-SRVC-UNITS.

IF OPPS-DISC-FACT (LN-SUB) = 1 THEN  
MOVE 1 TO H-DISC-RATE  
ELSE  
IF OPPS-DISC-FACT (LN-SUB) = 2 THEN



```

      COMPUTE H-DISC-RATE = (1 + DISC-FRACTION *
        (H-SRVC-UNITS - 1)) / H-SRVC-UNITS
ELSE
  IF OPPTS-DISC-FACT (LN-SUB) = 3 THEN
    COMPUTE H-DISC-RATE = TERM-PROC-DISC
      / H-SRVC-UNITS
ELSE
  IF OPPTS-DISC-FACT (LN-SUB) = 4 THEN
    COMPUTE H-DISC-RATE = (1 + DISC-FRACTION)
      / H-SRVC-UNITS
ELSE
  IF OPPTS-DISC-FACT (LN-SUB) = 5 THEN
    COMPUTE H-DISC-RATE = DISC-FRACTION
ELSE
  IF OPPTS-DISC-FACT (LN-SUB) = 6 THEN
    COMPUTE H-DISC-RATE = (TERM-PROC-DISC *
      DISC-FRACTION) / H-SRVC-UNITS
ELSE
  IF OPPTS-DISC-FACT (LN-SUB) = 7 THEN
    COMPUTE H-DISC-RATE = (DISC-FRACTION *
      (1 + DISC-FRACTION) / H-SRVC-UNITS)
ELSE
  IF OPPTS-DISC-FACT (LN-SUB) = 8 THEN
    MOVE 2 TO H-DISC-RATE
  ELSE
    MOVE 38 TO A-RETURN-CODE (LN-SUB).

```

2250-CALC-DISCOUNT-EXIT.  
EXIT.

```

*****
*   DETERMINE THE DEDUCTIBLE SEQUENCE. DEDUCTIBLE WILL BE   *
*   TAKEN FROM OPPTS SERVICES FIRST, THEN FROM ANY OTHER   *
*   TYPES OF SERVICES FROM THE CLAIM.                       *
*   - SET POINTERS TO THE HIGHEST RANKED PERCENTAGE        *
*   (LARGEST NATIONAL UNADJUSTED COINSURANCE /             *
*   THE APC PAYMENT)                                       *
*   - MOVE ALL PRICING VARIABLES TO STAGING AREA           *
*****

```

2300-COIN-DEDUCT.

```

ADD 1 TO W-LNC-MAX.
SET W-LP-INDX TO W-LNC-MAX.
PERFORM 2350-STAGE-ENTRY
  THRU 2350-STAGE-ENTRY-EXIT
  UNTIL W-LP-INDX = 1 OR
    H-RANK NOT < W-RANK (W-LP-INDX - 1).
MOVE LN-SUB TO W-LP-SUB (W-LP-INDX).
MOVE H-NAT-COIN TO W-NAT-COIN (W-LP-INDX).
MOVE H-MIN-COIN TO W-MIN-COIN (W-LP-INDX).
MOVE H-SUB-CHRG TO W-SUB-CHRG (W-LP-INDX).
MOVE H-APC-PYMT TO W-APC-PYMT (W-LP-INDX).
MOVE H-WINX1 TO W-WINX1 (W-LP-INDX).
MOVE H-RANK TO W-RANK (W-LP-INDX).
MOVE H-PPCT TO W-PPCT (W-LP-INDX).
MOVE H-DISC-RATE TO W-DISC-RATE (W-LP-INDX).

```

```

IF OPPTS-SRVC-IND (LN-SUB) = 'P' THEN
    MOVE 1 TO W-SRVC-UNITS (W-LP-INDX)
ELSE
    MOVE H-SRVC-UNITS TO W-SRVC-UNITS (W-LP-INDX).

MOVE 0 TO W-RED-COIN (W-LP-INDX).
PERFORM VARYING PS-SUB FROM 1 BY 1
    UNTIL PS-SUB > L-PSF-APC-LINE-CNT
        IF L-PSF-APC (PS-SUB) = OPPTS-APC (LN-SUB)
            COMPUTE W-RED-COIN (W-LP-INDX) ROUNDED =
                L-PSF-RED-COIN (PS-SUB) * H-SRVC-UNITS * H-DISC-RATE
            MOVE 25 TO A-RETURN-CODE (LN-SUB)
            MOVE L-PSF-APC-LINE-CNT TO PS-SUB
        END-IF
    END-PERFORM.

```

```

2300-COIN-DEDUCT-EXIT.
EXIT.

```

```

2350-STAGE-ENTRY.

```

```

    MOVE W-LP-ENTRY (W-LP-INDX - 1) TO
        W-LP-ENTRY (W-LP-INDX).
    SET W-LP-INDX DOWN BY 1.

```

```

2350-STAGE-ENTRY-EXIT.
EXIT.

```

```

*****
*   DETERMINE THE BLOOD DEDUCTIBLE SEQUENCE. DEDUCTIBLE   *
*   WILL BE CALCULATED BASED ON RANKING.                   *
*   - SET POINTERS TO THE HIGHEST RANKED APC               *
*   (SMALLEST UNADJUSTED APC PAYMENT)                     *
*   - MOVE ALL PRICING VARIABLES TO STAGING AREA           *
*****

```

```

2375-BLOOD-DEDUCT.

```

```

    ADD 1 TO W-BLD-MAX.
    SET W-BD-INDX TO W-BLD-MAX.
    PERFORM 2385-STAGE-ENTRY
        THRU 2385-STAGE-ENTRY-EXIT
        UNTIL W-BD-INDX = 1 OR
            H-BLOOD-RANK NOT < W-BD-RANK (W-BD-INDX - 1).
    MOVE LN-SUB TO W-BD-SUB (W-BD-INDX).
    MOVE H-NAT-COIN TO W-BD-NAT-COIN (W-BD-INDX).
    MOVE H-MIN-COIN TO W-BD-MIN-COIN (W-BD-INDX).
    MOVE H-SUB-CHRG TO W-BD-SUB-CHRG (W-BD-INDX).
    MOVE H-APC-PYMT TO W-BD-APC-PYMT (W-BD-INDX).
    MOVE H-WINX1    TO W-BD-WINX1 (W-BD-INDX).
    MOVE H-BLOOD-RANK TO W-BD-RANK (W-BD-INDX).
    MOVE H-PPCT      TO W-BD-PPCT (W-BD-INDX).
    MOVE H-DISC-RATE TO W-BD-DISC-RATE (W-BD-INDX).
    MOVE H-SRVC-UNITS TO W-BD-SRVC-UNITS (W-BD-INDX).

```

```

    MOVE 0 TO W-BD-RED-COIN (W-BD-INDX).
    PERFORM VARYING PS-SUB FROM 1 BY 1
        UNTIL PS-SUB > L-PSF-APC-LINE-CNT

```

```
IF L-PSF-APC (PS-SUB) = OPPS-APC (LN-SUB)
  COMPUTE W-BD-RED-COIN (W-BD-INDX) ROUNDED =
  L-PSF-RED-COIN (PS-SUB) * H-SRVC-UNITS * H-DISC-RATE
  MOVE 25 TO A-RETURN-CODE (LN-SUB)
  MOVE L-PSF-APC-LINE-CNT TO PS-SUB
END-IF
END-PERFORM.
```

```
2375-BLOOD-DEDUCT-EXIT.
EXIT.
```

```
2385-STAGE-ENTRY.
```

```
MOVE W-BD-ENTRY (W-BD-INDX - 1) TO
  W-BD-ENTRY (W-BD-INDX).
SET W-BD-INDX DOWN BY 1.
```

```
2385-STAGE-ENTRY-EXIT.
EXIT.
```

```
*****
* FIRST STEP IN DETERMINING A LINE ITEM PRICE. *
* CHECK ALL APPROPRIATE FLAGS AND INDICATORS PASSED *
* BY THE OCE. *
* - SET RETURN CODES TO INDICATE ERRORS OR STATUS *
* - '20' - LINE PROCESSED BUT PAYMENT = 0 *
* - BENE DEDUCTIBLE => ADJUSTED PAYMENT *
* *
*****
```

```
2400-CALCULATE.
```

```
MOVE W-LP-SUB (W-LP-INDX) TO LN-SUB.
IF A-RETURN-CODE (LN-SUB) > 25
  GO TO 2400-CALCULATE-EXIT.
```

```
IF OPPS-PYMT-IND (LN-SUB) = '1' OR '5'
  OR '6' OR '7' OR '8'
  PERFORM 2550-CALC-STANDARD
  THRU 2550-CALC-STANDARD-EXIT
ELSE
  GO TO 2400-CALCULATE-EXIT.
```

```
*****
* SET GJK FLAG *
* - TEST LINE ITEM DATE OF SERVICE > 20010630 *
* - TOTAL DRUG / DEVICE COINSURANCE *
*****
```

```
IF (A-RETURN-CODE (LN-SUB) < 30)
  PERFORM 2450-ADJ-PROC-COIN
  THRU 2450-ADJ-PROC-COIN-EXIT
ELSE
  NEXT SENTENCE.
```

```
*****
* SET ST0 AND STVX FLAGS *
* - TEST LINE ITEM DATE OF SERVICE > 20020331 *
```

\* - TOTAL PROCEDURE CHARGES AND BUNDLED CHARGES \*  
\*\*\*\*\*

PERFORM 2500-ADJ-CHRG  
THRU 2500-ADJ-CHRG-EXIT.

IF A-RETURN-CODE (LN-SUB) < 30  
  COMPUTE A-TOTAL-CLM-DEDUCT =  
    H-TOTAL-LN-DEDUCT + A-TOTAL-CLM-DEDUCT  
  COMPUTE H-TOT-PYMT = H-TOT-PYMT + H-LITEM-PYMT  
  COMPUTE A-BLOOD-DEDUCT-DUE =  
    A-BLOOD-DEDUCT-DUE + H-LN-BLOOD-DEDUCT  
  MOVE H-LITEM-PYMT TO A-LITEM-PYMT (LN-SUB)  
  MOVE H-LITEM-REIM TO A-LITEM-REIM (LN-SUB)  
  MOVE H-TOTAL-LN-DEDUCT TO A-TOTAL-LN-DEDUCT (LN-SUB)  
  MOVE H-LN-BLOOD-DEDUCT TO A-BLOOD-LN-DEDUCT (LN-SUB)  
  MOVE H-NAT-COIN TO A-ADJ-COIN (LN-SUB)  
  MOVE H-RED-COIN TO A-RED-COIN (LN-SUB)  
  IF H-RED-COIN > H-NAT-COIN  
    MOVE H-NAT-COIN TO A-RED-COIN (LN-SUB)  
  END-IF  
  IF OPSS-LITEM-ACT-FLAG (LN-SUB) = '4'  
    MOVE 0 TO A-LITEM-PYMT (LN-SUB)  
    MOVE 0 TO A-LITEM-REIM (LN-SUB)  
    COMPUTE H-TOT-PYMT = H-TOT-PYMT - H-LITEM-PYMT  
  END-IF  
END-IF.  
MOVE ZERO TO LINE-HOLD-ITEMS.

2400-CALCULATE-EXIT.  
EXIT.

\*\*\*\*\*  
\* SET GJK FLAG \*  
\* - STAGE BY SERVICE INDICATOR \*  
\* \*  
\*\*\*\*\*

2450-ADJ-PROC-COIN.

MOVE OPSS-LITEM-DOS (LN-SUB) TO H-DCP-DOS.  
IF OPSS-SRVC-IND (LN-SUB) = 'S' OR 'T' OR 'V'  
  COMPUTE H-NEW-WGNAT ROUNDED =  
    W-NAT-COIN (W-LP-INDX) \*  
    (.6 \* W-WINX1 (W-LP-INDX) + .4)  
  IF H-NEW-WGNAT > H-IP-LIMIT  
    MOVE H-IP-LIMIT TO H-NEW-WGNAT  
  END-IF  
  COMPUTE H-NEW-COIN = H-LITEM-PYMT -  
    H-TOTAL-LN-DEDUCT - H-LITEM-REIM - H-LN-BLOOD-DEDUCT  
  MOVE 1 TO H-DCP-CODE  
  PERFORM 2455-SEARCH-KEY  
    THRU 2455-SEARCH-KEY-EXIT  
ELSE  
  IF OPSS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR 'K'  
    COMPUTE H-NEW-COIN = H-LITEM-PYMT -  
      H-TOTAL-LN-DEDUCT - H-LITEM-REIM  
      - H-LN-BLOOD-DEDUCT

```
MOVE 'Y' TO GJK-FLAG
MOVE 1 TO H-DCP-CODE
PERFORM 2455-SEARCH-KEY
    THRU 2455-SEARCH-KEY-EXIT
MOVE 2 TO H-DCP-CODE
ADD 1 TO W-DCP-MAX
SET W-DCP-INDX TO W-DCP-MAX
PERFORM 2475-STAGE-DCP-ENTRY
    THRU 2475-STAGE-DCP-ENTRY-EXIT
    UNTIL W-DCP-INDX = 1 OR
        H-DCP-STAGE NOT < W-DCP-STAGE (W-DCP-INDX - 1)
MOVE LN-SUB TO W-DCP-SUB (W-DCP-INDX)
MOVE H-DCP-STAGE TO W-DCP-STAGE (W-DCP-INDX)
MOVE ZERO TO W-DCP-COIN1 (W-DCP-INDX)
MOVE ZERO TO W-DCP-WGNAT (W-DCP-INDX)
MOVE H-NEW-COIN TO W-DCP-COIN2 (W-DCP-INDX)
MOVE OPSS-SRVC-IND (LN-SUB) TO
    W-DCP-SRVC-IND (W-DCP-INDX).
```

```
2450-ADJ-PROC-COIN-EXIT.
EXIT.
```

```
2455-SEARCH-KEY.
```

```
SET W-DCP-INDX TO 1.
SEARCH W-DCP-ENTRY VARYING W-DCP-INDX
    AT END
        PERFORM 2460-ADD-ENTRY
            THRU 2460-ADD-ENTRY-EXIT
        WHEN W-DCP-STAGE (W-DCP-INDX) = H-DCP-STAGE
            PERFORM 2465-UPDATE-ENTRY
                THRU 2465-UPDATE-ENTRY-EXIT.
```

```
2455-SEARCH-KEY-EXIT.
EXIT.
```

```
2460-ADD-ENTRY.
```

```
ADD 1 TO W-DCP-MAX.
SET W-DCP-INDX TO W-DCP-MAX.
PERFORM 2475-STAGE-DCP-ENTRY
    THRU 2475-STAGE-DCP-ENTRY-EXIT
    UNTIL W-DCP-INDX = 1 OR
        H-DCP-STAGE NOT < W-DCP-STAGE (W-DCP-INDX - 1).
IF OPSS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR 'K'
    MOVE ZERO TO W-DCP-SUB (W-DCP-INDX)
    MOVE H-DCP-STAGE TO W-DCP-STAGE (W-DCP-INDX)
    MOVE ZERO TO W-DCP-COIN1 (W-DCP-INDX)
    MOVE ZERO TO W-DCP-WGNAT (W-DCP-INDX)
    MOVE H-NEW-COIN TO W-DCP-COIN2 (W-DCP-INDX)
    MOVE 'X' TO W-DCP-SRVC-IND (W-DCP-INDX)
ELSE
    MOVE LN-SUB TO W-DCP-SUB (W-DCP-INDX)
    MOVE H-DCP-STAGE TO W-DCP-STAGE (W-DCP-INDX)
    MOVE H-NEW-COIN TO W-DCP-COIN1 (W-DCP-INDX)
    MOVE H-NEW-WGNAT TO W-DCP-WGNAT (W-DCP-INDX)
    MOVE ZERO TO W-DCP-COIN2 (W-DCP-INDX)
```

```

                MOVE OPPTS-SRVC-IND (LN-SUB) TO
                        W-DCP-SRVC-IND (W-DCP-INDX) .

2460-ADD-ENTRY-EXIT.
        EXIT.

2465-UPDATE-ENTRY.

        IF OPPTS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR 'K'
            ADD H-NEW-COIN TO W-DCP-COIN2 (W-DCP-INDX)
        ELSE
            IF W-DCP-SRVC-IND (W-DCP-INDX) = 'X'
                PERFORM 2485-REPLACE-TYPE1
                    THRU 2485-REPLACE-TYPE1-EXIT
            ELSE
                PERFORM 2480-RANK-COIN
                    THRU 2480-RANK-COIN-EXIT.

2465-UPDATE-ENTRY-EXIT.
        EXIT.

2475-STAGE-DCP-ENTRY.

        MOVE W-DCP-ENTRY (W-DCP-INDX - 1) TO
                W-DCP-ENTRY (W-DCP-INDX) .
        SET W-DCP-INDX DOWN BY 1.

2475-STAGE-DCP-ENTRY-EXIT.
        EXIT.

2480-RANK-COIN.

        IF H-NEW-WGNAT > W-DCP-WGNAT (W-DCP-INDX)
            MOVE LN-SUB TO W-DCP-SUB (W-DCP-INDX)
            MOVE H-NEW-COIN TO W-DCP-COIN1 (W-DCP-INDX)
            MOVE H-NEW-WGNAT TO W-DCP-WGNAT (W-DCP-INDX)
            MOVE OPPTS-SRVC-IND (LN-SUB)
                TO W-DCP-SRVC-IND (W-DCP-INDX) .

2480-RANK-COIN-EXIT.
        EXIT.

2485-REPLACE-TYPE1.

        MOVE LN-SUB TO W-DCP-SUB (W-DCP-INDX)
        MOVE H-NEW-COIN TO W-DCP-COIN1 (W-DCP-INDX)
        MOVE H-NEW-WGNAT TO W-DCP-WGNAT (W-DCP-INDX)
        MOVE OPPTS-SRVC-IND (LN-SUB)
            TO W-DCP-SRVC-IND (W-DCP-INDX) .

2485-REPLACE-TYPE1-EXIT.
        EXIT.

2500-ADJ-CHRG.

```

```

*****
**** NEW LOGIC INSERTED FOR WEB USE ONLY *

```

```

*****
IF (OPPS-SRVC-IND (LN-SUB) = 'T') OR
((OPPS-SRVC-IND (LN-SUB) = 'S') AND
(OPPS-HCPCS (LN-SUB) > '09999' AND
OPPS-HCPCS (LN-SUB) < '70000')) AND
(W-SUB-CHRG (W-LP-INDX) < 1.01)
MOVE 'Y' TO ST0-FLAG.

IF (OPPS-SRVC-IND (LN-SUB) = 'T') OR
((OPPS-SRVC-IND (LN-SUB) = 'S') AND
(OPPS-HCPCS (LN-SUB) > '09999' AND
OPPS-HCPCS (LN-SUB) < '70000')) AND
(OPPS-PKG-FLAG (LN-SUB) = '0' OR '3')
COMPUTE H-TOT-ST-CHRG = W-SUB-CHRG (W-LP-INDX)
+ H-TOT-ST-CHRG
COMPUTE H-TOT-ST-PYMT = H-LITEM-PYMT
+ H-TOT-ST-PYMT.

IF (OPPS-SRVC-IND (LN-SUB) = 'S' OR 'T' OR 'V'
OR 'X' OR 'P')
AND (OPPS-PKG-FLAG (LN-SUB) = '0' OR '3')
COMPUTE H-TOT-STVX-PYMT = H-LITEM-PYMT
+ H-TOT-STVX-PYMT.

2500-ADJ-CHRG-EXIT.
EXIT.
*****
* 1. APPLY DEDUCTIBLE TO HIGHEST NATIONAL COINSURANCE AMOUNT *
* - DESCENDING UNTIL DEDUCTIBLE = 0. *
* 2. CALCULATE THE STANDARD LINE PRICE *
* (APC PAYMENT * WAGE INDEX * SERVICE UNITS *
* * DISCOUNT FACTOR) *
* - WAGE ADJUST 60% OF THE APC PAYMENT ONLY *
* 3. ADD LINE PRICE TO OUTLIER HOLD AREA. *
* *
* 4. CALCULATE THE LINE PRICE FOR DESIGNATED DEVICES *
* AND DRUGS *
*****
2550-CALC-STANDARD.

COMPUTE H-MAX-COIN ROUNDED = (H-IP-LIMIT *
W-SRVC-UNITS (W-LP-INDX) * W-DISC-RATE (W-LP-INDX)).
IF OPPS-SRVC-IND (LN-SUB) = 'S' OR 'V'
OR 'T' OR 'P' OR 'X' THEN
COMPUTE H-LITEM-PYMT ROUNDED =
(((W-APC-PYMT (W-LP-INDX) * .60) *
W-WINX1 (W-LP-INDX))
+ (W-APC-PYMT (W-LP-INDX) * .40)) *
W-SRVC-UNITS (W-LP-INDX) * W-DISC-RATE (W-LP-INDX)
PERFORM 2560-CALC-BENE-DEDUCT
THRU 2560-CALC-BENE-DEDUCT-EXIT
ELSE
IF OPPS-SRVC-IND (LN-SUB) = 'H' THEN
IF OPPS-PYMT-IND (LN-SUB) = '6' THEN
PERFORM 2555-CALC-H-STANDARD
THRU 2555-CALC-H-STANDARD-EXIT
PERFORM 2560-CALC-BENE-DEDUCT

```

```

        THRU 2560-CALC-BENE-DEDUCT-EXIT
    ELSE
        MOVE 44 TO A-RETURN-CODE (LN-SUB)
        GO TO 2550-CALC-STANDARD-EXIT
    END-IF
ELSE
    IF OPPTS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR 'K' THEN
    IF OPPTS-PYMT-IND (LN-SUB) = '1' OR '5' OR '7' THEN
        MOVE 0 TO H-BLOOD-FRACTION
        PERFORM 2550-CALC-GJK
            THRU 2550-CALC-GJK-EXIT
        PERFORM 2560-CALC-BENE-DEDUCT
            THRU 2560-CALC-BENE-DEDUCT-EXIT
    ELSE
        MOVE 41 TO A-RETURN-CODE (LN-SUB)
        GO TO 2550-CALC-STANDARD-EXIT
    END-IF
END-IF.

IF H-LITEM-PYMT > 0
    IF L-SERVICE-FROM-DATE >= 20040101
        IF OPPTS-APC (LN-SUB) = ('1716' OR '1717' OR
            '1718' OR '1719' OR '1720' OR '2616' OR '2633')
            COMPUTE H-LITEM-REIM ROUNDED =
                ((H-LITEM-PYMT - H-TOTAL-LN-DEDUCT) -
                    H-LN-BLOOD-DEDUCT) * .8
            COMPUTE H-NAT-COIN = H-LITEM-PYMT -
                H-TOTAL-LN-DEDUCT - H-LITEM-REIM - H-LN-BLOOD-DEDUCT
            MOVE W-MIN-COIN (W-LP-INDX) TO H-MIN-COIN
        ELSE
            COMPUTE H-LITEM-REIM ROUNDED =
                ((H-LITEM-PYMT - H-TOTAL-LN-DEDUCT) -
                    H-LN-BLOOD-DEDUCT) * W-PPCT (W-LP-INDX)
            COMPUTE H-NAT-COIN = H-LITEM-PYMT -
                H-TOTAL-LN-DEDUCT - H-LITEM-REIM - H-LN-BLOOD-DEDUCT
            MOVE W-MIN-COIN (W-LP-INDX) TO H-MIN-COIN
        ELSE
            COMPUTE H-LITEM-REIM ROUNDED =
                ((H-LITEM-PYMT - H-TOTAL-LN-DEDUCT) -
                    H-LN-BLOOD-DEDUCT) * W-PPCT (W-LP-INDX)
            COMPUTE H-NAT-COIN = H-LITEM-PYMT -
                H-TOTAL-LN-DEDUCT - H-LITEM-REIM - H-LN-BLOOD-DEDUCT
            MOVE W-MIN-COIN (W-LP-INDX) TO H-MIN-COIN
        ELSE
            COMPUTE H-LITEM-REIM ROUNDED =
                ((H-LITEM-PYMT - H-TOTAL-LN-DEDUCT) -
                    H-LN-BLOOD-DEDUCT) * W-PPCT (W-LP-INDX)
            COMPUTE H-NAT-COIN = H-LITEM-PYMT -
                H-TOTAL-LN-DEDUCT - H-LITEM-REIM - H-LN-BLOOD-DEDUCT
            MOVE W-MIN-COIN (W-LP-INDX) TO H-MIN-COIN
        NEXT SENTENCE.

    IF H-MIN-COIN > 0
        IF OPPTS-SRVC-IND (LN-SUB) = 'G' OR 'H' OR 'J' OR 'K'
            COMPUTE H-MIN-COIN ROUNDED = H-MIN-COIN *
                (W-SRVC-UNITS (W-LP-INDX) -
                    (W-SRVC-UNITS (W-LP-INDX) * H-BLOOD-FRACTION))
                * W-DISC-RATE (W-LP-INDX)
        ELSE
            IF OPPTS-APC (LN-SUB) = '0158' OR '0159'
                COMPUTE H-MIN-COIN ROUNDED = H-LITEM-PYMT * .25
            ELSE
                COMPUTE H-MIN-COIN ROUNDED = H-LITEM-PYMT * .2

```



END-IF  
END-IF  
END-IF.

MOVE W-RED-COIN (W-LP-INDX) TO H-RED-COIN.  
IF (H-RED-COIN > 0 AND < H-MIN-COIN)  
MOVE H-MIN-COIN TO H-RED-COIN  
ELSE  
IF H-RED-COIN < H-NAT-COIN AND > H-MIN-COIN  
AND > H-MAX-COIN  
COMPUTE H-LITEM-REIM = H-LITEM-REIM +  
(H-RED-COIN - H-MAX-COIN)  
END-IF  
END-IF.

IF H-NAT-COIN > H-MAX-COIN AND H-RED-COIN = 0 THEN  
COMPUTE H-LITEM-REIM = H-LITEM-REIM +  
(H-NAT-COIN - H-MAX-COIN)  
MOVE H-MAX-COIN TO H-NAT-COIN.

2550-CALC-STANDARD-EXIT.  
EXIT.

\*\*\*\*\*  
\* 1. CALCULATE LINE ITEM PAYMENT FOR SERVICE INDICATOR \*  
\* TYPES G , J , OR K. \*  
\* 2. EFFECTIVE 01/01/2003 REMOVE PRO RATA REDUCTION FOR \*  
\* ALL SERVICE INDICATOR 'G' PAYMENTS \*  
\*\*\*\*\*

2550-CALC-GJK.

IF OPPS-HCPCS(LN-SUB) = 'P9021' OR 'P9010' OR 'P9038'  
OR 'P9016' OR 'C1010' OR 'C1018' OR 'P9022'  
OR 'P9039' OR 'P9040' OR 'C1016' OR 'C1021'  
OR 'C1020'  
PERFORM 2550-SET-BLOOD-FRACTION  
THRU 2550-SET-BLOOD-FRACTION-EXIT  
ELSE  
COMPUTE H-LITEM-PYMT ROUNDED =  
W-APC-PYMT (W-LP-INDX) \* W-SRVC-UNITS (W-LP-INDX)  
\* W-DISC-RATE (W-LP-INDX)  
GO TO 2550-CALC-GJK-EXIT.

COMPUTE H-LITEM-PYMT ROUNDED =  
W-BD-APC-PYMT (W-BD-INDX) \* W-BD-SRVC-UNITS (W-BD-INDX)  
\* W-BD-DISC-RATE (W-BD-INDX).

COMPUTE H-LN-BLOOD-DEDUCT ROUNDED =  
H-LITEM-PYMT \* H-BLOOD-FRACTION.

SET W-BD-INDX UP BY 1.

2550-CALC-GJK-EXIT.  
EXIT.

2550-SET-BLOOD-FRACTION.

```
MOVE W-BD-SUB (W-BD-INDX) TO LN-SUB.
IF H-BENE-PINTS-USED > 0
  IF W-BD-SRVC-UNITS (W-BD-INDX) <= H-BENE-PINTS-USED
    MOVE 1 TO H-BLOOD-FRACTION
    COMPUTE H-BENE-PINTS-USED =
      H-BENE-PINTS-USED - W-BD-SRVC-UNITS (W-BD-INDX)
  ELSE
    IF W-BD-SRVC-UNITS (W-BD-INDX) > H-BENE-PINTS-USED
      COMPUTE H-BLOOD-FRACTION =
        H-BENE-PINTS-USED / W-BD-SRVC-UNITS (W-BD-INDX)
      MOVE 0 TO H-BENE-PINTS-USED
    ELSE
      MOVE 0 TO H-BLOOD-FRACTION
  ELSE
    MOVE 0 TO H-BLOOD-FRACTION.
```

2550-SET-BLOOD-FRACTION-EXIT.
EXIT.

```
*****
* 1. CALCULATE LINE ITEM PAYMENT FOR SERVICE INDICATOR *
* TYPE H. *
* 2. EFFECTIVE 04/01/2002 A PRO RATA REDUCTION APPLIES TO *
* ALL SERVICE INDICATOR 'H' PAYMENTS (CURRENTLY .689) *
*****
```

2555-CALC-H-TOT.

```
MOVE W-LP-SUB (W-LP-INDX) TO LN-SUB.
MOVE OPSS-SUB-CHRG (LN-SUB) TO H-SUB-CHRG.
IF OPSS-SRVC-IND (LN-SUB) = 'H'
  IF OPSS-PYMT-IND (LN-SUB) = '6'
    COMPUTE H-TOT-H-CHRG =
      (H-TOT-H-CHRG + H-SUB-CHRG)
  ELSE
    NEXT SENTENCE
ELSE
  NEXT SENTENCE.
```

2555-CALC-H-TOT-EXIT.
EXIT.

2555-CALC-H-STANDARD.

```
MOVE OPSS-SUB-CHRG (LN-SUB) TO H-SUB-CHRG.

COMPUTE T-LITEM-PYMT ROUNDED =
  (H-SUB-CHRG * L-PSF-OPCOST-RATIO).

IF (C-FLAG = 'Y')
  IF (H-TOT-OFF-UNITS > H-TOT-HTD-UNITS)
    COMPUTE H-TOTAL-WAOFF ROUNDED =
      ((H-TOTAL-OFFSET * .60) * A-WINX)
      + (H-TOTAL-OFFSET * .40)
      * (H-TOT-HTD-UNITS / H-TOT-OFF-UNITS)
```

```

        PERFORM 2700-CALC-H-OFFSET
        THRU 2700-CALC-H-OFFSET-EXIT
    ELSE
        COMPUTE H-TOTAL-WAOFF ROUNDED =
            ((H-TOTAL-OFFSET * .60) * A-WINX)
            + (H-TOTAL-OFFSET * .40)
        PERFORM 2700-CALC-H-OFFSET
        THRU 2700-CALC-H-OFFSET-EXIT
    ELSE
        NEXT SENTENCE.

    IF T-LITEM-PYMT < 0 THEN
        MOVE 0 TO H-LITEM-PYMT
    ELSE
        MOVE T-LITEM-PYMT TO H-LITEM-PYMT.

2555-CALC-H-STANDARD-EXIT.
EXIT.

2560-CALC-BENE-DEDUCT.

    IF OPPTS-PYMT-ADJ-FLAG (LN-SUB) = '4'
        GO TO 2560-CALC-BENE-DEDUCT-EXIT.
    IF H-BENE-DEDUCT > 0 THEN
        COMPUTE H-LN-BLD-PYMT =
            H-LITEM-PYMT - H-LN-BLOOD-DEDUCT
        IF H-BENE-DEDUCT <= H-LN-BLD-PYMT THEN
            MOVE H-BENE-DEDUCT TO H-TOTAL-LN-DEDUCT
            MOVE 0 TO H-BENE-DEDUCT
        ELSE
            COMPUTE H-BENE-DEDUCT =
                H-BENE-DEDUCT - H-LN-BLD-PYMT
            MOVE H-LN-BLD-PYMT TO H-TOTAL-LN-DEDUCT
            MOVE 20 TO A-RETURN-CODE (LN-SUB)
        END-IF
    END-IF.

2560-CALC-BENE-DEDUCT-EXIT.
EXIT.

*****
** - NEW FOR JANUARY 2004 **
** - CHECK >= 20040101 AND SRVC-IND = 'K' **
** - DISCONTINUE OUTLIER PROCESS **
*****
2600-ADJ-CHRG-OUTL.

    MOVE W-LP-SUB (W-LP-INDX) TO LN-SUB.
    * IF ((L-SERVICE-FROM-DATE > 20031231) AND
    * (L-SERVICE-FROM-DATE < 20040401)) AND
    * (OPPTS-SRVC-IND (LN-SUB) = 'K')
    * GO TO 2600-ADJ-CHRG-OUTL-EXIT.
*****
** - NEW FOR APRIL 2004 **
** - CHECK >= 20040101 AND SRVC-IND = 'K' **
** - CONTINUE OUTLIER PROCESS FOR SPECIFIED APCS **

```

\*\*\*\*\*

```
IF L-SERVICE-FROM-DATE >= 20040101
  IF OPSS-SRVC-IND (LN-SUB) = 'K'
    IF (OPPS-APC (LN-SUB) = '0702' OR '0704' OR
      '0705' OR '0737' OR '1045' OR '1064' OR '1065' OR
      '1079' OR '1080' OR '1081' OR '1089' OR '1091' OR
      '1092' OR '1095' OR '1096' OR '1122' OR '1200' OR
      '1201' OR '1600' OR '1603' OR '1604' OR '1619' OR
      '1620' OR '1622' OR '1624' OR '1625' OR '1628' OR
      '1775' OR '9013' OR '9025' OR '9100' OR '9117' OR
      '9118' OR '9400' OR '9402' OR '9403' OR '9404' OR
      '9405' OR '9408' OR '9434' OR '0701')
      NEXT SENTENCE
    ELSE
      GO TO 2600-ADJ-CHRG-OUTL-EXIT
    ELSE
      NEXT SENTENCE
  ELSE
    NEXT SENTENCE.

IF OPSS-SRVC-IND (LN-SUB) = 'G' OR 'J' OR
  'H' OR 'N'
  GO TO 2600-ADJ-CHRG-OUTL-EXIT.
```

\*\*\*\*\*  
\*\*\*\* NEW LOGIC INSERTED FOR WEB USE ONLY \*  
\*\*\*\*\*

```
IF (ST0-FLAG = 'Y') AND (OPPS-SRVC-IND (LN-SUB)
  = 'T' ) OR ((OPPS-SRVC-IND (LN-SUB) = 'S') AND
  (OPPS-HCPCS (LN-SUB) > '09999' AND
  OPPS-HCPCS (LN-SUB) < '70000'))
  AND (H-TOT-ST-PYMT > 0)
  COMPUTE H-CHRG-RATE ROUNDED =
    (A-LITEM-PYMT (LN-SUB) / H-TOT-ST-PYMT)
  COMPUTE W-SUB-CHRG (W-LP-INDX) ROUNDED =
    (H-CHRG-RATE * H-TOT-ST-CHRG)
ELSE
  IF (N-FLAG = 'Y' AND ST0-FLAG = 'N') AND
    ((OPPS-SRVC-IND (LN-SUB) = 'S' OR 'T' OR 'V' OR
    'X' OR 'P')
    AND (OPPS-PKG-FLAG (LN-SUB) = '0' OR '3'))
    AND (H-TOT-STVX-PYMT > 0)
    COMPUTE H-CHRG-RATE ROUNDED =
      (A-LITEM-PYMT (LN-SUB) / H-TOT-STVX-PYMT)
    COMPUTE H-SUB-CHRG ROUNDED =
      (H-CHRG-RATE * H-TOT-N-CHRG)
    COMPUTE W-SUB-CHRG (W-LP-INDX) ROUNDED =
      W-SUB-CHRG (W-LP-INDX) + H-SUB-CHRG.

IF (N-FLAG = 'Y' AND ST0-FLAG = 'Y') AND
  ((OPPS-SRVC-IND (LN-SUB) = 'S' OR 'T' OR 'V' OR
  'X' OR 'P')
  AND (OPPS-PKG-FLAG (LN-SUB) = '0' OR '3'))
  AND (H-TOT-STVX-PYMT > 0)
  COMPUTE H-CHRG-RATE ROUNDED =
    (A-LITEM-PYMT (LN-SUB) / H-TOT-STVX-PYMT)
  COMPUTE H-SUB-CHRG ROUNDED =
```

```

          (H-CHRG-RATE * H-TOT-N-CHRG)
    COMPUTE W-SUB-CHRG (W-LP-INDX) ROUNDED =
          W-SUB-CHRG (W-LP-INDX) + H-SUB-CHRG.
*****
** - NEW FOR JANUARY 2004
** - CHECK >= 20040101 AND PROVIDER RANGE FOR CMHC
** - COMPUTE H-LITEM-OUTL-PYMT USING NEW FORMULA
*****

    IF L-SERVICE-FROM-DATE >= 20040101
      MOVE 2.6 TO H-OUTLIER-FACTOR
      MOVE .50 TO H-OUTLIER-PCT
      IF (L-PSF-PROV-3456 >= '1400' AND
          L-PSF-PROV-3456 <= '1499') OR
          (L-PSF-PROV-3456 >= '4600' AND
          L-PSF-PROV-3456 <= '4799') OR
          (L-PSF-PROV-3456 >= '4900' AND
          L-PSF-PROV-3456 <= '4999')
        MOVE 3.65 TO H-OUTLIER-FACTOR
      ELSE
        NEXT SENTENCE
    ELSE
      MOVE 2.75 TO H-OUTLIER-FACTOR
      MOVE .45 TO H-OUTLIER-PCT.

    COMPUTE H-LITEM-OUTL-PYMT ROUNDED =
      ((W-SUB-CHRG (W-LP-INDX) * L-PSF-OPCOST-RATIO) -
      (H-OUTLIER-FACTOR * A-LITEM-PYMT (LN-SUB))) *
      H-OUTLIER-PCT.

    IF H-LITEM-OUTL-PYMT > 0
      COMPUTE H-OUTLIER-PYMT = H-OUTLIER-PYMT +
        H-LITEM-OUTL-PYMT.

    IF OPDS-LITEM-ACT-FLAG (LN-SUB) = '4'
      COMPUTE H-OUTLIER-PYMT = H-OUTLIER-PYMT -
        H-LITEM-OUTL-PYMT
      MOVE 0 TO H-LITEM-OUTL-PYMT.

    2600-ADJ-CHRG-OUTL-EXIT.
    EXIT.

*****
* 1. RE-CALCULATE LINE ITEM PAYMENT FOR SERVICE INDICATOR
*   TYPE H.
* 2. SUBTRACT WAGE ADJUSTED OFFSET AMOUNT FROM SI TYPE 'H'
*   WITH HCPCS CODE BEGINNING WITH 'C'
* 2. EFFECTIVE 04/01/2002
*****

    2700-CALC-H-OFFSET.

    IF H-TOT-H-CHRG > 0
      COMPUTE H-OFF-RATE ROUNDED =
        H-SUB-CHRG / H-TOT-H-CHRG
      COMPUTE T-LITEM-PYMT ROUNDED = T-LITEM-PYMT -
        (H-TOTAL-WAOFF * H-OFF-RATE)

```

```

ELSE
    NEXT SENTENCE.

2700-CALC-H-OFFSET-EXIT.
EXIT.

2800-ADJ-STV-REIM.

IF W-DCP-CODE (W-DCP-INDX) = 1
    PERFORM 2810-PROCESS-TYPE1
        THRU 2810-PROCESS-TYPE1-EXIT
ELSE
    PERFORM 2840-PROCESS-TYPE2
        THRU 2840-PROCESS-TYPE2-EXIT.

2800-ADJ-STV-REIM-EXIT.
EXIT.

2810-PROCESS-TYPE1.

IF W-DCP-COIN2 (W-DCP-INDX) > 0
    MOVE W-DCP-DOS (W-DCP-INDX) TO H-DCP-DOS
    MOVE W-DCP-WGNAT (W-DCP-INDX) TO H-TOTAL
    COMPUTE H-RATIO =
        (H-IP-LIMIT - W-DCP-WGNAT (W-DCP-INDX)) /
        W-DCP-COIN2 (W-DCP-INDX)
    IF H-RATIO < 0
        MOVE 0 TO H-RATIO.
    IF H-RATIO > 1
        MOVE 1 TO H-RATIO.

2810-PROCESS-TYPE1-EXIT.
EXIT.

2840-PROCESS-TYPE2.

IF W-DCP-DOS (W-DCP-INDX) >= 20010701
    IF W-DCP-DOS (W-DCP-INDX) = H-DCP-DOS
        MOVE W-DCP-SUB (W-DCP-INDX) TO LN-SUB
        COMPUTE H-SHIFT =
            W-DCP-COIN2 (W-DCP-INDX) * (1 - H-RATIO)
        COMPUTE H-TOTAL =
            A-ADJ-COIN (LN-SUB) + H-TOTAL - H-SHIFT
        IF H-TOTAL > H-IP-LIMIT
            COMPUTE H-SHIFT = H-SHIFT + H-TOTAL
                - H-IP-LIMIT
        END-IF
        COMPUTE A-ADJ-COIN (LN-SUB) =
            A-ADJ-COIN (LN-SUB) - H-SHIFT
        COMPUTE A-LITEM-REIM (LN-SUB) =
            A-LITEM-REIM (LN-SUB) + H-SHIFT
        MOVE 22 TO A-RETURN-CODE (LN-SUB)
    END-IF
ELSE
    IF W-DCP-DOS (W-DCP-INDX) >= 20010101
        IF W-DCP-COIN2 (W-DCP-INDX) > H-IP-LIMIT
            MOVE W-DCP-SUB (W-DCP-INDX) TO LN-SUB

```

```
      COMPUTE H-SHIFT =
          W-DCP-COIN2 (W-DCP-INDX) - H-IP-LIMIT
      COMPUTE A-ADJ-COIN (LN-SUB) =
          A-ADJ-COIN (LN-SUB) - H-SHIFT
      COMPUTE A-LITEM-REIM (LN-SUB) =
          A-LITEM-REIM (LN-SUB) + H-SHIFT
      MOVE 22 TO A-RETURN-CODE (LN-SUB)
      END-IF
      END-IF
      END-IF.
```

```
2840-PROCESS-TYPE2-EXIT.
      EXIT.
```

```
*****
* 1. MOVE TOTAL CLAIM CHARGE AMOUNT. *
* 2. MOVE TOTAL CLAIM PAYMENT AMOUNT. *
* 3. MOVE TOTAL CLAIM BLOOD PINTS USED. *
* 4. CALCULATE CLAIM LEVEL OUTLIER AMOUNT. *
*****
```

```
2900-END-PRICE-RTN.
```

```
      MOVE H-TOT-CHRG TO A-TOT-CLM-CHRG.
      MOVE H-TOT-PYMT TO A-TOT-CLM-PYMT.
      COMPUTE A-BLOOD-PINTS-USED =
          H-BENE-BLOOD-PINTS - H-BENE-PINTS-USED.
      IF H-OUTLIER-PYMT > 0
          MOVE H-OUTLIER-PYMT TO A-OUTLIER-PYMT.
```

```
2900-END-PRICE-RTN-EXIT.
      EXIT.
```