

/*APPENDIX 40*/

/*DERIVATIONS FOR 1997 *KEY* AND OTHER CREATED VARIABLES*/

/*June, 1999*/

/*INTRODUCTION. Below are the derivations for 1997 key and other created variables. Some variables were coded in SPSS, others in SAS. The programs have been put together here, paginated, and formatted to fit the page. Included are:

RNI/ISR ... p. 2

WORK, WUMP, WOLF... p. 6

1980 Industry/Occupation codes... p. 41

Net family assets, Net family income, Summation... p. 42

Highest grade completed... p. 45

Notes on pension and employer gaps rosters... p. 47

Hourly Rate of Pay... p. 50

Text entries for Hea9d, Par56, Par66... p. 51 */

```

*****
*****  SPSS code for 1997 RNI, ISR key variables, YW  *****
*****

COMPUTE  SERIAL =  R0000100          /* create SERIAL b/c it's easier to remember

*****  Recode OUTCOME (R3498300) so the categories are consistent with the
*****  the categories in RNI-CROSS-SECTION (R3498000)

RECODE  R3498300  (201, 203, 204, 205=0)      (213=11)   (214=5)   (216=1)
                  (217=6)   (218=9)   (219=11)   (223=7)   (234=8)   (250=10)
                  (251=13)   (260=4)   (261=2)   (270=2)   (271=2)
                  (999=99)      INTO    RNI97_XS

VAR LAB  RNI97_XS      'RNI97-CROSS-SECTION'
VAL LAB  RNI97_XS
0  '97 IW obtained'
1  'Unable to contact-Reason unspecified'
2  'Mover-good address, interview impossible'
3  'Mover-good address, unable to IW after repeated attempts'
4  'Mover-no good address given'
5  'Nonmover-unable to interview after repeated attempts'
6  'Temporarily absent'
7  'Armed Forces'
8  'Institutionalized'
9  'Refused'
10 'Deceased'
11 'Other'
12 'No interview for 2 yrs, dropped from sample'
13 'Moved outside US, not in Armed Forces'
14 'Congressional refusal'
99 'Out of scope in 97 (no contact attempt)'

*****
*****  CREATING THE CUMULATIVE RNI VARIABLE FOR 1997  *****

** start by making the CUMULATIVE RNI = CROSS SECTION for Rs in 1997 sample
COMPUTE  RNI97_CU = RNI97_XS

** Select the 2475 out-of-sample MW, assign them prior year's RNI code
DO IF  (RNI97_CU = 99)          /* selects 2475 out-of-scope MW respondents
.COMPUTE  RNI97_CU  = R1601200  /* copy 95 Cumulative RNI into 97 cumulative RNI
END IF

VAR LAB  RNI97_CU      '1997 Cumulative RNI'
VAL LAB  RNI97_CU
0  '97 interview obtained'
1  'Unable to contact - Reason unspecified'
2  'Mover-good address, interview impossible'
3  'mover-good address, unable to interview'
4  'mover-no good address given'
5  'Nonmover-unable to interview after repeated attempts'
6  'Temporarily absent'
7  'Armed Forces'
8  'Institutionalized'
9  'Refused'
10 'Deceased'
11 'Other'
12 'No interview for 2 yrs, dropped from sample'
13 'Moved outside US, not in Armed Forces'
14 'Congressional refusal'

```

```

##### 4 cases are Congressional Refusals
**** These 4 respondents notified Census after the 1995 Survey that they
**** were never to be re-interviewed. Thus, they are considered to be
**** Congressional Refusals (RNI = 14).

```

```

DO IF ANY(SERIAL, 1799, 3557, 3797)
.COMPUTE RNI97_CU = 14
END IF

```

```

DO IF (RNI97_XS = 99) /* selects out-of-scope Respondents
COMPUTE ISR97 = R1601300 /*copy previous ISR into ISR for 1997
ELSE
RECODE RNI97_CU (0 = 1) (10 = 2) (9 = 3) (12 = 4) (7 = 5)
(1,2,3,4,5,6,8,11,13 = 6) (14 = 8) (ELSE = -9) INTO ISR97
END IF

```

```

** 3 cases need values of 8 ("Congressional Refusal")
IF ANY(SERIAL, 1799, 3557, 3797) ISR97 = 8

```

```

** 1 case has to have the 1997 ISR changed from "Dropped.." to "deceased"
IF (SERIAL = 2936) ISR97 = 2

```

```

VAL LAB ISR97
1 '1997 IW obtained'
2 'Non-IW: Deceased'
3 'Non-IW: Refused'
4 'Non-IW: Dropped after 2 consecutive years as Non_interview'
5 'Non-IW: Armed Forces'
6 'Non-IW: Other'
7 'Code not used in 97'
8 'Non-IW: Congressional refusal'

```

```

** Convert SPSS namesd to Ref numbers
COMPUTE R3498000 = RNI97_XS
COMPUTE R3498100 = RNI97_CU
COMPUTE R3498200 = ISR_97

```

```

FINISH

```

```

*****
***** SPSS code for 1997 RNI, ISR key variables, YW *****
*****
=====
COMPUTE SERIAL = R0000100          /* create SERIAL b/c it's easier to remember

***** Recode OUTCOME (R3498300) so the categories are consistent with the
***** the categories in RNI-CROSS-SECTION (R3498000)

RECODE R3498300 (201, 203, 204, 205=0)      (213=11)      (214=5)      (216=1)
                (217=6)      (218=9)      (219=11)      (223=7)      (234=8)      (250=10)
                (251=13)      (260=4)      (261=2)      (270=2)      (271=2)
                (999=99)      INTO      RNI97_XS

VAR LAB RNI97_XS 'RNI97-CROSS-SECTION'
VAL LAB RNI97_XS
0 '97 IW obtained'
1 'Unable to contact-Reason unspecified'
2 'Mover-good address, interview impossible'
3 'Mover-good address, unable to IW after repeated attempts'
4 'Mover-no good address given'
5 'Nonmover-unable to interview after repeated attempts'
6 'Temporarily absent'
7 'Armed Forces'
8 'Institutionalized'
9 'Refused'
10 'Deceased'
11 'Other'
12 'No interview for 2 yrs, dropped from sample'
13 'Moved outside US, not in Armed Forces'
14 'Congressional refusal'
99 'Out of scope in 97 (no contact attempt)'

*****
***** CREATING THE CUMULATIVE RNI VARIABLE FOR 1997 *****

** start by making the CUMULATIVE RNI = CROSS SECTION for Rs in 1997 sample
COMPUTE RNI97_CU = RNI97_XS

** Select the 2475 out-of-sample MW, assign them prior year's RNI code
DO IF (RNI97_CU = 99)          /* selects out-of-scope MW respondents
.COMPUTE RNI97_CU = R1601200    /* copy 95 Cumulative RNI into 97 cumulative RNI
END IF

VAR LAB RNI97_CU '1997 Cumulative RNI'
VAL LAB RNI97_CU
0 '97 interview obtained'
1 'Unable to contact - Reason unspecified'
2 'Mover-good address, interview impossible'
3 'mover-good address, unable to interview'
4 'mover-no good address given'
5 'Nonmover-unable to interview after repeated attempts'
6 'Temporarily absent'
7 'Armed Forces'
8 'Institutionalized'
9 'Refused'
10 'Deceased'
11 'Other'
12 'No interview for 2 yrs, dropped from sample'
13 'Moved outside US, not in Armed Forces'
14 'Congressional refusal'

```

```

##### 4 cases are Congressional Refusals
** This YW notified Census after the 1995 Survey that she never be
** contacted again. Thus, she is coded as a Congressional Refusals (RNI = 14).

IF (SERIAL = 0238)          RNI97_CU = 14

```

```

DO IF (RNI97_XS = 99)      /* selects out-of-scope Respondents
COMPUTE   ISR97 = R1601300 /*copy previous ISR into ISR for 1997
ELSE
RECODE   RNI97_CU   (0 = 1)   (10 = 2)   (9 = 3)   (12 = 4)   (7 = 5)
              (1,2,3,4,5,6,8,11,13 = 6)   (14 = 8)   (ELSE = -9) INTO   ISR97
END IF

```

```

** 1 case needs to be assigned to "Congressional Refusal"
IF (SERIAL = 0238)   ISR97 = 8

```

```

VAL LAB   ISR97
  1  '1997 IW obtained'
  2  'Non-IW: Deceased'
  3  'Non-IW: Refused'
  4  'Non-IW: Dropped after 2 consecutive years as Non_interview'
  5  'Non-IW: Armed Forces'
  6  'Non-IW: Other'
  7  'Code not used in 97'
  8  'Non-IW: Congressional refusal'

```

```

** Convert SPSS names to Reference numbers

```

```

COMPUTE R3498000 = RNI97_XS
COMPUTE R3498100 = RNI97_CU
COMPUTE R3498200 = ISR_97

```

```

FINISH

```

```

*=====
*   SPSS code used to create key variables WORK, WUMP, WOLF for MW cohort
*=====

** This initializes variables that will be used at various points in the program.

NUMERIC   EMPST08D           /* EMP-START-ARR-08D
NUMERIC   EMPST09D           /* EMP-START-ARR-09D
NUMERIC   GAP_1.8 / GAP_1.9
NUMERIC   BLANK
NUMERIC   GAP_2.3 / GAP_2.5 / GAP_2.6 / GAP_2.7 / GAP_2.8 / GAP_2.9
NUMERIC   INTCK.9           /* square roster for INTER-CK-ARR

COMPUTE    SERIAL = R3495000

** This selects the Rs who were interviewed in 1997
SELECT IF  (R3498800 > -3)

*=====
**** The individual D-M-Y variables are combined to form a single variable
**** measuring the date. This is done using one of the the SPSS date aggregation
**** functions.
**** It makes adding, subtracting, and otherwise comparing dates much easier.

***      DOLI
COMPUTE   DOLI = DATE.DMY(R3498900, R3499000, R3499100)
PRINT FORMAT DOLI (DATE9)

***      DOI
COMPUTE   DOI = DATE.DMY(R3498600, R3498700, R3498800)
PRINT FORMAT DOI (DATE9)

*=====
**      Calculate Weeks Between Interview Dates   WBID.W
*=====
COMPUTE   WBID.RAW = ((CTIME.DAYS(DOI - DOLI)) / 7)
VAR LAB   WBID.RAW   '# weeks btwn intvs, not rounded off yet (DOI - DOLI)'

COMPUTE   WBID = RND((CTIME.DAYS(DOI - DOLI)) / 7)
VAR LAB   WBID   'WBID, rounded off'

*=====
*****      Employer start and End dates      *****
*****
*****      NO REFERENCE DATE CREATED IF ENAME HAD NO ACTIVITY SINCE DOLI
*****
*****      NO MISSING DATES WERE IMPUTED (if day is missing the entire
*****      date is missing)

DO REPEAT
  REFST_D = R4236000  R4236600  R4237200  R4237800  R4238400  R4239000
             R4239600  EMPST08D  EMPST09D
/  REFST_M = R4236100  R4236700  R4237300  R4237900  R4238500  R4239100
             R4239700  R4240200  R4240700
/  REFST_Y = R4236200  R4236800  R4237400  R4238000  R4238600  R4239200
             R4239800  R4240300  R4240800
/  REF_ST = REFST1 REFST2 REFST3 REFST4 REFST5 REFST6 REFST7
             REFST8 REFST9
/  REFEN_D = R4236300  R4236900  R4237500  R4238100  R4238700  R4239300
             R4239900  R4240400  R4240900
/  REFEN_M = R4236400  R4237000  R4237600  R4238200  R4238800  R4239400
             R4240000  R4240500  R4241000

```

```

/  REFEN_Y = R4236500  R4237100  R4237700  R4238300  R4238900  R4239500
              R4240100  R4240600  R4241100
/  REF_END = REFEND1  REFEND2  REFEND3  REFEND4  REFEND5  REFEND6  REFEND7
              REFEND8  REFEND9
/  CK_RSP_I = R3638000  R3658700  R3678200  R3698200  R3717500  R3736100
              R3752600  R3768800  R3784800
/  ST_DAY = ST1_D  ST2_D  ST3_D  ST4_D  ST5_D  ST6_D  ST7_D
              ST8_D  ST9_D
/  ST_MO = ST1_M  ST2_M  ST3_M  ST4_M  ST5_M  ST6_M  ST7_M
              ST8_M  ST9_M
/  ST_YEAR = ST1_Y  ST2_Y  ST3_Y  ST4_Y  ST5_Y  ST6_Y  ST7_Y
              ST8_Y  ST9_Y
/  END_DAY = END1_D  END2_D  END3_D  END4_D  END5_D  END6_D  END7_D
              END8_D  END9_D
/  END_MO = END1_M  END2_M  END3_M  END4_M  END5_M  END6_M  END7_M
              END8_M  END9_M
/  END_YEAR = END1_Y  END2_Y  END3_Y  END4_Y  END5_Y  END6_Y  END7_Y
              END8_Y  END9_Y
/  GAP_1 = R3887200  R3887800  R3888400  R3889000  R3889600  R3890200
              R3890800  GAP_1.8  GAP_1.9

** This creates copies of the original D-M-Y variables that will be
** used to create the start and stop dates.
COMPUTE ST_DAY = REFST_D
COMPUTE ST_MO = REFST_M
COMPUTE ST_YEAR = REFST_Y
COMPUTE END_DAY = REFEN_D
COMPUTE END_MO = REFEN_M
COMPUTE END_YEAR = REFEN_Y

* Create dates only if Ename had activity since DOLI, or if an END-GAP exists
DO IF NVAL(CK_RSP_I) ge 1 or NVAL(GAP_1) ge 1

.COMPUTE REF_ST = DATE.DMY(Refst_D, Refst_M, Refst_Y)
.COMPUTE REF_END = DATE.DMY(RefEN_D, RefEN_M, RefEN_Y)

.** Hand edit the End date for these respondents who had
.** the wrong PC system date and thus the wrong REFEND1 date
.DO IF ANY(SERIAL, 0058, 0778)
.COMPUTE REFEND1 = DOI
.END IF
END IF

PRINT FORMAT REF_ST REF_END (DATE9)
END REPEAT

VAR LAB REFEND1 'End date for Ename 1'
/ REFEND2 'End date for Ename 2'
/ REFEND3 'End date for Ename 3'
/ REFEND4 'End date for Ename 4'
/ REFEND5 'End date for Ename 5'
/ REFEND6 'End date for Ename 6'
/ REFEND7 'End date for Ename 7'
/ REFEND9 'End date for Ename 9'
/ REFST1 'Start date for Ename 1'
/ REFST2 'Start date for Ename 2'
/ REFST3 'Start date for Ename 3'
/ REFST4 'Start date for Ename 4'
/ REFST5 'Start date for Ename 5'
/ REFST6 'Start date for Ename 6'
/ REFST7 'Start date for Ename 7'
/ REFST9 'Start date for Ename 9'

```

```

*=====
*//////////
*//////////
*//////////
*//////////
*****
**
**      GAP_CENS:  This universe variable places respondents into 1 of 5
**                  groups.
**                  0 = No gaps at all
**                  1 = DOLI-gap only, No work at all
**                  2 = DOLI-gap only, worked since DOLI
**                  3 = End-gap only
**                  4 = Both kinds of gaps
*****

****  Calculate DBID: # DAYS Between Interview Dates
COMPUTE  DBID  =  CTIME.DAYS(DOI - DOLI)
VAR LAB   DBID   '# DAYS btwn intvs, not rounded off yet (DOI - DOLI)'

*****  number of gap questions hit
COUNT  NUM_GAP1 = R3887200, R3887800, R3888400, R3889000, R3889600, R3890200,
                R3890800, GAP_1.8, GAP_1.9 (-2 thru HI)
RECODE  NUM_GAP1  (1 thru HI = 1)  (0=0)  INTO  HIT_1

COMPUTE  HIT_1B = 0
IF (R3893000 ge -2)      HIT_1B = 1          /* DOLI-GAP (Gap-1B)

***  This is needed because a few cases have an End-gap and a pair
***  dates, but CK-RSP-I is blank.

NUMERIC  EMP01 to EMP09

IF  NVAL(R3638000, R3887200) > 0  EMP01 = 1
IF  NVAL(R3658700, R3887800) > 0  EMP02 = 1
IF  NVAL(R3678200, R3888400) > 0  EMP03 = 1
IF  NVAL(R3698200, R3889000) > 0  EMP04 = 1
IF  NVAL(R3717500, R3889600) > 0  EMP05 = 1
IF  NVAL(R3736100, R3890200) > 0  EMP06 = 1
IF  NVAL(R3752600, R3890800) > 0  EMP07 = 1
IF  NVAL(R3768800, GAP_1.8) > 0  EMP08 = 1
IF  NVAL(R3784800, GAP_1.9) > 0  EMP09 = 1

COUNT  NUM_ACT = EMP01, EMP02, EMP03, EMP04, EMP05, EMP06, EMP07, EMP08, EMP09
                (1)
RECODE  NUM_ACT  (0=0)  (1 THRU HI=1)  INTO  ACTIV
VAR LAB  NUM_ACT  '# Enames with activity, or and END-GAP'
        /  EMP01  'Ename 01 has activty, or END-GAP, or both'
        /  EMP02  'Ename 02 has activty, or END-GAP, or both'
        /  ACTIV  'Activty since DOLI, or END-GAP'

IF  (HIT_1 = 0) AND  (HIT_1B = 0)                CENS_GAP = 0
IF  (HIT_1B = 1) AND  (HIT_1 = 0) AND  (NUM_ACT = 0)  CENS_GAP = 1
IF  (HIT_1B = 1) AND  (HIT_1 = 0) AND  (NUM_ACT ge 1)  CENS_GAP = 2
IF  (HIT_1 = 1) AND  (HIT_1B = 0)                CENS_GAP = 3
IF  (HIT_1 = 1) AND  (HIT_1B = 1)                CENS_GAP = 4

VAR LAB  HIT_1B   'R has data in GAP-1B (thus, has a DOLI-GAP)'
        /  HIT_1   'R hit 1 or more GAP-1 question (thus, has an END-GAP)'
        /  CENS_GAP 'UNIVERES OF GAPS: using raw Census data & skips'

VAL LAB  CENS_GAP  0 'No gap at all'      1 'DOLI-gp, no work'
                  2 'DOLI-gp, worked'    3 'End-gp only'      4 'Both gaps'

```



```

*=====
*////
*////                                PART 2                                \\\
*////                                \\\
*////    IDENTIFY Rs who worked entire spell btwn interviews                \\\
*////                                \\\
*////    WORKALL = Use WEEKS array to identify Rs who worked the whole time \\\
*////                                \\\
*////    The variable WORKALL is created using Loops and the                 \\\
*////    WEEK array variables. It takes into account overlapping periods     \\\
*////    of work.                                                            \\\
*////                                \\\
****    NOTES:                                                                \\\
*****    NO REFERENCE DATES CREATED IF ENAME HAD NO ACTIVITY SINCE DOLI    \\\
*****    (if R3638000 and GAP-1B are blank [missing] no date was created)  \\\
*****    \\\
*****    MISSING DATES WERE NOT IMPUTED. That is, if the 'day' portion     \\\
*****    of DD/MM/YYYY is missing the entire date is missing.             \\\
*****    \\\
*****
*****

```

```

VECTOR WEEK(950)    /* create WEEK array variables, one for each week BID
VECTOR DAY(6650)    /* assigns 1=Working to each DAY variable

```

```
DO REPEAT
```

```

    START = REFST1 REFST2 REFST3 REFST4 REFST5 REFST6 REFST7
           REFST8 REFST9
    / END = REFEND1 REFEND2 REFEND3 REFEND4 REFEND5 REFEND6
           REFEND7 REFEND8 REFEND9
    / St.W = ST1.W ST2.W ST3.W ST4.W ST5.W ST6.W ST7.W
           ST8.W ST9.W
    / End.W = END1.W END2.W END3.W END4.W END5.W END6.W
           END7.W END8.W END9.W
    / WORKN = WORK1 WORK2 WORK3 WORK4 WORK5 WORK6 WORK7 WORK8
           WORK9
    / St.D = ST1.D ST2.D ST3.D ST4.D ST5.D ST6.D ST7.D
           ST8.D ST9.D
    / End.D = END1.D END2.D END3.D END4.D END5.D END6.D
           END7.D END8.D END9.D
    / WORKD = WORKD1 WORKD2 WORKD3 WORKD4 WORKD5 WORKD6 WORKD7
           WORKD8 WORKD9
    / DOLI.D = DOLI.1 DOLI.2 DOLI.3 DOLI.4 DOLI.5 DOLI.6 DOLI.7
           DOLI.8 DOLI.9

```

```

IF (END > DOI)          END = DOI
IF NVAL(START, END) = 1  MISSDATE = 1    /* marks cases with a missing job date

```

```
COMPUTE WORKD = CTIME.DAYS(END - START)
```

```

*** Convert employer dates to # DAYS after DOLI
COMPUTE ST.D = (CTIME.DAYS(START - DOLI)) + 1
COMPUTE END.D = (CTIME.DAYS(END - DOLI))
COMPUTE WORKD = CTIME.DAYS(END - START)

```

```

*** Loops for filling in 1=working
DO IF NVAL(ST.D, END.D) = 2
  .LOOP DAY# = ST.D to END.D
  .COMPUTE DAY(DAY#) = 1
END LOOP
END IF

```

```

** Convert employer dates to a week number (I#) with respect to the DOLI
** week#. 3.49 is subtracted to create ST.W to reduce the effects of rounding.
** But 2 cases are still misclassified so their ST.W value will be
** hand-edited later.

```

```

COMPUTE ST.W = RND((CTIME.DAYS(START - DOLI) - 3.49) / 7) + 1
COMPUTE END.W = RND(CTIME.DAYS(END - DOLI) / 7)
COMPUTE WORKN = RND(CTIME.DAYS(END - START) / 7) /* number of weeks working

*
*          HAND-EDIT data for selected cases
*
*          -----
** The DOLI-Gap is actually 4, 5, or 6 days so there is a **
** gap of 1 week, rounded up. This is consistent with what CASES found **
** and the values in R3893000 (Gap-1B). The respondents affected are **
** SERIAL = 4082, 1780 **
DO IF (CTIME.DAYS(START - DOLI) > 3) & (CTIME.DAYS(START - DOLI) < 7)
.COMPUTE ST.W = 2
END IF

IF NVAL(START, END) = 1 WORKW = -3
IF NVAL(START, END) ge 1 MISS_ST = 0
IF NVAL(END)=1 & SYSMIS(START) MISS_ST = 1

*** Loops for filling in 1=working
DO IF NVAL(START, END) = 2
.LOOP WRK# = ST.W to END.W
.COMPUTE WEEK(WRK#) = 1
END LOOP
END IF

END REPEAT

*****
** CREATE THE # DAYS WORKING and # WEEKS WORKING (WORKW)
**
COUNT WORKW.N = WEEK1 to WEEK950(1) /* # of weeks working
COUNT WORKD.N = DAY1 TO DAY6650(1) /* # of days working
COMPUTE WORKRAW = (WORKD.N / 7) /* # weeks working, not rounded
COMPUTE WORKW = RND(WORKD.N / 7) /* # weeks working, using DAYS ARRAY
*****

***** Create WORKALL *****
COMPUTE WORKALL = 0 /* default value
IF ABS(WORKD.N - DBID) < 3.5 WORKALL = 1

DO IF WORKALL ne 1
.IF MISSDATE = 1 WORKW = -3
END IF

VAR LAB WORKALL 'Rs worked entire spell between interviews'
/ WORKW.N '# WEEKS working, from WEEKS array'
/ WORKD.N '# of days working'
/ WORKW '# weeks working, using DAYS ARRAY'
/ WORK1 '# Weeks working for ENAME 1'
/ WORK2 '# Weeks working for ENAME 2'
/ WORK3 '# Weeks working for ENAME 3'
/ WORK4 '# Weeks working for ENAME 4'
/ MISS_ST 'A start date is missing (but END is ok)'

*****
*          Rs working entire spell *
** Assign WEMP1, WUMP1, and WOLF1 to respondents who worked **
** the entire spell, and those who have a missing date **
** and thus leaves the weeks-worked undefined. **
*=====

```

```

DO IF (WORKALL = 1)
  .COMPUTE WEMP1 = WORKW /* # weeks working computed from DAYS array
  .COMPUTE WUMP1 = 0
  .COMPUTE WOLF1 = 0
END IF

*=====
*//////////////////////////\////////////////////////////////////
*///////////////// PART 2 DOLI gap exists //////////////////////////////////
**
** Rs who did no work at all since DOLI, and those whose only
** gap in employment is from the DOLI to the earliest START DATE
** after the DOLI. (In other words, people who were not working
** on the DOLI, but who were working on the DOI have a DOLIGAP
** only).
**

DO IF (HIT_1B = 1) & (WORKALL ne 1) /* R did not work the whole time

COMPUTE DOLIGP0 = CTIME.DAYS(DOI - DOLI) /* This is the DOLIGAP for Rs who
/* didn't work at all

DO REPEAT
  START = REFST1 REFST2 REFST3 REFST4 REFST5 REFST6
  REFST7 REFST8 REFST9
  / END = REFEND1 REFEND2 REFEND3 REFEND4 REFEND5 REFEND6
  REFEND7 REFEND8 REFEND9
  / DOLI_GAP = DOLIGP1 to DOLIGP9

DO IF Nval(START, END) > 0 /** if there is at least one date
.COMPUTE DOLI_GAP = CTIME.DAYS(START - DOLI)
  .IF SYSMIS(START) DOLI_GAP = -3
  .IF SYSMIS(START) DOLIGAP0 = -3
END IF
END REPEAT

*** Find the smallest gap between DOLI and employer start dates
*** If a start date was missing then the shortest gap can't be determined
*** because the missing start-date might have been earlier than the
*** other Start-date or dates.

.COMPUTE MINGAP = MIN(DOLIGP0, DOLIGP1, DOLIGP2, DOLIGP3, DOLIGP4, DOLIGP5,
  DOLIGP6, DOLIGP7, DOLIGP8, DOLIGP9)
.MISS VAL MINGAP (-3) /* if any Start date is missing, DOLIGAP is missing

.COMPUTE DOLIGAP = RND(MINGAP / 7)

** -----
** HAND EDITS
** MINGAP is wrong because the START- and END-dates are messed up.
** For example, the last 3 Rs have DOLI = START1 = END1 (so the GAP = WBID)
IF SERIAL = 0368 DOLIGAP = WBID /** 113 weeks
IF SERIAL = 0607 DOLIGAP = WBID /** 108 weeks
IF SERIAL = 3014 DOLIGAP = WBID /** 115 weeks
IF SERIAL = 1106 DOLIGAP = WBID /** 107 weeks
IF SERIAL = 1816 DOLIGAP = WBID /** 107 weeks
IF SERIAL = 2640 DOLIGAP = WBID /** 106 weeks

.COMPUTE GAP_1B = R3893000 /** Gap-1b: looking for ALL, SOME, or NONE weeks?
.COMPUTE GAP_2B = R3893100 /** Gap-2b: # weeks looking for work

  .IF GAP_1B = 1 LOOKB = DOLIGAP /** looked for work "ALL" weeks in DOLI-GAP
  .IF GAP_1B = 1 OLFB = 0 /** was not looking for 0 weeks

```

```

      .IF GAP_1B = 2      LOOKB = GAP_2B      /* looked "SOME" weeks, # wks shown in GAP-2B
      .IF GAP_1B = 2      OLFB = R3893200      /** # weeks R was not looking

      .IF GAP_1B = 3      LOOKB = 0            /** looked for work "NONE" weeks in DOLI-GAP
      .IF GAP_1B = 3      OLFB = DOLIGAP      /** was not looking for entire gap (OLF)

** missing values
      .IF (GAP_1B < 0) or (GAP_2B < 0)      LOOKB = -3      /* will be declared missing
      .IF (GAP_1B < 0) or (GAP_2B < 0)      OLFB = -3      /* will be declared missing
END IF
EXECUTE

**** This R had a missing START date, but a closer look
*** shows that the DOLIGAP could be determined -- it's = R3893200
IF SERIAL = 4385      DOLIGAP = R3893200

*****
*** Assign WEMP2, WUMP2, WOLF2      ***
*** R's who have only a DOLI-GAP      ***
*****

DO IF (HIT_1B = 1) & (HIT_1 = 0)      /* DOLI GAP only
.COMPUTE WEMP2 = WORKW
.COMPUTE WUMP2 = LOOKB
.COMPUTE WOLF2 = OLFB
END IF

*****
execute

VAR LAB WEMP1      '# weeks wrking, WORKALL = 1'
      / WUMP1      '# Weeks looking, WORKALL = 1'
      / WOLF1      '# Weeks not looking, WORKALL = 1'
      / DOLIGAP      'DOLI-GAP ends this # weeks after DOLI'
      / WEMP2      '#wks working, DOLI-gp only'
      / WUMP2      '#wks looking, DOLI-gp only'
      / WOLF2      '#wks OLF, DOLI-gp only'

*****
*****
PART 4      END GAPS      *****
***
*** Determine END-GAP length for each Ename      *****
***
*
* These identifies the Start-Dt and End-Dt of each gap occurring at the
* of each Ename. The number assigned to the start and end of each gap is
* the number of weeks after the DOLI. Thus, if GST1.W = 20 it means that
* the gap after Ename 1 began 20 weeks after the DOLI. If GEND1.W = 35
* it means that the first gap ended 35 weeks after the DOLI. Therefore
* the duration of the gap was 15 weeks.
*****

**** UNIVERSE RESTRICTORS
DO IF (WORKALL ne 1)      /** Skips Rs who definitely worked the entire spell.
                        /** Only Rs who had a gap, or who weren't defined as
                        /** working the whole time because of missing dates
                        /** are going into this section.
      & (HIT_1 = 1)      /** select Rs who have data in GAP-1 array

```

```

.***** Ename in first column only (EMP 01)
.DO IF NVAL(EMP01)=1 & NVAL(EMP02 to EMP09) = 0
.COMPUTE GST1.W = END1.W + 1
.COMPUTE GEND1.W = WBID
.END IF

.*** Blank after 2
.DO IF NVAL(EMP02) = 1 & NVAL(EMP03 to EMP09) = 0
.COMPUTE GST2.W = END2.W + 1
.COMPUTE GEND2.W = WBID
.END IF

.*** Blank after 3
DO IF NVAL(EMP03) = 1 & NVAL(EMP04 to EMP09) = 0
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = WBID
END IF

.*** Blank after Column 4
DO IF NVAL(EMP04) = 1 & NVAL(EMP05 to EMP09) = 0
COMPUTE GST4.W = END4.W + 1
COMPUTE GEND4.W = WBID
END IF

.*** Blank after Column 5
DO IF NVAL(EMP05) = 1 & NVAL(EMP06 to EMP09) = 0
COMPUTE GST5.W = END5.W + 1
COMPUTE GEND5.W = WBID
END IF

.*** Blank after Column 6
DO IF NVAL(EMP06) = 1 & NVAL(EMP07 to EMP09) = 0
COMPUTE GST6.W = END6.W + 1
COMPUTE GEND6.W = WBID
END IF

.*** Blank after Column 7
DO IF NVAL(EMP07) = 1 & NVAL(EMP08 to EMP09) = 0
COMPUTE GST7.W = END7.W + 1
COMPUTE GEND7.W = WBID
END IF

.*** Blank after 8
DO IF NVAL(EMP08) = 1 & NVAL(EMP09) = 0
COMPUTE GST8.W = END8.W + 1
COMPUTE GEND8.W = WBID
END IF

.*** Blank after 9
DO IF NVAL(EMP09) = 1
COMPUTE GST9.W = END9.W + 1
COMPUTE GEND9.W = WBID
END IF

***** /// adjacent ENAMES \\*****
*** 1,2 adjacent -- Enames in column 1 & 2
DO IF NVAL(EMP01, EMP02) = 2
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W =ST2.W
END IF

```

```

*** 2,3 adjacent --   Enames in column 2 & 3
DO IF  NVAL(EMP02, EMP03) = 2
COMPUTE  GST2.W =  END2.W + 1
COMPUTE  GEND2.W =ST3.W
END IF

```

```

*** 3,4 adjacent --   Enames in column 3 & 4
DO IF  NVAL(EMP03, EMP04) = 2
COMPUTE  GST3.W =  END3.W + 1
COMPUTE  GEND3.W =ST4.W
END IF

```

```

*** 4,5 adjacent --   Enames in column 4 & 5
DO IF  NVAL(EMP04, EMP05) = 2
COMPUTE  GST4.W =  END4.W + 1
COMPUTE  GEND4.W =ST5.W
END IF

```

```

*** 5,6 adjacent --   Enames in column 5 & 6
DO IF  NVAL(EMP05, EMP06) = 2
COMPUTE  GST5.W =  END5.W + 1
COMPUTE  GEND5.W =ST6.W
END IF

```

```

*** 6,7 adjacent --   Enames in column 6 & 7
DO IF  NVAL(EMP06, EMP07) = 2
COMPUTE  GST6.W =  END6.W + 1
COMPUTE  GEND6.W = ST7.W
END IF

```

```

*** 7,8 adjacent --   Enames in column 7 & 8
DO IF  NVAL(EMP07, EMP08) = 2
COMPUTE  GST7.W =  END7.W + 1
COMPUTE  GEND7.W = ST8.W
END IF

```

```

*** 8,9 adjacent --   Enames in column 8 & 9
DO IF  NVAL(EMP08, EMP09) = 2
COMPUTE  GST8.W =  END8.W + 1
COMPUTE  GEND8.W = ST9.W
END IF

```

```

****   ///  EMP Rosters with blank columns  \\****
*** Column 1  &  3
DO IF  NVAL(EMP01, EMP03)=2   &   SYSMIS(EMP02)
COMPUTE  GST1.W =  END1.W + 1
COMPUTE  GEND1.W = ST3.W
END IF

```

```

*** Column 1  &  4
DO IF  NVAL(EMP01, EMP04)=2   &   NVAL(EMP02, EMP03)=0
COMPUTE  GST1.W =  END1.W + 1
COMPUTE  GEND1.W = ST4.W
END IF

```

```

*** Column 1  &  5
DO IF  NVAL(EMP01, EMP05)=2   &   NVAL(EMP02 to EMP04)=0
COMPUTE  GST1.W =  END1.W + 1
COMPUTE  GEND1.W = ST5.W
END IF

```

```

*** Column 1 & 6
DO IF NVAL(EMP01, EMP06)=2 & NVAL(EMP02 to EMP05)=0
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST6.W
END IF

```

```

*** Column 1 & 7
DO IF NVAL(EMP01, EMP07)=2 & NVAL(EMP02 to EMP06)=0
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST7.W
END IF

```

```

*** Column 1 & 8
DO IF NVAL(EMP01, EMP08)=2 & NVAL(EMP02 to EMP07)=0
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST8.W
END IF

```

```

*** Column 1 & 9
DO IF NVAL(EMP01, EMP09)=2 & NVAL(EMP02 to EMP08)=0
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST9.W
END IF

```

```

*** Column 2 & 4
DO IF NVAL(EMP02, EMP04)=2 & NVAL(EMP03)=0
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST4.W
END IF

```

```

*** Column 2 & 5
DO IF NVAL(EMP02, EMP05)=2 & NVAL(EMP03 to EMP04)=0
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST5.W
END IF

```

```

*** Column 2 & 6
DO IF NVAL(EMP02, EMP06)=2 & NVAL(EMP03 to EMP05)=0
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST6.W
END IF

```

```

*** Column 2 & 7
DO IF NVAL(EMP02, EMP07)=2 & NVAL(EMP03 to EMP06)=0
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST7.W
END IF

```

```

*** Column 2 & 8
DO IF NVAL(EMP02, EMP08)=2 & NVAL(EMP03 to EMP07)=0
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST8.W
END IF

```

```

*** Column 2 & 9
DO IF NVAL(EMP02, EMP09)=2 & NVAL(EMP03 to EMP08)=0
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST9.W
END IF

```

```

*** Column 3 & 5
DO IF NVAL(EMP03, EMP05)=2 & NVAL(EMP04)=0
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = ST5.W
END IF

*** Column 3 & 6
DO IF NVAL(EMP03, EMP06)=2 & NVAL(EMP04 to EMP05)=0
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = ST6.W
END IF

*** Column 3 & 7
DO IF NVAL(EMP03, EMP07)=2 & NVAL(EMP04 to EMP06)=0
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = ST7.W
END IF

*** Column 3 & 8
DO IF NVAL(EMP03, EMP08)=2 & NVAL(EMP04 to EMP07)=0
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = ST8.W
END IF

*** Column 3 & 9
DO IF NVAL(EMP03, EMP09)=2 & NVAL(EMP04 to EMP08)=0
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = ST9.W
END IF

*** Column 4 & 6
DO IF NVAL(EMP04, EMP06)=2 & NVAL(EMP05)=0
COMPUTE GST4.W = END4.W + 1
COMPUTE GEND4.W = ST6.W
END IF

*** Column 4 & 7
DO IF NVAL(EMP04, EMP07)=2 & NVAL(EMP05 to EMP06)=0
COMPUTE GST4.W = END4.W + 1
COMPUTE GEND4.W = ST7.W
END IF

*** Column 4 & 8
DO IF NVAL(EMP04, EMP08)=2 & NVAL(EMP05 to EMP07)=0
COMPUTE GST4.W = END4.W + 1
COMPUTE GEND4.W = ST8.W
END IF

*** Column 4 & 9
DO IF NVAL(EMP04, EMP09)=2 & NVAL(EMP05 to EMP08)=0
COMPUTE GST4.W = END4.W + 1
COMPUTE GEND4.W = ST9.W
END IF

*** Column 5 & 7
DO IF NVAL(EMP05, EMP07)=2 & NVAL(EMP06)=0
COMPUTE GST5.W = END5.W + 1
COMPUTE GEND5.W = ST7.W
END IF

```



```

*** Column 5 & 8
DO IF NVAL(EMP05, EMP08)=2 & NVAL(EMP06, EMP07)=0
COMPUTE GST5.W = END5.W + 1
COMPUTE GEND5.W = ST8.W
END IF

```

```

*** Column 5 & 9
DO IF NVAL(EMP05, EMP09)=2 & NVAL(EMP06 to EMP08)=0
COMPUTE GST5.W = END5.W + 1
COMPUTE GEND5.W = ST9.W
END IF

```

```

*** Column 6 & 8
DO IF NVAL(EMP06, EMP08)=2 & NVAL(EMP07)=0
COMPUTE GST6.W = END6.W + 1
COMPUTE GEND6.W = ST8.W
END IF

```

```

*** Column 6 & 9
DO IF NVAL(EMP06, EMP09)=2 & NVAL(EMP07 to EMP08)=0
COMPUTE GST6.W = END6.W + 1
COMPUTE GEND6.W = ST9.W
END IF

```

```

*** Column 7 & 9
DO IF NVAL(EMP07, EMP09)=2 & NVAL(EMP08)=0
COMPUTE GST7.W = END7.W + 1
COMPUTE GEND7.W = ST9.W
END IF
END IF

```

```

***** HAND EDIT GST.W & GEND.W DATA *****
***** Hand edit End-Gap week data for Ename 1
***** These cases need special attention because
***** the order of job dates didn't always follow
***** the order in which jobs were listed. For example,
***** the End1 date was after the ST2 date for these
***** Rs, so the GAP-1 length had to be computed by
***** looking at the start date of a later job, or
***** sometimes looking at the DOI. There are 21 Rs
***** in this DO-REPEAT loop.

```

```

DO REPEAT
CASEID = 3089 1278 4720 2741 3686
/ ST_WK = ST3.W WBID ST5.W WBID WBID
IF (SERIAL = CASEID) GST1.W = END1.W + 1
IF (SERIAL = CASEID) GEND1.W = ST_WK
IF (SERIAL = CASEID) HANDED = 1
END REPEAT

```

```

***** Hand edit End-Gap2 (GST2.W, GEND2.W) week data for Ename 2
DO REPEAT
CASEID = 13633 10511
/ ST_WK = WBID ST4.W

IF (SERIAL = CASEID) GST2.W = END2.W + 1
IF (SERIAL = CASEID) GEND2.W = ST_WK
IF (SERIAL = CASEID) HANDED = 2
END REPEAT

```

```

***** Other hand edits
IF (SERIAL = 2426) GST4.W = END4.W + 1
IF (SERIAL = 2426) GEND4.W = ST6.W

```

```

*****
****      COMPUTE # WEEKS IN EACH END-GAP      *****
****
**          BLANK Gap-dates if Census didn't find an End-Gap.
**      For example, if GAP-2-ARR-02 is blank then GST2.W,
**      GEND2.W, and GAP2.N are blanked out. This is because there
**      is no UMP or OLF data available if (a) there was
**      no End-gap or (b) there was actually a gap (as judged from
**      Ename reference dates) but the CAPI program missed it and
**      thus skipped questions GAP-1, GAP-2, and INTERCK.

*** some hand edits of data, only if dates are there and correct
IF (SERIAL=2276)   R3887400 = 28      /* definitely the correct gap length
IF (SERIAL=2267)   R3888000 = 0       /* END2 overlaps START3, so no gap

DO IF (WORKALL ne 1) & (CENS_GAP ge 3)

DO REPEAT
  GAP_1 = R3887200 R3887800 R3888400 R3889000 R3889600 R3890200
          R3890800 GAP_1.8 GAP_1.9
  / GAP_2 = R3887300 R3887900 GAP_2.3 R3889100 GAP_2.5 GAP_2.6
          GAP_2.7 GAP_2.8 GAP_2.9
  / Inter_ck = R3887400 R3888000 R3888600 R3889200 R3889800 R3890400
          R3891000 R3891600 INTCK.9
  / GAP_ST = GST1.W GST2.W GST3.W GST4.W GST5.W GST6.W GST7.W
          GST8.W GST9.W
  / GAP_END = GEND1.W GEND2.W GEND3.W GEND4.W GEND5.W GEND6.W
          GEND7.W GEND8.W GEND9.W
  / GAP_N = GAP1.N GAP2.N GAP3.N GAP4.N GAP5.N GAP6.N GAP7.N
          GAP8.N GAP9.N
  / GAPCENS = CGAP1 to CGAP09

IF SYSMIS(GAP_1)      GAP_ST = BLANK
IF SYSMIS(GAP_1)      GAP_END = BLANK

COMPUTE   GAP_N = (GAP_END - GAP_ST) + 1      /* # of weeks in END-gap

MISS VAL   GAP_2   INTER_CK(LO thru -1)
COMPUTE   GAPCENS =   GAP_2 + INTER_CK      /* Total # weeks not working
MISS VAL   GAP_2   INTER_CK()

IF NVAL(GAP_ST) = 1 & SYSMIS(GAP_END)      GAP_N = -3
IF NVAL(GAP_ST) = 1 & SYSMIS(GAP_END)      GAP_END = -3

END REPEAT
END IF

*****      Hand edits      *****
***      After reviewing each case it was determined that
***      the following Gap-length values are correct.
IF (SERIAL = 2267)   GAP2.N = 0
IF (SERIAL = 2289)   GAP2.N = 9      /* round to 9 instead of 10
IF (SERIAL = 2289)   GAP2.N = 9      /* round to 9 instead of 10

*****
****          PART 4.B  -- END-GAP EXISTS          *****
****
****          Creates LOOKA, OLFA, and END-GAP length variables      *****
*****

```

[illegible]

```

COMPUTE GAPCENS = LOOKA + OLFA          /* Total # weeks not working
END REPEAT
END IF                                  /* this ends the universe restrictor defined at the top

VAR LAB CGAP1 'Census END-gap1 length, in weeks'
    / CGAP2 'Census END-gap2 length, in weeks'

MISS VAL LOOKB OLFB (LO thru -1)

***** LOOP AND VECTOR *****
DO IF (WORKALL ne 1)                    /* and R didn't work the whole time
    & (HIT_1 = 1)                       /* only if R hit GAP-1 array questions,

VECTOR WEEK = WEEK1 to WEEK950
DO REPEAT
    St.WK = ST1.W ST2.W ST3.W ST4.W ST5.W ST6.W ST7.W ST8.W
           ST9.W
    / End.WK = END1.W END2.W END3.W END4.W END5.W END6.W END7.W
           END8.W END9.W
    / LOOKA = LOOKA1 LOOKA2 LOOKA3 LOOKA4 LOOKA5 LOOKA6 LOOKA7
           LOOKA8 LOOKA9
    / OLFA = OLFA1 OLFA2 OLFA3 OLFA4 OLFA5 OLFA6 OLFA7
           OLFA8 OLFA9
    / GAPST.W = GST1.W GST2.W GST3.W GST4.W GST5.W GST6.W GST7.W
           GST8.W GST9.W
    / GEND.W = GEND1.W GEND2.W GEND3.W GEND4.W GEND5.W GEND6.W GEND7.W
           GEND8.W GEND9.W
    / GAPA_N = GAP1.N GAP2.N GAP3.N GAP4.N GAP5.N GAP6.N GAP7.N
           GAP8.N GAP9.N

** missing values wont affect WEEKS
MISS VAL LOOKA OLFA GAPA_N LOOKB OLFB (LO thru -1)

***** MISSING DATES
.DO IF SYSMIS(ST.WK) & NVAL(END.WK) = 1          /* Enames with missing START date
.COMPUTE CK_MISS = 1
.LOOP I# = 1 to END.WK
.IF SYSMIS(WEEK(I#)) WEEK(I#) = -3
.END LOOP
.END IF

.DO IF SYSMIS(END.WK) & NVAL(ST.WK) = 1          /* Enames with missing END date
.COMPUTE CK_MISS = 1
.LOOP I# = ST.WK to (WBID - 1)
.IF SYSMIS(WEEK(I#)) WEEK(I#) = -3 /* from existing Start-week# to DOI week
number
.END LOOP
.END IF

***** UNEMPLOYED: Fill END-GAP wks lookng for work into WEEK array
LOOP I# = (GAPST.W) to (GAPST.W + LOOKA - 1)
COMPUTE WEEK(I#) = 2
END LOOP

***** OLF: Fill END-GAP wks not lookng for work (OLF) into WEEK array
LOOP I# = (GAPST.W + LOOKA) to (GAPST.W + LOOKA + OLFA - 1)
COMPUTE WEEK(I#) = 3
END LOOP

```

```

*** WORKING: Fill in the # of weeks worked, across all Enames
LOOP I# = ST.WK + 1 to (END.WK) /* fill wks working for Ename 1 into WEEK array
COMPUTE WEEK(I#) = 1
END LOOP
END REPEAT

***** DOLI GAP
LOOP I# = 1 to LOOKB /* fill wks looking during DOLIGAP into WEEK array
COMPUTE WEEK(I#) = 2
END LOOP

LOOP I# = (LOOKB +1) to (LOOKB + OLFB) /* fill wks not looking during DOLIGAP into
WEEK array
COMPUTE WEEK(I#) = 3
END LOOP

COUNT MISSWEEK = WEEK1 to WEEK950(-3)
.DO IF MISSWEEK = 0
.COMPUTE WEMP3 = WORKW
.count WEMPx = WEEK1 to WEEK950(1)
.COUNT WUMP3 = WEEK1 to WEEK950(2)
.COUNT WOLF3 = WEEK1 to WEEK950(3)
.END IF

IF MISSWEEK > 0 WEMP3 = -3
IF MISSWEEK > 0 WUMP3 = -3
IF MISSWEEK > 0 WOLF3 = -3
END IF

**** HAND EDITS
*** Some of these have an End-gap that was amissed
*** by CASES, so the weeks looking and OLF are undefined
DO IF ANY(SERIAL, 2426, 2588, 1661)
.COMPUTE WUMP3 = -3
.COMPUTE WOLF3 = -3
END IF

*****
*** Final Measure of weeks working!!!! ***
*** COMPILE 1, 2, 3 versions of WEMP, WUMP, WOLF ***
***
*****
DO IF NVAL(WEMP1, WUMP1, WOLF1) = 3
COMPUTE WORK = WEMP1
COMPUTE WUMP = WUMP1
COMPUTE WOLF = WOLF1
END IF

DO IF NVAL(WEMP2, WUMP2, WOLF2) = 3
COMPUTE WORK = WEMP2
COMPUTE WUMP = WUMP2
COMPUTE WOLF = WOLF2
END IF

DO IF NVAL(WEMP3, WUMP3, WOLF3) = 3
COMPUTE WORK = WEMP3
COMPUTE WUMP = WUMP3
COMPUTE WOLF = WOLF3
END IF

FINISH
/**** End of SPSS program creating WORK, WUMP, WOLF for 1997 Mature Women

```

```
#####
*   SPSS code used to create key variables WORK, WUMP, WOLF for YW cohort
#####
```

```
COMPUTE  SERIAL = R3495000
```

```
NUMERIC  BLANK          /* create a variable that is blank for all cases
```

```
** This selects the Rs who were interviewed in 1997
```

```
SELECT IF  (R3498800 > -3)
```

```
*=====
```

```
*** The individual D-M-Y variables are combined to form a single variable
```

```
*** measuring the date. This is done using one of the the SPSS date
```

```
*** aggregation functions. It makes adding, subtracting, and otherwise
```

```
*** comparing dates much easier.
```

```
**          DOLI
```

```
COMPUTE  DOLI = DATE.DMY(R3498900, R3499000, R3499100)
```

```
PRINT FORMAT  DOLI (DATE9)
```

```
**          DOI
```

```
COMPUTE  DOI = DATE.DMY(R3498600, R3498700, R3498800)
```

```
PRINT FORMAT  DOI (DATE9)
```

```
*=====
```

```
** Calculate Weeks Between Interview Dates  WBID.W
```

```
*=====
```

```
COMPUTE  WBID.RAW  = ((CTIME.DAYS(DOI - DOLI)) / 7)
```

```
VAR LAB  WBID.RAW  '# weeks btwn intvs, not rounded off yet (DOI - DOLI)'
```

```
COMPUTE  WBID  = RND((CTIME.DAYS(DOI - DOLI)) / 7)
```

```
VAR LAB  WBID  'WBID, rounded off'
```

```
EXECUTE
```

```
*=====
```

```
***** Employer start and End dates *****
```

```
*****
```

```
***** NO REFERENCE DATE CREATED IF ENAME HAD NO ACTIVITY SINCE DOLI
```

```
*****
```

```
***** NO MISSING DATES WERE IMPUTED (if day is missing the entire  
***** date is missing)
```

```
*****
```

```
NUMERIC  GAP_1.10      /* creates GAP-1.10 to square the list for DO REPEATs
```

```
DO REPEAT
```

```
REFST_D = R4236000 R4236600 R4237200 R4237800 R4238400 R4239000
```

```
R4239600 R4247200 R4247300 R4247400 R4248000
```

```
/ REFST_M = R4236100 R4236700 R4237300 R4237900 R4238500 R4239100
```

```
R4239700 R4240200 R4240700 R4247500 R4248100
```

```
/ REFST_Y = R4236200 R4236800 R4237400 R4238000 R4238600 R4239200
```

```
R4239800 R4240300 R4240800 R4247600 R4248200
```

```
/ REF_ST = REFST1 REFST2 REFST3 REFST4 REFST5 REFST6 REFST7
```

```
REFST8 REFST9 REFST10 REFST11
```

```
/ REFEN_D = R4236300 R4236900 R4237500 R4238100 R4238700 R4239300
```

```
R4239900 R4240400 R4240900 R4247700 R4248300
```

```
/ REFEN_M = R4236400 R4237000 R4237600 R4238200 R4238800 R4239400
```

```
R4240000 R4240500 R4241000 R4247800 R4248400
```

```
/ REFEN_Y = R4236500 R4237100 R4237700 R4238300 R4238900 R4239500
```

```
R4240100 R4240600 R4241100 R4247900 R4248500
```

```
/ REF_END = REFEND1 REFEND2 REFEND3 REFEND4 REFEND5 REFEND6 REFEND7
```

```

      REFEND8 REFEND9 REFEND10 REFEND11
/ CK_RSP_I = R3638000 R3658700 R3678200 R3698200 R3717500 R3736100
      R3752600 R3768800 R3784800 R3795500 R3803500
/ ST_DAY = ST1_D ST2_D ST3_D ST4_D ST5_D ST6_D ST7_D
      ST8_D ST9_D ST10_D ST11_D
/ ST_MO = ST1_M ST2_M ST3_M ST4_M ST5_M ST6_M ST7_M
      ST8_M ST9_M ST10_M ST11_M
/ ST_YEAR = ST1_Y ST2_Y ST3_Y ST4_Y ST5_Y ST6_Y ST7_Y
      ST8_Y ST9_Y ST10_Y ST11_Y
/ END_DAY = END1_D END2_D END3_D END4_D END5_D END6_D END7_D
      END8_D END9_D END10_D END11_D
/ END_MO = END1_M END2_M END3_M END4_M END5_M END6_M END7_M
      END8_M END9_M END10_M END11_M
/ END_YEAR = END1_Y END2_Y END3_Y END4_Y END5_Y END6_Y END7_Y
      END8_Y END9_Y END10_Y END11_Y
/ GAP_1 = R3887200 R3887800 R3888400 R3889000 R3889600 R3890200
      R3890800 R3891400 R3892000 GAP_1.10 R3892600

** This creates copies of the original D-M-Y variables that will be
** used to create the start and stop dates.
COMPUTE ST_DAY = REFST_D
COMPUTE ST_MO = REFST_M
COMPUTE ST_YEAR = REFST_Y
COMPUTE END_DAY = REFEN_D
COMPUTE END_MO = REFEN_M
COMPUTE END_YEAR = REFEN_Y

* Create dates only if Ename had activity since DOLI, or if an END-GAP exists
DO IF NVAL(CK_RSP_I) ge 1 or NVAL(GAP_1) ge 1

.COMPUTE REF_ST = DATE.DMY(Refst_D, Refst_M, Refst_Y)
.COMPUTE REF_END = DATE.DMY(RefEN_D, RefEN_M, RefEN_Y)

.** Hand edit the End date for these respondents who had
.** the wrong PC system date and thus the wrong REFEND1 date
.DO IF ANY(SERIAL, 0067, 4753, 4760)
.COMPUTE REFEND1 = DOI
.END IF
END IF

PRINT FORMAT REF_ST REF_END (DATE9)
END REPEAT
EXECUTE

VAR LAB REFEND1 'End date for Ename 1'
      / REFEND2 'End date for Ename 2'
      / REFEND3 'End date for Ename 3'
      / REFST1 'Start date for Ename 1'
      / REFST2 'Start date for Ename 2'

*=====
*////////// PART 1 //////////////////////////////////////////
*////////// Preliminary variables //////////////////////////////////////////
*////////// //////////////////////////////////////////
*=====
**
** GAP_CENS: This universe variable places respondents into 1 of 5
** groups.
** 0 = No gaps at all
** 1 = DOLI-gap only, No work at all
** 2 = DOLI-gap only, worked since DOLI
** 3 = End-gap only

```

```

**                4 = Both kinds of gaps
*****

**** Calculate DBID: # DAYS Between Interview Dates
COMPUTE DBID = CTIME.DAYS(DOI - DOLI)
VAR LAB DBID '# DAYS btwn intvs, not rounded off yet (DOI - DOLI)'

***** number of gap questions hit
COUNT NUM_GAP1 = R3887200, R3887800, R3888400, R3889000, R3889600, R3890200,
                R3890800, R3891400, R3892000, R3892600 (-2 thru HI)
RECODE NUM_GAP1 (1 thru HI = 1) (0=0) INTO HIT_1

COMPUTE HIT_1B = 0
IF (R3893000 ge -2) HIT_1B = 1 /* DOLI-GAP (Gap-1B)

*** This is needed because a few cases have an End-gap and a pair
*** dates, but CK-RSP-I is blank.

NUMERIC EMP01 to EMP11

IF NVAL(R3638000, R3887200) > 0 EMP01 = 1
IF NVAL(R3658700, R3887800) > 0 EMP02 = 1
IF NVAL(R3678200, R3888400) > 0 EMP03 = 1
IF NVAL(R3698200, R3889000) > 0 EMP04 = 1
IF NVAL(R3717500, R3889600) > 0 EMP05 = 1
IF NVAL(R3736100, R3890200) > 0 EMP06 = 1
IF NVAL(R3752600, R3890800) > 0 EMP07 = 1
IF NVAL(R3768800, R3891400) > 0 EMP08 = 1
IF NVAL(R3784800, R3892000) > 0 EMP09 = 1
IF NVAL(R3795500, GAP_1.10) > 0 EMP10 = 1
IF NVAL(R3803500, R3892600) > 0 EMP11 = 1

COUNT NUM_ACT = EMP01, EMP02, EMP03, EMP04, EMP05, EMP06, EMP07, EMP08, EMP09,
                EMP10, EMP11(1)
RECODE NUM_ACT (0=0) (1 THRU HI=1) INTO ACTIV
VAR LAB NUM_ACT '# Enames with activity, or and END-GAP'
/ EMP01 'Ename 01 has activity, or END-GAP, or both'
/ EMP02 'Ename 02 has activity, or END-GAP, or both'
/ ACTIV 'Activity since DOLI, or END-GAP'

IF (HIT_1 = 0) AND (HIT_1B = 0) CENS_GAP = 0
IF (HIT_1B = 1) AND (HIT_1 = 0) AND (NUM_ACT = 0) CENS_GAP = 1
IF (HIT_1B = 1) AND (HIT_1 = 0) AND (NUM_ACT ge 1) CENS_GAP = 2
IF (HIT_1 = 1) AND (HIT_1B = 0) CENS_GAP = 3
IF (HIT_1 = 1) AND (HIT_1B = 1) CENS_GAP = 4

VAR LAB HIT_1B 'R has data in GAP-1B (thus, has a DOLI-GAP)'
/ HIT_1 'R hit 1 or more GAP-1 question (thus, has an END-GAP)'
/ CENS_GAP 'UNIVERES OF GAPS: using raw Census data & skips'

VAL LAB CENS_GAP 0 'No gap at all' 1 'DOLI-gp, no work'
                2 'DOLI-gp, worked' 3 'End-gp only' 4 'Both gaps'

*=====
*//////
*////// PART 2
*//////
*////// IDENTIFY Rs who worked entire spell btwn interviews
*//////
*////// WORKALL = Use WEEKS array to identify Rs who worked the whole time
*//////
*////// The variable WORKALL is created using Loops and the WEEK array
*////// variables. It takes into account overlapping periods of work.
*//////

```



```

      *** NOTES:
***** NO REFERENCE DATES CREATED IF ENAME HAD NO ACTIVITY SINCE DOLI
***** (if R3638000 and GAP-1B are blank [missing] no date was created)
***** MISSING DATES WERE NOT IMPUTED. That is, if the 'day' portion
***** of DD/MM/YYYY is missing the entire date is missing.
*****
*****
*****
*****

```

```
VECTOR WEEK(950) /* create WEEK array variables, one for each week BID
VECTOR DAY(6650) /* assigns 1=Working to each DAY variable
```

DO REPEAT

```

START = REFST1 REFST2 REFST3 REFST4 REFST5 REFST6
      REFST7 REFST8 REFST9 REFST10 REFST11
/ END = REFEND1 REFEND2 REFEND3 REFEND4 REFEND5 REFEND6
      REFEND7 REFEND8 REFEND9 REFEND10 REFEND11
/ St.W = ST1.W ST2.W ST3.W ST4.W ST5.W ST6.W ST7.W
        ST8.W ST9.W ST10.W ST11.W
/ End.W = END1.W END2.W END3.W END4.W END5.W END6.W
        END7.W END8.W END9.W END10.W END11.W
/ WORKN = WORK1 WORK2 WORK3 WORK4 WORK5 WORK6 WORK7 WORK8
        WORK9 WORK10 WORK11
/ St.D = ST1.D ST2.D ST3.D ST4.D ST5.D ST6.D ST7.D
        ST8.D ST9.D ST10.D ST11.D
/ End.D = END1.D END2.D END3.D END4.D END5.D END6.D
        END7.D END8.D END9.D END10.D END11.D
/ WORKD = WORKD1 WORKD2 WORKD3 WORKD4 WORKD5 WORKD6 WORKD7
        WORKD8 WORKD9 WORKD10 WORKD11
/ DOLI.D = DOLI.1 to DOLI.11

```

```

IF (END > DOI)          END = DOI
IF NVAL(START, END) = 1  MISDATE = 1    /* marks cases with a missing job date

```

```
COMPUTE WORKD = CTIME.DAYS (END - START)
```

```

*** Convert employer dates to # DAYS after DOLI
COMPUTE ST.D = (CTIME.DAYS(START - DOLI)) + 1
COMPUTE END.D = (CTIME.DAYS(END - DOLI))
COMPUTE WORKD = CTIME.DAYS(END - START)

```

```

***  Loops for filling in 1=working
DO IF  NVAL(ST.D, END.D) = 2
.LOOP DAY# = ST.D to END.D
.COMPUTE DAY(DAY#) = 1
END LOOP
END IF

```

```

**      Convert employer dates to a week number (I#) with respect to the DOLI
**      week#. 3.49 is subtracted to create ST.W to reduce the effects of rounding.
**      But 5 cases are still misclassified so their ST.W value will be
**      hand-edited later.
COMPUTE  ST.W = RND((CTIME.DAYS(START - DOLI) - 3.49) / 7) + 1
COMPUTE  END.W = RND(CTIME.DAYS(END - DOLI) / 7)
COMPUTE  WORKN = RND(CTIME.DAYS(END - START) / 7) /* number of weeks working

```

* HAND-EDIT data for selected cases

```

** The DOLI-Gap is actually 4, 5, or 6 days so there is a
** gap of 1 week, rounded up. This is consistent with what CASES found
** and the values in R3893000 (Gap-1B). The respondents affected are
** SERIAL = 1507, 1831, 0985)
**

```

```
DO IF (CTIME.DAYS(START - DOLI) > 3) & (CTIME.DAYS(START - DOLI) < 7)
.COMPUTE ST.W = 2
END IF
```

```
IF NVAL(START, END) = 1      WORKW = -3
IF NVAL(START, END) ge 1     MISS_ST = 0
IF NVAL(END)=1 & SYSMIS(START) MISS_ST = 1
```

```
*** Loops for filling in 1=working
DO IF NVAL(START, END) = 2
.LOOP WRK# = ST.W to END.W
.COMPUTE WEEK(WRK#) = 1
END LOOP
END IF
END REPEAT
```

```
**      -- HAND-EDIT ---
**      End-date is 3 days before DOI, but CASES found no END-gap
**      so the WEEK109=1, and END2.W = 109 instead of 108
DO IF (SERIAL = 2188)
.COMPUTE      END2.W = 109
.COMPUTE      WEEK109 = 1      /* last week in array is 'worked'
.COMPUTE      WORK2 = 109      /* add 1 to # weeks wormed for ENAME2
.RECODE DAY759, DAY760, DAY761 (SYSMIS=1) /* worked on these days
END IF
```

```
**      End2-Date is 5 days before DOI, but CASES found no end-gap2,
**      so WEEK110 = 1, and DAY773 to DAY767 = 1
DO IF (SERIAL = 0863)
.COMPUTE      END2.W = 109
.COMPUTE      WEEK110 = 1      /* make last week in array a worked week
.COMPUTE      WORK2 = 12      /* add 1 to # weeks wormed for ENAME2
.RECODE DAY763, DAY764, DAY765, DAY766, DAY767 (SYSMIS=1) /* worked on these days
END IF
```

```
*****
```

```
** CREATE THE # DAYS WORKING and # WEEKS WORKING (WORKW)
```



```
COUNT      WORKW.N  =  WEEK1 to WEEK950(1)    /* # of weeks working
```



```
COUNT      WORKD.N  =  DAY1 TO DAY6650(1)  /* # of days working
```



```

COMPUTE  WORKRAW =      (WORKD.N / 7)      /* # weeks working, not rounded
COMPUTE  WORKW  =      RND(WORKD.N / 7)    /* # weeks working, using DAYS ARRAY
*****

```

```

***** Create WORKALL *****

```

```

** There are 7 cases with a gap of 3, 4, or 5 days between
** the end of 1 Ename and the start of the next. None of these
** has data in Gap-1 array, so there is no end-gap info
** available. Thus the R worked the whole time.

```

```

COMPUTE  WORKALL = 0      /* default value
IF      ABS(WORKD.N - DBID) < 3.5    WORKALL = 1

```

```

DO IF ANY(SERIAL, 0130, 0350, 1598, 1852, 1881, 4710, 4545)

```

```

.COMPUTE  WORKALL = 1

```

```

.COMPUTE  CENS_GAP = 0

```

```

END IF

```

```

***** This R worked the whole time, even though a date is missing
IF (SERIAL = 0367)      WORKALL = 1  /** Enames 2 and 5 DEFINITELY overlapped
IF (SERIAL = 0367)      WORKW  = WBID
IF (SERIAL = 0367)      R3887800 = BLANK  /* blank because there is no gap

```

```

DO IF WORKALL ne 1
  .IF MISSDATE = 1      WORKW = -3
END IF

```

```

VAR LAB  WORKALL  'Rs worked entire spell between interviews'
/        WORKW.N   '# WEEKS working, from WEEKS array'
/        WORKD.N   '# of days working'
/        WORKW     '# weeks working, using DAYS ARRAY'
/        WORK1     '# Weeks working for ENAME 1'
/        WORK2     '# Weeks working for ENAME 2'
/        WORK3     '# Weeks working for ENAME 3'
/        WORK4     '# Weeks working for ENAME 4'
/        MISS_ST   'A start date is missing (but END is ok)'

```

```

*****
*              Rs working entire spell                      *
**      Assign  WEMP1, WUMP1, and WOLF1 to respondents who worked      **
**      the entire spell, and those who have a missing date          **
**      and thus leaves the weeks-worked undefined.                  **
**                                                                    **
**              WORKED ENTIRE SPELL                                **
**                                                                    **
*=====

```

```

DO IF (WORKALL = 1)
  .COMPUTE  WEMP1 = WORKW      /* # weeks working computed from DAYS array
  .COMPUTE  WUMP1 = 0
  .COMPUTE  WOLF1 = 0
END IF

```

```

VAR LAB  WEMP1  'UNIVERSE: Rs who worked the entire spell (=WBID)'
        WUMP1  'UNIVERSE: Rs who worked the entire spell (=0)'
        WOLF1  'UNIVERSE: Rs who worked the entire spell (=0)'

```

```

*=====
*//////////////////////////\\
*/////////      PART 2      DOLI gap exists      \\\
**                                                    **
**      Rs who did no work at all since DOLI, and those whose only    **
**      gap in employment is from the DOLI to the earliest START DATE  **
**      after the DOLI. (In other words, people who were not working   **
**      on the DOLI, but who were working on the DOI have a DOLIGAP    **
**      only).                                                           **

```

```

DO IF (HIT_1B = 1) & (WORKALL ne 1)  /* R did not work the whole time

```

```

COMPUTE  DOLIGP0 =  CTIME.DAYS(DOI - DOLI)  /* This is the DOLIGAP for Rs who
                                           /* didn't work at all

```

```

DO REPEAT
  START =  REFST1 REFST2 REFST3 REFST4 REFST5 REFST6
           REFST7 REFST8 REFST9 REFST10 REFST11
  / END  =  REFEND1 REFEND2 REFEND3 REFEND4 REFEND5 REFEND6
           REFEND7 REFEND8 REFEND9 REFEND10 REFEND11
  / DOLI_GAP = DOLIGP1 to DOLIGP11

```

```

DO IF Nval(START, END) > 0      /** if there is at least one date
.COMPUTE DOLI_GAP = CTIME.DAYS(START - DOLI)
  .IF SYSMIS(START)      DOLI_GAP = -3
  .IF SYSMIS(START)      DOLIGAP0 = -3
END IF
END REPEAT

```

```

*** Find the smallest gap between DOLI and employer start dates
*** If a start date was missing then the shortest gap can't be determined
*** because the missing start-date might have been earlier than the
*** other Start-date or dates.

```

```

.COMPUTE MINGAP = MIN(DOLIGP0, DOLIGP1, DOLIGP2, DOLIGP3, DOLIGP4, DOLIGP5,
                     DOLIGP6, DOLIGP7, DOLIGP8, DOLIGP9, DOLIGP10, DOLIGP11)
.MISS VAL MINGAP (-3) /* if any Start date is missing, DOLIGAP is missing

```

```

.COMPUTE DOLIGAP = RND(MINGAP / 7)

```

```

** -----

```

```

** HAND EDIT: These Rs had a missing START date, but a closer look
** shows that the DOLIGAP could be determined -- it's = R3893200
IF SERIAL = 0895 DOLIGAP = R3893200
IF SERIAL = 2276 DOLIGAP = R3893200
IF SERIAL = 1894 R3893000 = BLANK /* R worked whole time, has no DOLIGAP
IF SERIAL = 1894 DOLIGAP = 0

```

```

.COMPUTE GAP_1B = R3893000 /* Gap-1b: looking for ALL, SOME, or NONE
weeks?

```

```

.COMPUTE GAP_2B = R3893100 /* Gap-2b: # weeks looking for work

```

```

  .IF GAP_1B = 1 LOOKB = DOLIGAP /* looked for work "ALL" weeks in DOLI-GAP
  .IF GAP_1B = 1 OLFB = 0 /* was not looking for 0 weeks

```

```

  .IF GAP_1B = 2 LOOKB = GAP_2B /* looked "SOME" weeks, # wks shown in GAP-2B
  .IF GAP_1B = 2 OLFB = R3893200 /* # weeks R was not looking (shown in INTER-
CHK)

```

```

  .IF GAP_1B = 3 LOOKB = 0 /* looked for work "NONE" weeks in DOLI-GAP
  .IF GAP_1B = 3 OLFB = DOLIGAP /* was not looking for entire gap (OLF)

```

```

** missing values

```

```

  .IF (GAP_1B < 0) or (GAP_2B < 0) LOOKB = -3 /* will be declared missing
  .IF (GAP_1B < 0) or (GAP_2B < 0) OLFB = -3 /* will be declared missing

```

```

  .IF (MINGAP = -3) LOOKB = -3
  .IF (MINGAP = -3) OLFB = -3
  .IF (HIT_1B = 1) & SYSMIS(DOLIGAP) LOOKB = -3
  .IF (HIT_1B = 1) & SYSMIS(DOLIGAP) OLFB = -3

```

```

END IF
EXECUTE

```

```

*****
***          Assign WEMP2, WUMP2, WOLF2          ***
*** R's who have only a DOLI-GAP                ***
*****

```

```

DO IF (HIT_1B = 1) & (HIT_1 = 0) /* DOLI GAP only
.COMPUTE WEMP2 = WORKW
.COMPUTE WUMP2 = LOOKB
.COMPUTE WOLF2 = OLFB
END IF

```

```

VAR LAB    WEMP2    'UNIVERSE: DOLI->START gap only (Currtly working)'
          WUMP2    'UNIVERSE: DOLI->START gap only (Currtly working)'
          WOLF2    'UNIVERSE: DOLI->START gap only (Currtly working)'

```

```

*****
****      Hand edit data for Rs with Gaps and Work discrepancies      **
**      ----- **

```

```

**      2 cases have dates to compute WEMP1, but there is no
**      gaps data, so WUMP1 or WOLF1 are undefined.
DO IF ANY(SERIAL, 3160, 4096)
.COMPUTE WEMP1 = WORKW /* # weeks working computed from DAYS array
.COMPUTE WUMP1 = -3
.COMPUTE WOLF1 = -3
END IF

```

```

**      2 Cases have no gap data but Job dates are missing,
**      so the weeks looking and OLF are undefined.
DO IF ANY(SERIAL, 2845, 1278)
.COMPUTE WEMP1 = -3
.COMPUTE WUMP1 = -3
.COMPUTE WOLF1 = -3
END IF

```

```

VAR LAB WEMP1      '# weeks wrking, WORKALL = 1'
      / WUMP1      '# Weeks looking, WORKALL = 1'
      / WOLF1      '# Weeks not looking, WORKALL = 1'
      / DOLIGAP    'DOLI-GAP ends this # weeks after DOLI'
      / WEMP2      '#wks working, DOLI-gp only'
      / WUMP2      '#wks looking, DOLI-gp only'
      / WOLF2      '#wks OLF, DOLI-gp only'

```

```

*****
*****                                PART 4  END GAPS                                *****
*****      Determine END-GAP length for each Ename      *****
*
*      These identifies the Start-Dt and End-Dt of each gap occurring at the
*      of each Ename. The number assigned to the start anf end of each gap is
*      the number of weeks after the DOLI. Thus, if GST1.W = 20 it means that
*      the gap after Ename 1 began 20 weeks after the DOLI. If GEND1.W = 35
*      it means that the first gap ended 35 weeks after the DOLI. Therefore
*      the duration of the gap was 15 weeks.
*****

```

```

*****      UNIVERSE RESTRICTORS
DO IF (WORKALL ne 1) /* Skips Rs who definitely worked the entire spell.
                  /* Only Rs who had a gap, or who weren't defined as
                  /* working the whole time because of missing dates
                  /* are going into this section.
      & (HIT_1 = 1) /* select Rs who have data in GAP-1 array

***** Ename in first column only (EMP 01)
DO IF NVAL(EMP01)=1 & NVAL(EMP02 to EMP11) = 0
.COMPUTE GST1.W = END1.W + 1
.COMPUTE GEND1.W = WBID
.END IF

```

```

.*** Blank after 2
.DO IF NVAL(EMP02) = 1 & NVAL(EMP03 to EMP11) = 0
.COMPUTE GST2.W = END2.W + 1
.COMPUTE GEND2.W = WBID
.END IF

```

```

*** Blank after 3
DO IF NVAL(EMP03) = 1 & NVAL(EMP04 to EMP11) = 0
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = WBID
END IF

```

```

*** Blank after Column 4
DO IF NVAL(EMP04) = 1 & NVAL(EMP05 to EMP11) = 0
COMPUTE GST4.W = END4.W + 1
COMPUTE GEND4.W = WBID
END IF

```

```

*** Blank after Column 5
DO IF NVAL(EMP05) = 1 & NVAL(EMP06 to EMP11) = 0
COMPUTE GST5.W = END5.W + 1
COMPUTE GEND5.W = WBID
END IF

```

```

*** Blank after Column 6
DO IF NVAL(EMP06) = 1 & NVAL(EMP07 to EMP11) = 0
COMPUTE GST6.W = END6.W + 1
COMPUTE GEND6.W = WBID
END IF

```

```

*** Blank after Column 7
DO IF NVAL(EMP07) = 1 & NVAL(EMP08 to EMP11) = 0
COMPUTE GST7.W = END7.W + 1
COMPUTE GEND7.W = WBID
END IF

```

```

*** Blank after 8
DO IF NVAL(EMP08) = 1 & NVAL(EMP09 to EMP11) = 0
COMPUTE GST8.W = END8.W + 1
COMPUTE GEND8.W = WBID
END IF

```

```

*** Blank after 9
DO IF NVAL(EMP09) = 1 & NVAL(EMP10, EMP11) = 0
COMPUTE GST9.W = END9.W + 1
COMPUTE GEND9.W = WBID
END IF

```

```

*** Blank after 10
DO IF NVAL(EMP10) = 1 & NVAL(EMP11) = 0
COMPUTE GST10.W = END10.W + 1
COMPUTE GEND10.W = WBID
END IF

```

```

*** Name in Column 11
DO IF NVAL(EMP11) = 1 & NVAL(EMP01 to EMP10) = 0
COMPUTE GST11.W = END11.W + 1
COMPUTE GEND11.W = WBID
END IF

```

```

***** /// adjacent ENAMES \\\*****
*** 1,2 adjacent -- Enames in column 1 & 2
DO IF NVAL(EMP01, EMP02) = 2
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST2.W
END IF

*** 2,3 adjacent -- Enames in column 2 & 3
DO IF NVAL(EMP02, EMP03) = 2
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST3.W
END IF

*** 3,4 adjacent -- Enames in column 3 & 4
DO IF NVAL(EMP03, EMP04) = 2
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = ST4.W
END IF

*** 4,5 adjacent -- Enames in column 4 & 5
DO IF NVAL(EMP04, EMP05) = 2
COMPUTE GST4.W = END4.W + 1
COMPUTE GEND4.W = ST5.W
END IF

*** 5,6 adjacent -- Enames in column 5 & 6
DO IF NVAL(EMP05, EMP06) = 2
COMPUTE GST5.W = END5.W + 1
COMPUTE GEND5.W = ST6.W
END IF

*** 6,7 adjacent -- Enames in column 6 & 7
DO IF NVAL(EMP06, EMP07) = 2
COMPUTE GST6.W = END6.W + 1
COMPUTE GEND6.W = ST7.W
END IF

*** 7,8 adjacent -- Enames in column 7 & 8
DO IF NVAL(EMP07, EMP08) = 2
COMPUTE GST7.W = END7.W + 1
COMPUTE GEND7.W = ST8.W
END IF

*** 8,9 adjacent -- Enames in column 8 & 9
DO IF NVAL(EMP08, EMP09) = 2
COMPUTE GST8.W = END8.W + 1
COMPUTE GEND8.W = ST9.W
END IF

*** 9,10 adjacent -- Enames in column 9 & 10
DO IF NVAL(EMP09, EMP10) = 2
COMPUTE GST9.W = END9.W + 1
COMPUTE GEND9.W = ST10.W
END IF

*** 10,11 adjacent -- Enames in column 10 & 11
DO IF NVAL(EMP10, EMP11) = 2
COMPUTE GST10.W = END10.W + 1
COMPUTE GEND10.W = ST11.W
END IF

```

```

*** 11 -- Ename in column 11
DO IF NVAL(EMP11) = 1
COMPUTE GST11.W = END11.W + 1
COMPUTE GEND11.W = WBID
END IF

**** /// EMP Rosters with blank columns \\****
*** Column 1 & 3
DO IF NVAL(EMP01, EMP03)=2 & SYSMIS(EMP02)
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST3.W
END IF

*** Column 1 & 4
DO IF NVAL(EMP01, EMP04)=2 & NVAL(EMP02, EMP03)=0
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST4.W
END IF

*** Column 1 & 5
DO IF NVAL(EMP01, EMP05)=2 & NVAL(EMP02 to EMP04)=0
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST5.W
END IF

*** Column 1 & 6
DO IF NVAL(EMP01, EMP06)=2 & NVAL(EMP02 to EMP05)=0
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST6.W
END IF

*** Column 1 & 7
DO IF NVAL(EMP01, EMP07)=2 & NVAL(EMP02 to EMP06)=0
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST7.W
END IF

*** Column 1 & 8
DO IF NVAL(EMP01, EMP08)=2 & NVAL(EMP02 to EMP07)=0
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST8.W
END IF

*** Column 1 & 9
DO IF NVAL(EMP01, EMP09)=2 & NVAL(EMP02 to EMP08)=0
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST9.W
END IF

*** Column 1 & 10
DO IF NVAL(EMP01, EMP10)=2 & NVAL(EMP02 to EMP09)=0
COMPUTE GST1.W = END1.W + 1
COMPUTE GEND1.W = ST10.W
END IF

*** Column 2 & 4
DO IF NVAL(EMP02, EMP04)=2 & NVAL(EMP03)=0
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST4.W
END IF

```

```

*** Column 2 & 5
DO IF NVAL(EMP02, EMP05)=2 & NVAL(EMP03 to EMP04)=0
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST5.W
END IF

```

```

*** Column 2 & 6
DO IF NVAL(EMP02, EMP06)=2 & NVAL(EMP03 to EMP05)=0
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST6.W
END IF

```

```

*** Column 2 & 7
DO IF NVAL(EMP02, EMP07)=2 & NVAL(EMP03 to EMP06)=0
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST7.W
END IF

```

```

*** Column 2 & 8
DO IF NVAL(EMP02, EMP08)=2 & NVAL(EMP03 to EMP07)=0
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST8.W
END IF

```

```

*** Column 2 & 9
DO IF NVAL(EMP02, EMP09)=2 & NVAL(EMP03 to EMP08)=0
COMPUTE GST2.W = END2.W + 1
COMPUTE GEND2.W = ST9.W
END IF

```

```

*** Column 3 & 5
DO IF NVAL(EMP03, EMP05)=2 & NVAL(EMP04)=0
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = ST5.W
END IF

```

```

*** Column 3 & 6
DO IF NVAL(EMP03, EMP06)=2 & NVAL(EMP04 to EMP05)=0
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = ST6.W
END IF

```

```

*** Column 3 & 7
DO IF NVAL(EMP03, EMP07)=2 & NVAL(EMP04 to EMP06)=0
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = ST7.W
END IF

```

```

*** Column 3 & 8
DO IF NVAL(EMP03, EMP08)=2 & NVAL(EMP04 to EMP07)=0
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = ST8.W
END IF

```

```

*** Column 3 & 9
DO IF NVAL(EMP03, EMP09)=2 & NVAL(EMP04 to EMP08)=0
COMPUTE GST3.W = END3.W + 1
COMPUTE GEND3.W = ST9.W
END IF

```



```

*** Column 4 & 6
DO IF NVAL(EMP04, EMP06)=2 & NVAL(EMP05)=0
COMPUTE GST4.W = END4.W + 1
COMPUTE GEND4.W = ST6.W
END IF

```

```

*** Column 4 & 7
DO IF NVAL(EMP04, EMP07)=2 & NVAL(EMP05 to EMP06)=0
COMPUTE GST4.W = END4.W + 1
COMPUTE GEND4.W = ST7.W
END IF

```

```

*** Column 4 & 8
DO IF NVAL(EMP04, EMP08)=2 & NVAL(EMP05 to EMP07)=0
COMPUTE GST4.W = END4.W + 1
COMPUTE GEND4.W = ST8.W
END IF

```

```

*** Column 4 & 9
DO IF NVAL(EMP04, EMP09)=2 & NVAL(EMP05 to EMP08)=0
COMPUTE GST4.W = END4.W + 1
COMPUTE GEND4.W = ST9.W
END IF

```

```

*** Column 5 & 7
DO IF NVAL(EMP05, EMP07)=2 & NVAL(EMP06)=0
COMPUTE GST5.W = END5.W + 1
COMPUTE GEND5.W = ST7.W
END IF

```

```

*** Column 5 & 8
DO IF NVAL(EMP05, EMP08)=2 & NVAL(EMP06, EMP07)=0
COMPUTE GST5.W = END5.W + 1
COMPUTE GEND5.W = ST8.W
END IF

```

```

*** Column 5 & 9
DO IF NVAL(EMP05, EMP09)=2 & NVAL(EMP06 to EMP08)=0
COMPUTE GST5.W = END5.W + 1
COMPUTE GEND5.W = ST9.W
END IF

```

```

*** Column 6 & 8
DO IF NVAL(EMP06, EMP08)=2 & NVAL(EMP07)=0
COMPUTE GST6.W = END6.W + 1
COMPUTE GEND6.W = ST8.W
END IF

```

```

*** Column 6 & 9
DO IF NVAL(EMP06, EMP09)=2 & NVAL(EMP07 to EMP08)=0
COMPUTE GST6.W = END6.W + 1
COMPUTE GEND6.W = ST9.W
END IF

```

```

*** Column 7 & 9
DO IF NVAL(EMP07, EMP09)=2 & NVAL(EMP08)=0
COMPUTE GST7.W = END7.W + 1
COMPUTE GEND7.W = ST9.W
END IF

```

*** Column 7 & 10

```
DO IF NVAL(EMP07, EMP10)=2 & NVAL(EMP08 to EMP09)=0
  COMPUTE GST7.W = END7.W + 1
  COMPUTE GEND7.W = ST10.W
END IF
END IF
```

***** HAND EDIT GST.W & GEND.W DATA *****

```
***** Hand edit End-Gap week data for Ename 1
***** These cases need special attention because
***** the order of job dates didn't always follow
***** the order in which jobs were listed. For example,
***** the End1 date was after the ST2 date for these
***** Rs, so the GAP1 length had to be computed by
***** looking at the start date of a later job, or
***** sometimes looking at the DOI.
```

DO REPEAT

```
  CASEID = 4996 1649 2586 1665 0919 1243 3905
           4724 0926 4026 4801 2069 1981
           0953 3753 3699 0285 4954 0230
  / ST_WK = WBID ST4.W ST3.W ST3.W WBID WBID ST4.W
           ST3.W WBID WBID ST5.W ST3.W ST4.W
           ST4.W WBID WBID ST3.W ST3.W ST3.W
```

```
IF (SERIAL = CASEID) GST1.W = END1.W + 1
IF (SERIAL = CASEID) GEND1.W = ST_WK
IF (SERIAL = CASEID) HANDED = 1
END REPEAT
```

***** Hand edit End-Gap2 (GST2.W, GEND2.W) week data for Ename 2

DO REPEAT

```
  CASEID = 2831 1118 1289 3930 1898 3674 1738 0786 2917
  / ST_WK = WBID WBID WBID ST5.W ST4.W ST4.W WBID ST4.W ST5.W
```

```
IF (SERIAL = CASEID) GST2.W = END2.W + 1
IF (SERIAL = CASEID) GEND2.W = ST_WK
IF (SERIAL = CASEID) HANDED = 2
```

END REPEAT

***** Other hand edits

```
IF (SERIAL = 3891) GST5.W = END5.W + 1 /* END5, ST4 both in Sep, but ST4
IF (SERIAL = 3891) GEND5.W = ST4.W /* is missing, so GEND5.W = missing
```

```
IF (SERIAL = 0786) GST4.W = END4.W + 1
IF (SERIAL = 0786) GEND4.W = ST3.W
```

```
IF (SERIAL = 0895) GST6.W = END6.W + 1
IF (SERIAL = 0895) GEND6.W = ST5.W
```

```
IF (SERIAL = 3891) GST5.W = END5.W + 1
IF (SERIAL = 3891) GEND5.W = ST4.W /* ST4 is missing, so GAP5.N = -3
```

```
DO IF (SERIAL = 3266) /* Enames were out of order
  COMPUTE GST5.W = END5.W + 1
  COMPUTE GEND5.W = ST8.W
  COMPUTE GST8.W = END8.W + 1
  COMPUTE GEND8.W = ST7.W
END IF
```

```

*****
****      COMPUTE # WEEKS IN EACH END-GAP      *****
**      BLANK Gap-dates if Census didn't find an End-Gap.
**      For example, if GAP-2-ARR-02 is blank then GST2.W,
**      GEND2.W, and GAP2.N are blanked out. This is because there
**      is no UMP or OLF data available if (a) there was
**      no End-gap or (b) there was actually a gap (as judged from
**      Ename reference dates) but the CAPI program missed it and
**      thus skipped questions GAP-1, GAP-2, and INTERCK.

NUMERIC  GAP_2.9 / GAP_2.10 / GAP_2.11      /* square roster for GAP-2-ARR
NUMERIC  INTCK.9 / INTCK.10 / INTCK.11      /* square roster for INTER-CK-ARR

*** some hand edits of data, only if dates are there and correct
IF (SERIAL=2244) R3887400 = 0      /* END1 = DOI, so Gap-length=0
IF (SERIAL=0382) R3887400 = 0      /* END1 > ST2, so no gap (Gap-length=0)
IF (SERIAL=0551) R3888600 = 38      /* the gap is not 17 weeks, it is 38
IF (SERIAL=2713) R3887400 = 0      /* END1 > ST2, so no gap (Gap-length=0)
IF (SERIAL=2713) R3888600 = 0      /* END3 = DOI, so no gap (Gap-length=0)
IF (SERIAL=3608) R3889200 = 0      /* END4 = DOI, so no gap (Gap-length=0)
IF (SERIAL=3120) R3889200 = 0      /* END4 = DOI, so no gap (Gap-length=0)

DO IF (WORKALL ne 1) & (CENS_GAP ge 3)
DO REPEAT
    GAP_1 = R3887200 R3887800 R3888400 R3889000 R3889600 R3890200
            R3890800 R3891400 R3892000 GAP_1.10 R3892600
    / GAP_2 = R3887300 R3887900 R3888500 R3889100 R3889700 R3890300
            R3890900 R3891500 GAP_2.9 GAP_2.10 GAP_2.11
    / Inter_ck = R3887400 R3888000 R3888600 R3889200 R3889800 R3890400
            R3891000 R3891600 INTCK.9 INTCK.10 INTCK.11
    / GAP_ST = GST1.W GST2.W GST3.W GST4.W GST5.W GST6.W GST7.W
            GST8.W GST9.W GST10.W GST11.W
    / GAP_END = GEND1.W GEND2.W GEND3.W GEND4.W GEND5.W GEND6.W
            GEND7.W GEND8.W GEND9.W GEND10.W GEND11.W
    / GAP_N = GAP1.N GAP2.N GAP3.N GAP4.N GAP5.N GAP6.N GAP7.N
            GAP8.N GAP9.N GAP10.N GAP11.N
    / GAPCENS = CGAP1 to CGAP11

IF SYSMIS(GAP_1)      GAP_ST = BLANK
IF SYSMIS(GAP_1)      GAP_END = BLANK

COMPUTE  GAP_N = (GAP_END - GAP_ST) + 1      /* # of weeks in END-gap

MISS VAL  GAP_2  INTER_CK(LO thru -1)
COMPUTE  GAPCENS = GAP_2 + INTER_CK      /* Total # weeks not working
MISS VAL  GAP_2  INTER_CK()

IF NVAL(GAP_ST) = 1 & SYSMIS(GAP_END)      GAP_N = -3
IF NVAL(GAP_ST) = 1 & SYSMIS(GAP_END)      GAP_END = -3

END REPEAT
END IF

*****      Hand edits      *****
***      After reviewing each case it was determined that
***      the following Gap-length values are correct.
IF (SERIAL = 0916) GAP4.N = 0      /* END4 = DOI
IF (SERIAL = 0018) GAP4.N = 0      /* END4 = DOI, gap length = 0
IF (SERIAL = 0367) GAP2.N = 0      /* R worked the whole time
IF (SERIAL = 3891) GAP1.N = 0      /* Job 1 subsumed by job 4, no End-gap1
IF (SERIAL = 3891) GAP6.N = 0      /* END6 = DOI (no gap)
IF (SERIAL = 1649) GAP2.N = -3      /* although probably = 0 if Start-Day=15

```

```

IF (SERIAL = 3120)   GAP1.N = -3      /* although probably = 0 if Start-Day=15
IF (SERIAL = 4156)   GAP3.N = 6
IF (SERIAL = 1302)   GAP3.N =11
IF (SERIAL = 3727)   GAP3.N = 8
IF (SERIAL = 3727)   GAP4.N = 3
IF (SERIAL = 1981)   GAP1.N =13
IF (SERIAL = 1981)   GAP4.N = 5
IF (SERIAL = 3019)   GAP1.N =13
IF (SERIAL = 3019)   GAP2.N =11
IF (SERIAL = 3789)   GAP1.N = 9
IF (SERIAL = 3789)   GAP3.N = 4
IF (SERIAL = 0551)   GAP4.N =10

```

```

*****
****          PART 4.B  -- END-GAP EXISTS          ****
****  Creates LOOKA, OLFA, and END-GAP length variables  ****
**
**  This recodes GAP-1, GAP-2, INTER-CK so they can be  **
**  used in LOOP and VECTOR commands to construct      **
**  the weeks variables.                               **
**
**  If  GAP-1 = 1 (all weeks looking):                  **
**      LOOKA = # wks in End-gap                        **
**      OLFA  = 0                                       **
**  If  GAP-1 = 2 (some weeks looking):                 **
**      LOOKA = GAP-2                                  **
**      OLFA  = INTER-CK                               **
**  If  GAP-1 = 3 (no weeks looking):                   **
**      LOOKA = 0                                       **
**      OLFA  = INTER-CK                               **
**=====

```

```

DO IF      (WORKALL ne 1)          /* and R didn't work the whole time
      & (HIT_1 = 1)                /* only if R hit GAP-1 array questions,

```

```

DO REPEAT
  GAP_1 = R3887200 R3887800 R3888400 R3889000 R3889600 R3890200
          R3890800 R3891400 R3892000 GAP_1.10 R3892600
  / GAP_2 = R3887300 R3887900 R3888500 R3889100 R3889700 R3890300
          R3890900 R3891500 GAP_2.9 GAP_2.10 GAP_2.11
  / Inter_ck = R3887400 R3888000 R3888600 R3889200 R3889800 R3890400
          R3891000 R3891600 INTCK.9 INTCK.10 INTCK.11
  / GAP_N = GAP1.N GAP2.N GAP3.N GAP4.N GAP5.N GAP6.N GAP7.N
          GAP8.N GAP9.N GAP10.N GAP11.N
  / GAPCENS = CGAP1 CGAP2 CGAP3 CGAP4 CGAP5 CGAP6 CGAP7 CGAP8
          CGAP9 CGAP10 CGAP11
  / LOOKA = LOOKA1 to LOOKA11
  / OLFA = OLFA1 to OLFA11

```

```

DO IF (GAP_1 = 1)          /* looked for all weeks during gap (all weeks UMP)
.COMPUTE LOOKA = GAP_N
.COMPUTE OLFA = 0
END IF

```

```

DO IF (GAP_1 = 2)          /* looked for work some of weeks in gap, mix of UMP & OLF
.COMPUTE LOOKA = GAP_2
.COMPUTE OLFA = INTER_CHK
END IF

```

```

DO IF (GAP_1 = 3)          /* didn't look at all during gap (all weeks OLF)
.COMPUTE LOOKA = 0
.COMPUTE OLFA = INTER_CHK
END IF

```

```

DO IF (GAP_1 < 0) /* Don't Know or Refuse to say # weeks looking
.COMPUTE LOOKA = -3
.COMPUTE OLFA = -3
END IF

DO IF (GAP_2 < 0) /* Don't Know or Refuse to say # weeks looking
.COMPUTE LOOKA = -3
.COMPUTE OLFA = -3
END IF

RECODE LOOKA OLFA (LO thru -1 = SYSMIS) /* If DK #wks looking
/* then LOOKA is missing

COMPUTE GAPCENS = LOOKA + OLFA /* Total # weeks not working
END REPEAT
END IF /* this ends the universe restrictor defined at the top

IF (SERIAL = 5091) OLFA6 = -3 /* GAP-2-ARR-03 = 28, but the gap is only 2 weeks
IF (SERIAL = 5091) LOOKA6 = -3 /* GAP-2-ARR-03 = 28, but the gap is only 2 weeks
IF (SERIAL = 0895) OLFA6 = 2 /* END6 < ST5 by 2 weeks, and R3890200 = 3

VAR LAB CGAP1 'Census END-gap1 length, in weeks'
/ CGAP2 'Census END-gap2 length, in weeks'

MISS VAL LOOKB OLFB (LO thru -1)

***** LOOP AND VECTOR *****
DO IF (WORKALL ne 1) /* and R didn't work the whole time
& (HIT_1 = 1) /* only if R hit GAP-1 array questions,

VECTOR WEEK = WEEK1 to WEEK950

DO REPEAT
St.WK = ST1.W ST2.W ST3.W ST4.W ST5.W ST6.W ST7.W ST8.W
ST9.W ST10.W ST11.W
/ End.WK = END1.W END2.W END3.W END4.W END5.W END6.W END7.W
END8.W END9.W END10.W END11.W
/ LOOKA = LOOKA1 LOOKA2 LOOKA3 LOOKA4 LOOKA5 LOOKA6 LOOKA7
LOOKA8 LOOKA9 LOOKA10 LOOKA11
/ OLFA = OLFA1 OLFA2 OLFA3 OLFA4 OLFA5 OLFA6 OLFA7
OLFA8 OLFA9 OLFA10 OLFA11
/ GAPST.W = GST1.W GST2.W GST3.W GST4.W GST5.W GST6.W GST7.W
GST8.W GST9.W GST10.W GST11.W
/ GEND.W = GEND1.W GEND2.W GEND3.W GEND4.W GEND5.W GEND6.W GEND7.W
GEND8.W GEND9.W GEND10.W GEND11.W
/ GAPA_N = GAP1.N GAP2.N GAP3.N GAP4.N GAP5.N GAP6.N GAP7.N
GAP8.N GAP9.N GAP10.N GAP11.N

** missing values wont affect WEEKS
MISS VAL LOOKA OLFA GAPA_N LOOKB OLFB (LO thru -1)

***** MISSING DATES
.DO IF SYSMIS(ST.WK) & NVAL(END.WK) = 1 /* Enames with missing START date
.COMPUTE CK_MISS = 1
.LOOP I# = 1 to END.WK
.IF SYSMIS(WEEK(I#)) WEEK(I#) = -3
.END LOOP
.END IF

.DO IF SYSMIS(END.WK) & NVAL(ST.WK) = 1 /* Enames with missing END date
.COMPUTE CK_MISS = 1
.LOOP I# = ST.WK to (WBID - 1)

```

```

      .IF SYSMIS(WEEK(I#))   WEEK(I#)=-3  /* from existing Start-week# to DOI week number
      .END LOOP
      .END IF

***** UNEMPLOYED:  Fill END-GAP wks lookng for work into WEEK array
LOOP I# = (GAPST.W) to (GAPST.W + LOOKA - 1)
COMPUTE   WEEK(I#) = 2
END LOOP

***** OLF:  Fill END-GAP wks not lookng for work (OLF) into WEEK array
LOOP I# = (GAPST.W + LOOKA) to (GAPST.W + LOOKA + OLFA - 1)
COMPUTE   WEEK(I#) = 3
END LOOP

*** WORKING:  Fill in the # of weeks worked, across all Enames
LOOP I# = ST.WK + 1 to (END.WK)      /* fill wks working for Ename 1 into WEEK array
COMPUTE   WEEK(I#) = 1
END LOOP
END REPEAT

***** DOLI GAP
MISS VAL  LOOKB  OLFB  (LO thru -1)
LOOP I# = 1 to LOOKB      /* fill wks looking during DOLIGAP into WEEK array
COMPUTE   WEEK(I#) = 2
END LOOP

LOOP I# = (LOOKB +1) to (LOOKB + OLFB)      /* fill wks not looking during DOLIGAP into
WEEK array
COMPUTE   WEEK(I#) = 3
END LOOP

COUNT MISSWEEK = WEEK1 to WEEK950(-3)
.DO IF MISSWEEK = 0
.COMPUTE WEMP3 = WORKW
.count WEMPx = WEEK1 to WEEK950(1)
.COUNT WUMP3 = WEEK1 to WEEK950(2)
.COUNT WOLF3 = WEEK1 to WEEK950(3)
.END IF

IF MISSWEEK > 0      WEMP3 = -3
IF MISSWEEK > 0      WUMP3 = -3
IF MISSWEEK > 0      WOLF3 = -3
END IF

**** HAND EDITS
*** Some of these have an End-gap that was missed
*** by CASES, so the weeks looking and OLF are undefined
DO IF ANY(SERIAL, 1118, 1302, 2984, 0551)
.COMPUTE WUMP3 = -3
.COMPUTE WOLF3 = -3
END IF

*****
***                               Final Measure of weeks working!!!!                               ***
***                               COMPILE 1, 2, 3 versions of WEMP, WUMP, WOLF                               ***
*****

DO IF NVAL(WEMP1, WUMP1, WOLF1) = 3
COMPUTE WORK = WEMP1
COMPUTE WUMP = WUMP1
COMPUTE WOLF = WOLF1
END IF

```

```
DO IF NVAL(WEMP2, WUMP2, WOLF2) = 3
  COMPUTE WORK = WEMP2
  COMPUTE WUMP = WUMP2
  COMPUTE WOLF = WOLF2
END IF
```

```
DO IF NVAL(WEMP3, WUMP3, WOLF3) = 3
  COMPUTE WORK = WEMP3
  COMPUTE WUMP = WUMP3
  COMPUTE WOLF = WOLF3
END IF
```

```
***** HAND-EDIT WOLF FOR 1 CASE:
IF (SERIAL = 3431) WOLF = 110
```

```
MISS VAL WORK WUMP WOLF ()
```

```
FINISH
```

```
/*1980 Industry Codes */
```

```
ARRAY X R3639800 R3660500 R3679900 R3699900 R3719200 R3737700 R3754200 R3770400
        R3786400 R3796700 R3804800 R3960800 R3972700 R3983900 R3994300 R4004300
        R4014400 R4023200 R4029900 R4035800 R4040400 R4045500;
ARRAY Y R3639900 R3660600 R3680000 R3700000 R3719300 R3737800 R3754300 R3770500
        R3786500 R3796800 R3804900 R3960900 R3972800 R3984000 R3994400 R4004400
        R4014500 R4023300 R4030000 R4035900 R4040500 R4045600;
```

```
Do over Y; IF X = 012 THEN Y =020; ELSE IF X = 020 THEN Y =021; ELSE
IF X = 030 THEN Y =020; ELSE IF X = 031 THEN Y =030; ELSE
IF X = 032 THEN Y =031; ELSE IF X = 450 THEN Y =460; ELSE
IF X = 451 THEN Y =461; ELSE IF X = 452 THEN Y =462; ELSE
IF X = 623 THEN Y =630; ELSE IF X = 630 THEN Y =631; ELSE
IF X = 631 THEN Y =632; ELSE IF X = 632 THEN Y =640; ELSE
IF X = 633 THEN Y =640; ELSE IF X = 640 THEN Y =682; ELSE
IF X = 661 THEN Y =682; ELSE IF X = 662 THEN Y =661; ELSE
IF X = 663 THEN Y =662; ELSE IF X = 891 THEN Y =730; ELSE
IF X = 732 THEN Y =740; ELSE IF X = 740 THEN Y =741; ELSE
IF X = 741 THEN Y =742; ELSE IF X = 742 THEN Y =742; ELSE
IF X = 801 THEN Y =682; ELSE IF X = 802 THEN Y =801; ELSE
IF X = 810 THEN Y =802; ELSE IF X = 863 THEN Y =892; ELSE
IF X = 873 THEN Y =892; ELSE IF X = 893 THEN Y =892; ELSE
IF X = 940 THEN Y =990; ELSE IF X = 941 THEN Y =990; ELSE
IF X = 942 THEN Y =990; ELSE IF X = 950 THEN Y =990; ELSE
IF X = 951 THEN Y =990; ELSE IF X = 952 THEN Y =990; ELSE
IF X = 960 THEN Y =990; ELSE IF X = 991 THEN Y =990; ELSE
IF X = 992 THEN Y =990; ELSE Y =X;
END;
```

```
/*1980 Occupation Codes */
```

```
ARRAY XX R3640200 R3660900 R3680300 R3700300 R3719600 R3738100 R3754600 R3770700
        R3786700 R3796900 R3805000 R3961300 R3973400 R3984700 R3995100 R4005100
        R4015200 R4024000 R4030400 R4036300 R4040900 R4046000;
ARRAY YY R3640300 R3661000 R3680400 R3700400 R3719700 R3738200 R3754700 R3770800
        R3786800 R3797000 R3805100 R3961400 R3973500 R3984800 R3995200 R4005200
        R4015300 R4024100 R4030500 R4036400 R4041000 R4046100;
```

```
DO OVER YY; IF XX = 016 THEN YY =017; ELSE IF XX = 017 THEN YY =019; ELSE
IF XX = 018 THEN YY =016; ELSE IF XX = 019 THEN YY =018; ELSE
IF XX = 021 THEN YY =019; ELSE IF XX = 022 THEN YY =019; ELSE
IF XX = 353 THEN YY =349; ELSE IF XX = 368 THEN YY =369; ELSE
IF XX = 436 THEN YY =437; ELSE IF XX = 461 THEN YY =463; ELSE
IF XX = 462 THEN YY =464; ELSE IF XX = 463 THEN YY =465; ELSE
IF XX = 464 THEN YY =466; ELSE IF XX = 465 THEN YY =467; ELSE
IF XX = 466 THEN YY =468; ELSE IF XX = 467 THEN YY =468; ELSE
IF XX = 628 THEN YY =633; ELSE IF XX = 674 THEN YY =673; ELSE
IF XX = 795 THEN YY =794; ELSE IF XX = 804 THEN YY =805; ELSE
IF XX = 864 THEN YY =863; ELSE IF XX = 865 THEN YY =864; ELSE
IF XX = 866 THEN YY =865; ELSE IF XX = 867 THEN YY =866; ELSE
IF XX = 868 THEN YY =867; ELSE IF XX = 874 THEN YY =873; ELSE
IF XX = 903 THEN YY = -4; ELSE IF XX = 904 THEN YY = -4; ELSE
IF XX = 905 THEN YY = -4; ELSE IF XX = 909 THEN YY = -4; ELSE YY = XX;
END;
```



```

/*Net Family Assets, Net Family Income, and Summation
NOTE. Amounts have been top-coded at Census */
C97=0; IF R4146300=0 & R4146400<0 & R4146500<0 THEN HOUSE97=0;
IF R4146500>=0 & R4147000>=0 THEN PROPDE97 = R4146500 + R4147000;
IF R4146500=-4 & R4147000>=0 THEN PROPDE97 = R4147000;
IF R4146500>=0 & R4147000=-4 THEN PROPDE97 = R4146500;
IF R4146500<0 | R4147000<0 THEN PROPDE97 = . ;
ELSE IF R4146400>=0 & PROPDE97>=0 THEN HOUSE97=R4146400-PROPDE97;
IF R4151200=1 & R4146400<=R4151100 & R4146500<=R4151300
& R4151100^=-4 & R4151300 ^=-4 THEN HOUSE97=0;
IF HOUSE97^=. THEN ASSET97=HOUSE97; ELSE C97=C97+1;
IF (R4147300=0 | R4161300=0) & R4147400<0 THEN SAVE97=0;
ELSE IF R4147400>=0 THEN SAVE97=R4147400;
IF R4147500=1 & R4147600=1 THEN SAVE97=40000;
IF R4147500=1 & R4147600=0 THEN SAVE97=25000;
IF R4147500=0 & R4147700=1 THEN SAVE97=5500;
IF R4147500=0 & R4147700=0 THEN SAVE97=1000;
IF R4147500=-1 |R4147600=-1 |R4147700=-1 |R4147500=-2 |
R4147600=-2|R4147700=-2 THEN SAVE97=. ;
IF SAVE97^=. THEN ASSET97=ASSET97+SAVE97; ELSE C97=C97+1;
IF (R4147800=0 | R4161300=0) & R4147900<0 THEN BOND97=0;
ELSE IF R4147900>=0 THEN BOND97=R4147900;
IF R4148000=1 & R4148100=1 THEN BOND97=5000;
IF R4148000=1 & R4148100=0 THEN BOND97=3000;
IF R4148000=0 & R4148200=1 THEN BOND97=750;
IF R4148000=0 & R4148200=0 THEN BOND97=500;
IF R4148000=-1 |R4148100=-1 |R4148200=-1 |R4148000=-2 |
R4148100=-2 |R4148200=-2 THEN BOND97=. ;
IF BOND97 ^=. THEN ASSET97=ASSET97+BOND97; ELSE C97=C97+1;
IF (R4148300=0 | R4161300=0) & R4148400<0 THEN STOCK97=0;
ELSE IF R4148400>=0 THEN STOCK97=R4148400;
IF R4148500=1 & R4148600=1 THEN STOCK97=40000;
IF R4148500=1 & R4148600=0 THEN STOCK97=27500;
IF R4148500=0 & R4148700=1 THEN STOCK97=10000;
IF R4148500=0 & R4148700=0 THEN STOCK97=5000;
IF R4148500=-1 |R4148600=-1 |R4148700=-1 |R4148500=-2 |
R4148600=-2 |R4148700=-2 THEN STOCK97=. ;
IF STOCK97^=. THEN ASSET97=ASSET97+STOCK97; ELSE C97=C97+1;
IF R4148800=0 & R4148900<0 THEN IRA97=0; ELSE IF R4148900 GE 0 THEN IRA97=R4148900;
IF R4149000=1 & R4149100=1 THEN IRA97=30000;
IF R4149000=1 & R4149100=0 THEN IRA97=22500;
IF R4149000=0 & R4149200=1 THEN IRA97=10000;
IF R4149000=0 & R4149200=0 THEN IRA97=5000;
IF R4149000=-1 |R4149100=-1 |R4149200=-1 |R4149000=-2 |
R4149100=-2 |R4149200=-2 THEN IRA97 = . ;
IF IRA97^=. THEN ASSET97=ASSET97+IRA97; ELSE C97=C97+1;
IF R4149300=0 & R4149400<0 THEN LOAN97=0; ELSE IF R4149400>=0 THEN LOAN97=R4149400;
IF LOAN97^=. THEN ASSET97=ASSET97+LOAN97; ELSE C97=C97+1;
IF (R4149500=0 | R4149700=1) & R4149800<0 THEN INSU97=0;
ELSE IF R4149800 GE 0 THEN INSU97=R4149800;
IF INSU97^=. THEN ASSET97=ASSET97+INSU97; ELSE C97=C97+1;
IF R4150100=0 & R4150200<0 THEN TRUS97=0;
ELSE IF R4150200 >= 0 THEN TRUS97 = R4150200;
IF TRUS97^=. THEN ASSET97=ASSET97+TRUS97; ELSE C97=C97+1;
IF R4150600=0 & (R4150700<0 | R4150700=-4) THEN SETTLE=0;
ELSE IF R4150600=1 & R4150700<0 THEN SETTLE=0;
ELSE IF R4150700>=0 THEN SETTLE=R4150700;
IF SETTLE ^=. THEN ASSET97=ASSET97+SETTLE; ELSE C97=C97+1;
IF (R4151000=0 | R4160900=0) & R4151100<0 & R4151300<0 THEN IFARM=0;
ELSE IF R4151100>=0 & R4151300>=0 THEN IFARM=R4151100-R4151300;
IF IFARM ^=. THEN ASSET97=ASSET97+IFARM; ELSE C97=C97+1;
IF (R4151400=0 | (R4153600=0 & R4157400=0)) & R4151500<0 & R4151600<0 THEN IBUS=0;
ELSE IF R4151500>=0 & R4151600>=0 THEN IBUS=R4151500-R4151600;
IF IBUS ^=. THEN ASSET97=ASSET97+IBUS; ELSE C97=C97+1;

```

```

IF (R4151700=0 | R4161100=0) & R4151800<0 & R4151900<0 & R4152000<0 THEN IREAL=0;
ELSE IF R4151800 >=0 & R4151900>=0 & R4152000>=0
THEN IREAL=R4151800 -R4151900-R4152000;
IF IREAL ^=. THEN ASSET97=ASSET97+IREAL; ELSE C97=C97+1;
IF R4152100=0 & R4152200<0 THEN DEBT=0; ELSE IF R4152200>=0 THEN DEBT=R4152200;
IF DEBT ^=. THEN ASSET97=ASSET97-DEBT; ELSE C97=C97+1;
IF C97=0 THEN DO; IF ASSET97<-99999 THEN R4171300=-99999;
IF ASSET97>999999 THEN R4171300=999999;
IF -99999 LE ASSET97 LE 999999 THEN R4171300=ASSET97; END;

IF R4153500<0 THEN RWAGE=0; IF R4153500>=0 THEN RWAGE=R4153500;
IF (R4153600=0 | R4153600=4) & (R4153700=-1 | R4153700=-2 | R4153700=-4) THEN RBUS=0;
IF R4153600=1 & R4153700>=0 THEN RBUS=R4153700;
IF R4153600=3 & R4153700>=0 THEN RBUS=0-R4153700;
IF R4153800=0 & R4153900<0 & R4154000<0 THEN RUNEM=0;
IF R4153900>=0 & R4154000>=0 THEN RUNEM=R4153900*R4154000;
IF (RUNEM=0 | R4154100=0) & R4154200<0 & R4154300<0 THEN RSUB=0;
IF R4154200>=0 & R4154300>=0 THEN RSUB=R4154200*R4154300;
IF R4154400=0 & R4154500<0 & R4154600<0 THEN RSOC=0;
IF R4154500>=0 & R4154600>=0 THEN RSOC=R4154500*R4154600;
IF R4154800=0 & R4154900<0 THEN RVET=0; IF R4154900>=0 THEN RVET=R4154900;
IF R4155000=0 & R4155100<0 THEN RCOM=0; IF R4155100>=0 THEN RCOM=R4155100;
IF R4155200=0 & R4155300<0 THEN RSSD=0; IF R4155300>=0 THEN RSSD=R4155300;
IF R4155400=0 & R4155500<0 THEN RDIS=0; IF R4155500>=0 THEN RDIS=R4155500;
IF R4155600=0 THEN DO;
REPRI=0; REMIL=0; REFED=0; RESTE=0; REUNI=0; REIRA=0; REOTH=0; END;
IF R4155700=0 & R4155800<0 THEN REPRI=0; IF R4155800>=0 THEN REPRI=R4155800;
IF R4155900=0 & R4156000<0 THEN REMIL=0; IF R4156000>=0 THEN REMIL=R4156000;
IF R4156100=0 & R4156200<0 THEN REFED=0; IF R4156200>=0 THEN REFED=R4156200;
IF R4156300=0 & R4156400<0 THEN RESTE=0; IF R4156400>=0 THEN RESTE=R4156400;
IF R4156500=0 & R4156600<0 THEN REUNI=0; IF R4156600>=0 THEN REUNI=R4156600;
IF R4156700=0 & R4156800<0 THEN REIRA=0; IF R4156800>=0 THEN REIRA=R4156800;
IF R4156900=0 & R4157000<0 THEN REOTH=0; IF R4157000>=0 THEN REOTH=R4157000;
IF (R4157100^=1 | R4157200=0) & R4157300<0 THEN SWAGE=0;
IF R4157300>=0 THEN SWAGE=R4157300;
IF (R4157100^=1 | R4157400=0 | R4157400=4) & R4157500<0 THEN SBUS=0;
IF R4157400=1 & R4157500>=0 THEN SBUS=R4157500;
IF R4157400=3 & R4157500>=0 THEN SBUS=0-R4157500;
IF (R4157100^=1 | R4157600=0) & R4157700<0 & R4157800<0 THEN SUNEM=0;
IF R4157700>=0 & R4157800>=0 THEN SUNEM=R4157700*R4157800;
IF (R4157100^=1 | R4157900=0 | SUNEM=0) & R4158000<0 & R4158100<0 THEN SSUB=0;
IF R4158000>=0 & R4158100>=0 THEN SSUB=R4158000*R4158100;
IF (R4157100^=1 | R4158200=0) & R4158300<0 & R4158400<0 THEN SSOC=0;
IF R4158300>=0 & R4158400>=0 THEN SSOC=R4158300*R4158400;
IF (R4157100^=1 | R4158600=0) & R4158700<0 THEN SVET=0;
IF R4158700>=0 THEN SVET=R4158700;
IF (R4157100^=1 | R4158800=0) & R4158900<0 THEN SCOM=0;
IF R4158900>=0 THEN SCOM=R4158900;
IF (R4157100^=1 | R4159000=0) & R4159100<0 THEN SSSD=0;
IF R4159100>=0 THEN SSSD=R4159100;
IF (R4157100^=1 | R4159200=0) & R4159300<0 THEN SDIS=0;
IF R4159300>=0 THEN SDIS=R4159300;
IF (R4157100^=1 | R4159400=0) THEN DO;
SREPRI=0; SREMIL=0; SREFED=0; SRESTE=0; SREUNI=0; SREIRA=0; SREOTH=0; END;
IF R4159500=0 & R4159600<0 THEN SREPRI=0; IF R4159600>=0 THEN SREPRI=R4159600;
IF R4159700=0 & R4159800<0 THEN SREMIL=0; IF R4159800>=0 THEN SREMIL=R4159800;
IF R4159900=0 & R4160000<0 THEN SREFED=0; IF R4160000>=0 THEN SREFED=R4160000;
IF R4160100=0 & R4160200<0 THEN SRESTE=0; IF R4160200>=0 THEN SRESTE=R4160200;
IF R4160300=0 & R4160400<0 THEN SREUNI=0; IF R4160400>=0 THEN SREUNI=R4160400;
IF R4160500=0 & R4160600<0 THEN SREIRA=0; IF R4160600>=0 THEN SREIRA=R4160600;
IF R4160700=0 & R4160800<0 THEN SREOTH=0; IF R4160800>=0 THEN SREOTH=R4160800;
IF (R4160900=0 | R4160900=4) & (R4161000=-4 | R4161000=-1 | R4161000=-2) THEN FARM=0;
IF R4160900=1 & R4161000>=0 THEN FARM=R4161000;
IF R4160900=3 & R4161000>=0 THEN FARM=0-R4161000;

```

```

IF (R4161100=0 | R4161200=4) & (R4161200=-4 | R4161200=-1 | R4161200=-2) THEN RENT=0;
IF R4161100=1 & R4161200>=0 THEN RENT=R4161200;
IF R4161100=3 & R4161200>=0 THEN RENT=0-R4161200;
IF R4161300=0 & R4161400<0 THEN INTT=0; IF R4161400>=0 THEN INTT=R4161400;
IF R4161500=0 & R4161600<0 & R4161700<0 THEN FOODS=0;
IF R4161600>=0 & R4161700>=0 THEN FOODS=R4161600*R4161700;
IF R4161800=0 & R4161900<0 & R4162000<0 THEN AFDC=0;
IF R4161900>=0 & R4162000>=0 THEN AFDC=R4161900*R4162000;
IF R4162100=0 & R4162200<0 & R4162300<0 THEN SSI=0;
IF R4162200>=0 & R4162300>=0 THEN SSI=R4162200*R4162300;
IF (R4162400=0 & R4162600=0) & R4162500<0 & R4162700<0 THEN ALI=0;
IF R4162500>=0 & R4162700>=0 THEN ALI=R4162500-R4162700;
IF R4162500>=0 & R4162700=-4 THEN ALI=R4162500;
IF R4162500=-4 & R4162700>=0 THEN ALI=R4162700;
IF R4162900=0 & R4163000<0 THEN CHDSUP=0; ELSE IF R4163000>=0 THEN CHDSUP=R4163000;
  ELSE IF R4163700=2 & R4166200>=0 THEN CHDSUP=R4166200;
  ELSE IF R4163700=1 & R4167000>=0 THEN CHDSUP=R4167000;
  ELSE IF R4163700=3 & R4166200>=0 & R4167000>=0 THEN CHDSUP=R4166200+R4167000;
IF R4167400=0 & (R4167500<0 & R4167600<0 & R4167700<0) THEN RCHD=0;
  ELSE IF R4167500>=0 THEN RCHD=52*R4167500; ELSE IF R4167600>=0 THEN RCHD=12*R4167600;
  ELSE IF R4167700>=0 THEN RCHD=R4167700;
IF (R4157100^=1 | R4168000=0) & (R4168100<0 & R4168200<0 & R4168300<0) THEN HPCHD=0;
  ELSE IF R4168100>=0 THEN HPCHD=52*R4168100;
  ELSE IF R4168200>=0 THEN HPCHD=12*R4168200;
  ELSE IF R4168300>=0 THEN HPCHD=R4168300;
IF CHDSUP>=0 & RCHD>=0 & HPCHD>=0 THEN CHILD=CHDSUP-RCHD-HPCHD;
IF R4168500=0 & R4168600<0 THEN OTHER=0; IF R4168600>=0 THEN OTHER=R4168600;
IF R4168800 NE 2 & R4168900<0 THEN FAM=0; ELSE IF R4168800=2 THEN DO;
  IF R4168900=1 THEN FAM= 2000; ELSE IF R4168900=2 THEN FAM= 5000;
  ELSE IF R4168900=3 THEN FAM= 6750; ELSE IF R4168900=4 THEN FAM= 8250;
  ELSE IF R4168900=5 THEN FAM= 12500; ELSE IF R4168900=6 THEN FAM= 16250;
  ELSE IF R4168900=7 THEN FAM= 18750; ELSE IF R4168900=8 THEN FAM= 22500;
  ELSE IF R4168900=9 THEN FAM= 30000; ELSE IF R4168900=10 THEN FAM= 42500;
  ELSE IF R4168900=11 THEN FAM= 62500; ELSE IF R4168900=12 THEN FAM= 87500;
  ELSE IF R4168900=13 THEN FAM=100000; ELSE IF (R4168900=0 | R4168900=14) THEN FAM=0;
END;
IF RWAGE^=. &RBUS^=. &RUNEM^=. &RSUB^=. &RSOC^=. &RVET^=. &RCOM^=. &RSSD^=. &RDIS^=.
&REPRI^=. &REMIL^=. &REFED^=. &RESTE^=. &REUNI^=. &REIRA^=. &REOTH^=. &SWAGE^=.
&SBUS^=. &SUNEM^=. &SSUB^=. &SSOC^=. &SVET^=. &SCOM^=. &SSSD^=. &SDIS^=. &SREPRI^=.
&SREMIL^=. &SREFED^=. &SRESTE^=. &SREUNI^=. &SREIRA^=. &SREOTH^=. &FARM^=. &RENT^=.
&INTT^=. &FOODS^=. &AFDC^=. &SSI^=. &ALI^=. &CHILD^=. &OTHER^=. &FAM^=. THEN
R4171400=RWAGE+RBUS+RUNEM+RSUB+RSOC+RVET+RCOM+RSSD+RDIS+REPRI+REMIL+REFED+RESTE+
REUNI+REIRA+REOTH+SWAGE+SBUS+SUNEM+SSUB+SSOC+SVET+SCOM+SSSD+SDIS+SREPRI+SREMIL+
SREFED+SRESTE+SREUNI+SREIRA+SREOTH+FARM+RENT+INTT+FOODS+AFDC+SSI+ALI+CHILD+OTHER+FAM;
IF R4171400^=. & R4171400<=-99999 THEN R4171400=-99999;
IF R4171400>=999999 THEN R4171400=999999;

NI=0;
Array ALLINC RWAGE RBUS RUNEM RSUB RSOC RVET RCOM RSSD RDIS REPRI REMIL REFED RESTE
REUNI REIRA REOTH SWAGE SBUS SUNEM SSUB SSOC SVET SCOM SSSD SDIS SREPRI SREMIL SREFED
SRESTE SREUNI SREIRA SREOTH FARM RENT INTT FOODS AFDC SSI ALI CHILD OTHER FAM;
DO OVER ALLINC; IF ALLINC GE 0 THEN do; NI=NI+1;
if NI=1 then R4171500=allinc if NI>1 then R4171500=R4171500+ALLINC; END; end;
IF R4171500=. & R4153300>=1 THEN DO;
IF R4153300=1 THEN R4171500=2000; ELSE IF R4153300=2 THEN R4171500=5000;
ELSE IF R4153300=3 THEN R4171500=6750; ELSE IF R4153300=4 THEN R4171500=8250;
ELSE IF R4153300=5 THEN R4171500=12500; ELSE IF R4153300=6 THEN R4171500=16250;
ELSE IF R4153300=7 THEN R4171500=18750; ELSE IF R4153300=8 THEN R4171500=22500;
ELSE IF R4153300=9 THEN R4171500=30000; ELSE IF R4153300=10 THEN R4171500=42500;
ELSE IF R4153300=11 THEN R4171500=62500; ELSE IF R4153300=12 THEN R4171500=87500;
ELSE IF R4153300=13 THEN R4171500=100000; ELSE IF R4153300=14 THEN R4171500=0; END;
IF R4171500^=. & R4171500<-99999 THEN R4171500=-99999; IF R4171500>999999 THEN
R4171500=999999;

```

/*The highest grade completed for 1997 is the highest grade completed in the previous survey year updated with additional schooling reported in 1997.*/

```

if R0079000 = -127 then R0079000 = .;
if R0079000 = -128 then R0079000 = .;
if R0381500 = -127 then R0381500 = .;
if R0381500 = -128 then R0381500 = .;
if R0989700 = -127 then R0989700 = .;
if R0989700 = -128 then R0989700 = .;
if R3476600 = -4 then R3476600 = .;
if R3476600 = -5 then R3476600 = .;
HGC=R3476600;
IF R3499100=1993 THEN HGC=R1520410;
IF R3499100=1991 THEN HGC=R1346410;
IF R3499100=1988 THEN HGC=R1215110;
IF R3499100=1987 THEN HGC=R1097410;
IF R3499100=1985 THEN HGC=R1051610;
IF R3499100=1983 THEN HGC=R0929510;
IF R3499100=1982 THEN HGC=R0797110;
IF R3499100=1980 THEN HGC=R0749910;
IF COHORT='MW' & (R3499100=1992 OR R3499100=1989) THEN HGC=R0989700;
IF COHORT='MW' & (1979 LE R3499100 LE 1987) THEN HGC=R0381500;
IF COHORT='MW' & HGC=. THEN HGC=R0079000;

```

```

IF HGC=-7 THEN HGC=95;
IF HGC=-6 THEN HGC=12;
IF HGC=-5 THEN HGC=94;
IF HGC=-4 THEN HGC=93;
IF HGC=-3 THEN HGC=98;
IF HGC=-2 THEN HGC=97;
IF HGC=-1 THEN HGC=96;

```

```

IF 1 LE R4193300 LE 10 THEN R4192800=R4193300+7;
IF R4193300=11 THEN R4192800=95;
IF 1 LE R4193500 LE 3 THEN R4192800=12;
IF 3 LE R4193800 LE 4 THEN R4192800=18;
IF R4193800=2 THEN R4192800=16;
IF R4193800=1 THEN R4192800=14;
IF HGC>R4192800 THEN R4192800=HGC;

```

/*PENSION DATA FOR 1997 AS COMPARED TO 1995

In 1995 pension questions are part of the Employer roster: RSP-108 (HSP-108) through CK-P1-I (CK-HP1-I) for the first (husband's) pension; RSP-208 (HSP-208) through CK-P2-I (CK-HP2-I) for the second, RSP-308 (HSP-308) through CK-P3-I (CK-HP3-I) for the third, and RSP-408 (HSP-408) through CK-P4-I (CK-HP4-I) for the fourth. So if Respondent X's current employer is on Line 3, and she is participating in two pensions from that employer, pension data for 1995 will be present in the 108 and 208 series (pensions one and two) at -ARR-03 (Employer 3).

In 1997, pension data comprise a separate roster within the Employer roster. In addition, questions have been added (CK-PEN-2, RSP-102A,B,C) that skip a respondent around information collected in 1995 if the pension for her current job has not changed. Questions have also been added (CK-PEN, CK-PEN3, CK-PEN4, RSP-146A to RSP-146H) that ask about pensions from employers that are not current but were active since the last interview.

To help make the transition from one roster to another, R3PENS on the Employer roster shows the line number of the pension roster where pensions for that Employer begin. R7PENS on the pension roster shows the count of a pension for a particular employer, and R7EMPS on the pension roster shows the line number of the Employer that is associated with a particular pension. Suppose for example, that Respondent X's current employer is on Line 6, that she is participating in two pensions from that employer, and that she has a non-current employer on Line 2. R3PENS-ROST02 (Employer 2) shows "1" (for pensions to begin at Line 1 of the respondent pension roster), R7PENS-ROST1

shows "1" (for the first pension of that employer), R7EMPS-ROST1 shows "2" (for Employer 2). R3PENS-ROST06 (Employer 6) shows "2" (for pensions to begin at Line 2 of the pension roster), R7PENS-ROST2 shows "1" (for the first pension of that employer), R7EMPS-ROST2 shows "6" (for Employer 6), R7PENS-ROST3 shows "2" (for the second pension of that employer), and R7EMPS-ROST3 shows "6" (for Employer 6). Documentation variables for the husband/partner are R5PENS (on the H/P Employer roster), and R8PENS and R8EMPS (on the H/P pension roster).

Because gaps in employment follow the same roster structure as do pensions, variables that document transitions between rosters are analogous. R7PENS, R7EMPS, R8PENS, and R8EMPS are created at the Census Bureau, along with R4EGAP, R4EMPS, R6EGAP, and R6EMPS. Following is the code for R3PENS, R5PENS, R3EGAP and R5EGAP. */

```

IF R3847500=1 THEN DO; IF R3847600=1 THEN R3655700=1;
  ELSE IF R3847600=2 THEN R3675200=1; ELSE IF R3847600=3 THEN R3695200=1;
  ELSE IF R3847600=4 THEN R3714500=1; ELSE IF R3847600=5 THEN R3733100=1;
  ELSE IF R3847600=6 THEN R3749600=1; ELSE IF R3847600=7 THEN R3765800=1;
  ELSE IF R3847600=8 THEN R3781900=1; ELSE IF R3847600=9 THEN R3793300=1;
  ELSE IF R3847600=10 THEN R3801300=1; ELSE IF R3847600=11 THEN R3809400=1; END;
IF R3859000=1 THEN DO; IF R3859100=1 THEN R3655700=2;
  ELSE IF R3859100=2 THEN R3675200=2; ELSE IF R3859100=3 THEN R3695200=2;
  ELSE IF R3859100=4 THEN R3714500=2; ELSE IF R3859100=5 THEN R3733100=2;
  ELSE IF R3859100=6 THEN R3749600=2; ELSE IF R3859100=7 THEN R3765800=2;
  ELSE IF R3859100=8 THEN R3781900=2; ELSE IF R3859100=9 THEN R3793300=2;
  ELSE IF R3859100=10 THEN R3801300=2; ELSE IF R3859100=11 THEN R3809400=2; END;
IF R3868000=1 THEN DO; IF R3868100=1 THEN R3655700=3;
  ELSE IF R3868100=2 THEN R3675200=3; ELSE IF R3868100=3 THEN R3695200=3;
  ELSE IF R3868100=4 THEN R3714500=3; ELSE IF R3868100=5 THEN R3733100=3;
  ELSE IF R3868100=6 THEN R3749600=3; ELSE IF R3868100=7 THEN R3765800=3;
  ELSE IF R3868100=8 THEN R3781900=3; ELSE IF R3868100=9 THEN R3793300=3;
  ELSE IF R3868100=10 THEN R3801300=3; ELSE IF R3868100=11 THEN R3809400=3; END;
IF R3876000=1 THEN DO; IF R3876100=1 THEN R3655700=4;
  ELSE IF R3876100=2 THEN R3675200=4; ELSE IF R3876100=3 THEN R3695200=4;
  ELSE IF R3876100=4 THEN R3714500=4; ELSE IF R3876100=5 THEN R3733100=4;
  ELSE IF R3876100=6 THEN R3749600=4; ELSE IF R3876100=7 THEN R3765800=4;
  ELSE IF R3876100=8 THEN R3781900=4; ELSE IF R3876100=9 THEN R3793300=4;
  ELSE IF R3876100=10 THEN R3801300=4; ELSE IF R3876100=11 THEN R3809400=4; END;
IF R3881900=1 THEN DO; IF R3882000=1 THEN R3655700=5;
  ELSE IF R3882000=2 THEN R3675200=5; ELSE IF R3882000=3 THEN R3695200=5;
  ELSE IF R3882000=4 THEN R3714500=5; ELSE IF R3882000=5 THEN R3733100=5;
  ELSE IF R3882000=6 THEN R3749600=5; ELSE IF R3882000=7 THEN R3765800=5;
  ELSE IF R3882000=8 THEN R3781900=5; ELSE IF R3882000=9 THEN R3793300=5;
  ELSE IF R3882000=10 THEN R3801300=5; ELSE IF R3882000=11 THEN R3809400=5; END;
IF R3885200=1 THEN DO; IF R3885300=1 THEN R3655700=6;
  ELSE IF R3885300=2 THEN R3675200=6; ELSE IF R3885300=3 THEN R3695200=6;
  ELSE IF R3885300=4 THEN R3714500=6; ELSE IF R3885300=5 THEN R3733100=6;
  ELSE IF R3885300=6 THEN R3749600=6; ELSE IF R3885300=7 THEN R3765800=6;
  ELSE IF R3885300=8 THEN R3781900=6; ELSE IF R3885300=9 THEN R3793300=6;
  ELSE IF R3885300=10 THEN R3801300=6; ELSE IF R3885300=11 THEN R3809400=6; END;
IF R3885800=1 THEN DO; IF R3885900=1 THEN R3655700=7;
  ELSE IF R3885900=2 THEN R3675200=7; ELSE IF R3885900=3 THEN R3695200=7;
  ELSE IF R3885900=4 THEN R3714500=7; ELSE IF R3885900=5 THEN R3733100=7;
  ELSE IF R3885900=6 THEN R3749600=7; ELSE IF R3885900=7 THEN R3765800=7;
  ELSE IF R3885900=8 THEN R3781900=7; ELSE IF R3885900=9 THEN R3793300=7;
  ELSE IF R3885900=10 THEN R3801300=7; ELSE IF R3885900=11 THEN R3809400=7; END;
IF R3886500=1 THEN DO; IF R3886600=1 THEN R3655700=8;
  ELSE IF R3886600=2 THEN R3675200=8; ELSE IF R3886600=3 THEN R3695200=8;
  ELSE IF R3886600=4 THEN R3714500=8; ELSE IF R3886600=5 THEN R3733100=8;
  ELSE IF R3886600=6 THEN R3749600=8; ELSE IF R3886600=7 THEN R3765800=8;
  ELSE IF R3886600=8 THEN R3781900=8; ELSE IF R3886600=9 THEN R3793300=8;
  ELSE IF R3886600=10 THEN R3801300=8; ELSE IF R3886600=11 THEN R3809400=8; END;

```

```

IF R4088700=1 THEN DO; IF R4088800=1 THEN R3968500=1;
  ELSE IF R4088800=2 THEN R3979800=1; ELSE IF R4088800=3 THEN R3990200=1;
  ELSE IF R4088800=4 THEN R4000300=1; ELSE IF R4088800=5 THEN R4010400=1;
  ELSE IF R4088800=6 THEN R4019200=1; ELSE IF R4088800=7 THEN R4026900=1;
  ELSE IF R4088800=8 THEN R4033000=1; ELSE IF R4088800=9 THEN R4038000=1;
  ELSE IF R4088800=10 THEN R4041500=1; ELSE IF R4088800=12 THEN R4044000=1;
  ELSE IF R4088800=13 THEN R4047300=1; END;
IF R4100100=1 THEN DO; IF R4100200=1 THEN R3968500=2;
  ELSE IF R4100200=2 THEN R3979800=2; ELSE IF R4100200=3 THEN R3990200=2;
  ELSE IF R4100200=4 THEN R4000300=2; ELSE IF R4100200=5 THEN R4010400=2;
  ELSE IF R4100200=6 THEN R4019200=2; ELSE IF R4100200=7 THEN R4026900=2;
  ELSE IF R4100200=8 THEN R4033000=2; ELSE IF R4100200=9 THEN R4038000=2;
  ELSE IF R4100200=10 THEN R4041500=2; ELSE IF R4100200=12 THEN R4044000=2;
  ELSE IF R4100200=13 THEN R4047300=2; END;
IF R4108000=1 THEN DO; IF R4108100=1 THEN R3968500=3;
  ELSE IF R4108100=2 THEN R3979800=3; ELSE IF R4108100=3 THEN R3990200=3;
  ELSE IF R4108100=4 THEN R4000300=3; ELSE IF R4108100=5 THEN R4010400=3;
  ELSE IF R4108100=6 THEN R4019200=3; ELSE IF R4108100=7 THEN R4026900=3;
  ELSE IF R4108100=8 THEN R4033000=3; ELSE IF R4108100=9 THEN R4038000=3;
  ELSE IF R4108100=10 THEN R4041500=3; ELSE IF R4108100=12 THEN R4044000=3;
  ELSE IF R4108100=13 THEN R4047300=3; END;
IF R4114200=1 THEN DO; IF R4114300=1 THEN R3968500=4;
  ELSE IF R4114300=2 THEN R3979800=4; ELSE IF R4114300=3 THEN R3990200=4;
  ELSE IF R4114300=4 THEN R4000300=4; ELSE IF R4114300=5 THEN R4010400=4;
  ELSE IF R4114300=6 THEN R4019200=4; ELSE IF R4114300=7 THEN R4026900=4;
  ELSE IF R4114300=8 THEN R4033000=4; ELSE IF R4114300=9 THEN R4038000=4;
  ELSE IF R4114300=10 THEN R4041500=4; ELSE IF R4114300=12 THEN R4044000=4;
  ELSE IF R4114300=13 THEN R4047300=4; END;
IF R4117900=1 THEN DO; IF R4118000=1 THEN R3968500=5;
  ELSE IF R4118000=2 THEN R3979800=5; ELSE IF R4118000=3 THEN R3990200=5;
  ELSE IF R4118000=4 THEN R4000300=5; ELSE IF R4118000=5 THEN R4010400=5;
  ELSE IF R4118000=6 THEN R4019200=5; ELSE IF R4118000=7 THEN R4026900=5;
  ELSE IF R4118000=8 THEN R4033000=5; ELSE IF R4118000=9 THEN R4038000=5;
  ELSE IF R4118000=10 THEN R4041500=5; ELSE IF R4118000=12 THEN R4044000=5;
  ELSE IF R4118000=13 THEN R4047300=5; END;
IF R4118900=1 THEN DO; IF R4119000=1 THEN R3968500=6;
  ELSE IF R4119000=2 THEN R3979800=6; ELSE IF R4119000=3 THEN R3990200=6;
  ELSE IF R4119000=4 THEN R4000300=6; ELSE IF R4119000=5 THEN R4010400=6;
  ELSE IF R4119000=6 THEN R4019200=6; ELSE IF R4119000=7 THEN R4026900=6;
  ELSE IF R4119000=8 THEN R4033000=6; ELSE IF R4119000=9 THEN R4038000=6;
  ELSE IF R4119000=10 THEN R4041500=6; ELSE IF R4119000=12 THEN R4044000=6;
  ELSE IF R4119000=13 THEN R4047300=6; END;
IF R4119900=1 THEN DO; IF R4120000=1 THEN R3968500=7;
  ELSE IF R4120000=2 THEN R3979800=7; ELSE IF R4120000=3 THEN R3990200=7;
  ELSE IF R4120000=4 THEN R4000300=7; ELSE IF R4120000=5 THEN R4010400=7;
  ELSE IF R4120000=6 THEN R4019200=7; ELSE IF R4120000=7 THEN R4026900=7;
  ELSE IF R4120000=8 THEN R4033000=7; ELSE IF R4120000=9 THEN R4038000=7;
  ELSE IF R4120000=10 THEN R4041500=7; ELSE IF R4120000=12 THEN R4044000=7;
  ELSE IF R4120000=13 THEN R4047300=7; END;

IF R3813900=1 THEN DO; IF R3814000=1 THEN R3639000=1;
  ELSE IF R3814000=2 THEN R3659700=1; ELSE IF R3814000=3 THEN R3679100=1;
  ELSE IF R3814000=4 THEN R3699100=1; ELSE IF R3814000=5 THEN R3718400=1;
  ELSE IF R3814000=6 THEN R3736900=1; ELSE IF R3814000=7 THEN R3753400=1;
  ELSE IF R3814000=8 THEN R3769600=1; ELSE IF R3814000=9 THEN R3785600=1;
  ELSE IF R3814000=10 THEN R3796200=1; ELSE IF R3814000=11 THEN R3804300=1; END;
IF R3815900=1 THEN DO; IF R3816000=1 THEN R3639000=2;
  ELSE IF R3816000=2 THEN R3659700=2; ELSE IF R3816000=3 THEN R3679100=2;
  ELSE IF R3816000=4 THEN R3699100=2; ELSE IF R3816000=5 THEN R3718400=2;
  ELSE IF R3816000=6 THEN R3736900=2; ELSE IF R3816000=7 THEN R3753400=2;
  ELSE IF R3816000=8 THEN R3769600=2; ELSE IF R3816000=9 THEN R3785600=2;
  ELSE IF R3816000=10 THEN R3796200=2; ELSE IF R3816000=11 THEN R3804300=2; END;

```

```

IF R3817900=1 THEN DO; IF R3818000=1 THEN R3639000=3;
  ELSE IF R3818000=2 THEN R3659700=3; ELSE IF R3818000=3 THEN R3679100=3;
  ELSE IF R3818000=4 THEN R3699100=3; ELSE IF R3818000=5 THEN R3718400=3;
  ELSE IF R3818000=6 THEN R3736900=3; ELSE IF R3818000=7 THEN R3753400=3;
  ELSE IF R3818000=8 THEN R3769600=3; ELSE IF R3818000=9 THEN R3785600=3;
  ELSE IF R3818000=10 THEN R3796200=3; ELSE IF R3818000=11 THEN R3804300=3; END;
IF R3820100=1 THEN DO; IF R3820200=1 THEN R3639000=4;
  ELSE IF R3820200=2 THEN R3659700=4; ELSE IF R3820200=3 THEN R3679100=4;
  ELSE IF R3820200=4 THEN R3699100=4; ELSE IF R3820200=5 THEN R3718400=4;
  ELSE IF R3820200=6 THEN R3736900=4; ELSE IF R3820200=7 THEN R3753400=4;
  ELSE IF R3820200=8 THEN R3769600=4; ELSE IF R3820200=9 THEN R3785600=4;
  ELSE IF R3820200=10 THEN R3796200=4; ELSE IF R3820200=11 THEN R3804300=4; END;
IF R3822100=1 THEN DO; IF R3822200=1 THEN R3639000=5;
  ELSE IF R3822200=2 THEN R3659700=5; ELSE IF R3822200=3 THEN R3679100=5;
  ELSE IF R3822200=4 THEN R3699100=5; ELSE IF R3822200=5 THEN R3718400=5;
  ELSE IF R3822200=6 THEN R3736900=5; ELSE IF R3822200=7 THEN R3753400=5;
  ELSE IF R3822200=8 THEN R3769600=5; ELSE IF R3822200=9 THEN R3785600=5;
  ELSE IF R3822200=10 THEN R3796200=5; ELSE IF R3822200=11 THEN R3804300=5; END;
IF R3824100=1 THEN DO; IF R3824200=1 THEN R3639000=6;
  ELSE IF R3824200=2 THEN R3659700=6; ELSE IF R3824200=3 THEN R3679100=6;
  ELSE IF R3824200=4 THEN R3699100=6; ELSE IF R3824200=5 THEN R3718400=6;
  ELSE IF R3824200=6 THEN R3736900=6; ELSE IF R3824200=7 THEN R3753400=6;
  ELSE IF R3824200=8 THEN R3769600=6; ELSE IF R3824200=9 THEN R3785600=6;
  ELSE IF R3824200=10 THEN R3796200=6; ELSE IF R3824200=11 THEN R3804300=6; END;
IF R3826100=1 THEN DO; IF R3826200=1 THEN R3639000=7;
  ELSE IF R3826200=2 THEN R3659700=7; ELSE IF R3826200=3 THEN R3679100=7;
  ELSE IF R3826200=4 THEN R3699100=7; ELSE IF R3826200=5 THEN R3718400=7;
  ELSE IF R3826200=6 THEN R3736900=7; ELSE IF R3826200=7 THEN R3753400=7;
  ELSE IF R3826200=8 THEN R3769600=7; ELSE IF R3826200=9 THEN R3785600=7;
  ELSE IF R3826200=10 THEN R3796200=7; ELSE IF R3826200=11 THEN R3804300=7; END;
IF R3828100=1 THEN DO; IF R3828200=1 THEN R3639000=8;
  ELSE IF R3828200=2 THEN R3659700=8; ELSE IF R3828200=3 THEN R3679100=8;
  ELSE IF R3828200=4 THEN R3699100=8; ELSE IF R3828200=5 THEN R3718400=8;
  ELSE IF R3828200=6 THEN R3736900=8; ELSE IF R3828200=7 THEN R3753400=8;
  ELSE IF R3828200=8 THEN R3769600=8; ELSE IF R3828200=9 THEN R3785600=8;
  ELSE IF R3828200=10 THEN R3796200=8; ELSE IF R3828200=11 THEN R3804300=8; END;
IF R3830000=1 THEN DO; IF R3830100=1 THEN R3639000=9;
  ELSE IF R3830100=2 THEN R3659700=9; ELSE IF R3830100=3 THEN R3679100=9;
  ELSE IF R3830100=4 THEN R3699100=9; ELSE IF R3830100=5 THEN R3718400=9;
  ELSE IF R3830100=6 THEN R3736900=9; ELSE IF R3830100=7 THEN R3753400=9;
  ELSE IF R3830100=8 THEN R3769600=9; ELSE IF R3830100=9 THEN R3785600=9;
  ELSE IF R3830100=10 THEN R3796200=9; ELSE IF R3830100=11 THEN R3804300=9; END;
IF R3831900=1 THEN DO; IF R3832000=1 THEN R3639000=10;
  ELSE IF R3832000=2 THEN R3659700=10; ELSE IF R3832000=3 THEN R3679100=10;
  ELSE IF R3832000=4 THEN R3699100=10; ELSE IF R3832000=5 THEN R3718400=10;
  ELSE IF R3832000=6 THEN R3736900=10; ELSE IF R3832000=7 THEN R3753400=10;
  ELSE IF R3832000=8 THEN R3769600=10; ELSE IF R3832000=9 THEN R3785600=10;
  ELSE IF R3832000=10 THEN R3796200=10; ELSE IF R3832000=11 THEN R3804300=10; END;
IF R3833700=1 THEN DO; IF R3833800=1 THEN R3639000=11;
  ELSE IF R3833800=2 THEN R3659700=11; ELSE IF R3833800=3 THEN R3679100=11;
  ELSE IF R3833800=4 THEN R3699100=11; ELSE IF R3833800=5 THEN R3718400=11;
  ELSE IF R3833800=6 THEN R3736900=11; ELSE IF R3833800=7 THEN R3753400=11;
  ELSE IF R3833800=8 THEN R3769600=11; ELSE IF R3833800=9 THEN R3785600=11;
  ELSE IF R3833800=10 THEN R3796200=11; ELSE IF R3833800=11 THEN R3804300=11; END;
IF R3835400=1 THEN DO; IF R3835500=1 THEN R3639000=12;
  ELSE IF R3835500=2 THEN R3659700=12; ELSE IF R3835500=3 THEN R3679100=12;
  ELSE IF R3835500=4 THEN R3699100=12; ELSE IF R3835500=5 THEN R3718400=12;
  ELSE IF R3835500=6 THEN R3736900=12; ELSE IF R3835500=7 THEN R3753400=12;
  ELSE IF R3835500=8 THEN R3769600=12; ELSE IF R3835500=9 THEN R3785600=12;
  ELSE IF R3835500=10 THEN R3796200=12; ELSE IF R3835500=11 THEN R3804300=12; END;

```

```

IF R4050800=1 THEN DO; IF R4050900=1 THEN R3960000=1;
  ELSE IF R4050900=2 THEN R3971900=1; ELSE IF R4050900=3 THEN R3983200=1;
  ELSE IF R4050900=4 THEN R3993600=1; ELSE IF R4050900=5 THEN R4003600=1;
  ELSE IF R4050900=6 THEN R4013700=1; ELSE IF R4050900=7 THEN R4022500=1;
  ELSE IF R4050900=8 THEN R4029400=1; ELSE IF R4050900=9 THEN R4035300=1; END;
IF R4052800=1 THEN DO; IF R4052900=1 THEN R3960000=2;
  ELSE IF R4052900=2 THEN R3971900=2; ELSE IF R4052900=3 THEN R3983200=2;
  ELSE IF R4052900=4 THEN R3993600=2; ELSE IF R4052900=5 THEN R4003600=2;
  ELSE IF R4052900=6 THEN R4013700=2; ELSE IF R4052900=7 THEN R4022500=2;
  ELSE IF R4052900=8 THEN R4029400=2; ELSE IF R4052900=9 THEN R4035300=2; END;
IF R4054800=1 THEN DO; IF R4054900=1 THEN R3960000=3;
  ELSE IF R4054900=2 THEN R3971900=3; ELSE IF R4054900=3 THEN R3983200=3;
  ELSE IF R4054900=4 THEN R3993600=3; ELSE IF R4054900=5 THEN R4003600=3;
  ELSE IF R4054900=6 THEN R4013700=3; ELSE IF R4054900=7 THEN R4022500=3;
  ELSE IF R4054900=8 THEN R4029400=3; ELSE IF R4054900=9 THEN R4035300=3; END;
IF R4056800=1 THEN DO; IF R4056900=1 THEN R3960000=4;
  ELSE IF R4056900=2 THEN R3971900=4; ELSE IF R4056900=3 THEN R3983200=4;
  ELSE IF R4056900=4 THEN R3993600=4; ELSE IF R4056900=5 THEN R4003600=4;
  ELSE IF R4056900=6 THEN R4013700=4; ELSE IF R4056900=7 THEN R4022500=4;
  ELSE IF R4056900=8 THEN R4029400=4; ELSE IF R4056900=9 THEN R4035300=4; END;
IF R4058800=1 THEN DO; IF R4058900=1 THEN R3960000=5;
  ELSE IF R4058900=2 THEN R3971900=5; ELSE IF R4058900=3 THEN R3983200=5;
  ELSE IF R4058900=4 THEN R3993600=5; ELSE IF R4058900=5 THEN R4003600=5;
  ELSE IF R4058900=6 THEN R4013700=5; ELSE IF R4058900=7 THEN R4022500=5;
  ELSE IF R4058900=8 THEN R4029400=5; ELSE IF R4058900=9 THEN R4035300=5; END;
IF R4060700=1 THEN DO; IF R4060800=1 THEN R3960000=6;
  ELSE IF R4060800=2 THEN R3971900=6; ELSE IF R4060800=3 THEN R3983200=6;
  ELSE IF R4060800=4 THEN R3993600=6; ELSE IF R4060800=5 THEN R4003600=6;
  ELSE IF R4060800=6 THEN R4013700=6; ELSE IF R4060800=7 THEN R4022500=6;
  ELSE IF R4060800=8 THEN R4029400=6; ELSE IF R4060800=9 THEN R4035300=6; END;
IF R4062600=1 THEN DO; IF R4062700=1 THEN R3960000=7;
  ELSE IF R4062700=2 THEN R3971900=7; ELSE IF R4062700=3 THEN R3983200=7;
  ELSE IF R4062700=4 THEN R3993600=7; ELSE IF R4062700=5 THEN R4003600=7;
  ELSE IF R4062700=6 THEN R4013700=7; ELSE IF R4062700=7 THEN R4022500=7;
  ELSE IF R4062700=8 THEN R4029400=7; ELSE IF R4062700=9 THEN R4035300=7; END;
IF R4064500=1 THEN DO; IF R4064600=1 THEN R3960000=8;
  ELSE IF R4064600=2 THEN R3971900=8; ELSE IF R4064600=3 THEN R3983200=8;
  ELSE IF R4064600=4 THEN R3993600=8; ELSE IF R4064600=5 THEN R4003600=8;
  ELSE IF R4064600=6 THEN R4013700=8; ELSE IF R4064600=7 THEN R4022500=8;
  ELSE IF R4064600=8 THEN R4029400=8; ELSE IF R4064600=9 THEN R4035300=8; END;
IF R4066400=1 THEN DO; IF R4066500=1 THEN R3960000=9;
  ELSE IF R4066500=2 THEN R3971900=9; ELSE IF R4066500=3 THEN R3983200=9;
  ELSE IF R4066500=4 THEN R3993600=9; ELSE IF R4066500=5 THEN R4003600=9;
  ELSE IF R4066500=6 THEN R4013700=9; ELSE IF R4066500=7 THEN R4022500=9;
  ELSE IF R4066500=8 THEN R4029400=9; ELSE IF R4066500=9 THEN R4035300=9; END;
IF R4067300=1 THEN DO; IF R4067400=1 THEN R3960000=10;
  ELSE IF R4067400=2 THEN R3971900=10; ELSE IF R4067400=3 THEN R3983200=10;
  ELSE IF R4067400=4 THEN R3993600=10; ELSE IF R4067400=5 THEN R4003600=10;
  ELSE IF R4067400=6 THEN R4013700=10; ELSE IF R4067400=7 THEN R4022500=10;
  ELSE IF R4067400=8 THEN R4029400=10; ELSE IF R4067400=9 THEN R4035300=10; END;
IF R4069200=1 THEN DO; IF R4069300=1 THEN R3960000=11;
  ELSE IF R4069300=2 THEN R3971900=11; ELSE IF R4069300=3 THEN R3983200=11;
  ELSE IF R4069300=4 THEN R3993600=11; ELSE IF R4069300=5 THEN R4003600=11;
  ELSE IF R4069300=6 THEN R4013700=11; ELSE IF R4069300=7 THEN R4022500=11;
  ELSE IF R4069300=8 THEN R4029400=11; ELSE IF R4069300=9 THEN R4035300=11; END;
IF R4071100=1 THEN DO; IF R4071200=1 THEN R3960000=12;
  ELSE IF R4071200=2 THEN R3971900=12; ELSE IF R4071200=3 THEN R3983200=12;
  ELSE IF R4071200=4 THEN R3993600=12; ELSE IF R4071200=5 THEN R4003600=12;
  ELSE IF R4071200=6 THEN R4013700=12; ELSE IF R4071200=7 THEN R4022500=12;
  ELSE IF R4071200=8 THEN R4029400=12; ELSE IF R4071200=9 THEN R4035300=12; END;

```



```

IF R4073000=1 THEN DO; IF R4073100=1 THEN R3960000=13;
  ELSE IF R4073100=2 THEN R3971900=13; ELSE IF R4073100=3 THEN R3983200=13;
  ELSE IF R4073100=4 THEN R3993600=13; ELSE IF R4073100=5 THEN R4003600=13;
  ELSE IF R4073100=6 THEN R4013700=13; ELSE IF R4073100=7 THEN R4022500=13;
  ELSE IF R4073100=8 THEN R4029400=13; ELSE IF R4073100=9 THEN R4035300=13; END;
IF R4074900=1 THEN DO; IF R4075000=1 THEN R3960000=14;
  ELSE IF R4075000=2 THEN R3971900=14; ELSE IF R4075000=3 THEN R3983200=14;
  ELSE IF R4075000=4 THEN R3993600=14; ELSE IF R4075000=5 THEN R4003600=14;
  ELSE IF R4075000=6 THEN R4013700=14; ELSE IF R4075000=7 THEN R4022500=14;
  ELSE IF R4075000=8 THEN R4029400=14; ELSE IF R4075000=9 THEN R4035300=14; END;
IF R4076800=1 THEN DO; IF R4076900=1 THEN R3960000=15;
  ELSE IF R4076900=2 THEN R3971900=15; ELSE IF R4076900=3 THEN R3983200=15;
  ELSE IF R4076900=4 THEN R3993600=15; ELSE IF R4076900=5 THEN R4003600=15;
  ELSE IF R4076900=6 THEN R4013700=15; ELSE IF R4076900=7 THEN R4022500=15;
  ELSE IF R4076900=8 THEN R4029400=15; ELSE IF R4076900=9 THEN R4035300=15; END;

```

/*Hourly rate of pay, KEY*/

```

array hrop R3644200 R3664900 R3684300 R3704300 R3723500 R3740700 R3758100 R3773800
R3788200 R3799000 R3805800 R3962700 R3974600 R3985900 R3996300
R4006200 R4016300 R4024900 R4031000 R4036900 HROP21;
array ehrop R3644300 R3665000 R3684400 EHROP4 R3723600 EHROP6 R3758200 EHROP8-EHROP11
EHROP12-EHROP21;
array wrop R3647800 R3667800 R3687600 R3707200 R3726000 R3742700 R3760200 R3775700
WROP9-WROP11 R3965600 R3977000 R3987600 R3997800 R4008000 R4017000
WROP18-WROP21;
array ewrop R3647900 R3667900 R3687700 R3707300 R3726100 EWROP6 R3760300
EWROP8- EWROP11 R3965700 R3977100 R3987700 R3997900 R4008100
R4017100 EWROP18-EWROP21;
array mrop R3648300 R3668300 R3688100 R3707700 R3726500 R3743100 MROP7 R3776000
MROP9-MROP11 R3966100 R3977500 R3988100 R3998200 R4008400 R4017400
MROP18-MROP21;
array emrop R3648400 EMROP2-EMROP11 R3966200 R3977600 R3988200 R3998300
EMROP16-EMROP21;
array arop R3648800 R3668600 R3688400 R3708000 R3726800 R3743400 R3760700
R3776300 R3788900 AROP10-AROP11 R3966600 R3978000 R3988600 R3998700
R4008700 R4017700 R4025600 R4032200 AROP20 R4046600;
array earop R3648900 R3668700 R3688500 R3708100 R3726900 EAROP6 R3760800
EAROP8-EAROP11 R3966700 R3978100 R3988700 R3998800 EAROP16-EAROP21;
array brop R3649300 R3669100 R3688900 R3708500 R3727200 R3743700 R3761100
BROP8-BROP11 R3967100 R3978500 R3989100 R3999200 R4009100
R4018100 R4025900 BROP19-BROP21;
array ebrop R3649400 R3669200 R3689000 EBROP4-EBROP11
R3967200 R3978600 EBROP14 EBROP15 R4009200 EBROP17-EBROP21;
ARRAY OROP R3650200 R3669900 R3689800 R3709200 R3727900 R3744400 R3761800
R3776900 OROP9-OROP11 R3968000 R3979300 R3989800 R3999900 R4009900
R4018800 R4026500 OROP19-OROP21;

array swhrp R3650600 R3670300 R3690100 R3709600 R3728200 R3744700 R3762100 R3777200
R3789600 R3800000 R3806800 R3968400 R3979700 R3990100 R4000200 R4010300
R4019100 R4026800 R4032900 R4037900 R4047200;
array cow R3637700 R3658400 R3677900 R3697900 R3717200 R3735800 R3752300 R3768500
R3784600 R3795400 R3803400 R3958700 R3970700 R3982000 R3992400 R4002500
R4012600 R4021400 R4028500 R4034500 R4044600;
array ushd R3639100 R3659800 R3679200 R3699200 R3718500 R3737000 R3753500 R3769700
R3785700 R3796300 R3804400 R3960100 R3972000 R3983300 R3993700 R4003700
R4013800 R4022600 R4029500 R4035400 R4045000;
array ushw R3639600 R3660300 R3679700 R3699700 R3719000 R3737500 R3754000 R3770200
R3786200 R3796600 R3804700 R3960600 R3972500 R3983800 R3994200 R4004200
R4014300 R4023100 R4029800 R4035700 R4045400;
array turp R3643900 R3664600 R3684000 R3704000 R3723200 R3740400 R3757800 R3773500
R3787900 R3798700 R3805500 R3962400 R3974300 R3985600 R3996000 R4005900
R4016000 R4024600 R4030700 R4036600 R4046300;

```

```
array uswy R3649800 R3669500 R3689400 R3708800 R3727500 R3744000 R3761400 R3776600
R3789200 R3799700 R3806500 R3967600 R3978900 R3989400 R3999500 R4009500
R4018400 R4026200 R4032500 R4037600 R4046900;
```

```
do over SWHRP;
  IF EHROP>0 THEN HROP=EHROP; IF EWROP>0 THEN WROP=EWROP;
  IF EMROP>0 THEN MROP=EMROP; IF EAROP>0 THEN AROP=EAROP;
  IF EBROP>0 THEN BROP=EBROP;
  IF COW GE 1 THEN DO;
    if turp=1 & HROP>0 then swhrp=Hrop; IF USHW LT USHD THEN USHW=-3;
    if ushw>0 then do;
      if ((turp=2 | TURP=3 | TURP=7) & WROP>0) then swhrp=((Wrop*100)/ushw);
      if (turp=4 & BROP>0) then swhrp=((Brop*100)/(ushw*2));
      if ((turp=5 | TURP=8) & MROP>0) then swhrp=((Mrop*100)/(ushw*4.33));
      if (turp=6 & uswy ge 1 & AROP>0) then swhrp=((Arop*100)/(ushw*uswy));
    IF OROP>0 THEN SWHRP=OROP;
    swHRP = FLOOR(swHRP + .5); IF swHRP < 1 | swHRP >99999 THEN swHRP = .;
  END; END; if swhrp=. then swhrp=-4; END;
```

```
/*Re-coding of text entries HEA_9D, PAR-56 and PAR-66
```

```
IF HEA_9D='ARM' ' then hea9d=13;
IF HEA_9D='BACK' ' then hea9d=20;
IF HEA_9D='BASAL CELL CANCER' ' then hea9d=14;
IF HEA_9D='BEHIND BREAST ON RIB' ' then hea9d= 2;
IF HEA_9D='BLADDER' ' then hea9d= 1;
IF HEA_9D='BLADDER, UTERUS' ' then hea9d= 1;
IF HEA_9D='BLOOD' ' then hea9d=20;
IF HEA_9D='BRAIN' ' then hea9d= 4;
IF HEA_9D='BREAST' ' then hea9d= 3;
IF HEA_9D='BREAST CANCER' ' then hea9d= 3;
IF HEA_9D='BREAST RIGHT' ' then hea9d= 3;
IF HEA_9D='BREAST-' ' then hea9d= 3;
IF HEA_9D='BREST' ' then hea9d= 3;
IF HEA_9D='CAROTID GLAND' ' then hea9d=20;
IF HEA_9D='CEREVICAL' ' then hea9d= 5;
IF HEA_9D='CERVAEAL' ' then hea9d= 5;
IF HEA_9D='CERVERICAL' ' then hea9d= 5;
IF HEA_9D='CERVEX' ' then hea9d= 5;
IF HEA_9D='CERVICAL' ' then hea9d= 5;
IF HEA_9D='CERVIX' ' then hea9d= 5;
IF HEA_9D='CERVIX CANCER' ' then hea9d= 5;
IF HEA_9D='CHEEK BONE IN FACE' ' then hea9d=14;
IF HEA_9D='CHEEKBONE AREA FACE' ' then hea9d=14;
IF HEA_9D='CHEST WALL' ' then hea9d=20;
IF HEA_9D='COLIN CANCER' ' then hea9d= 6;
IF HEA_9D='COLLON' ' then hea9d= 6;
IF HEA_9D='COLON' ' then hea9d= 6;
IF HEA_9D='COLON/RECTAL' ' then hea9d= 6;
IF HEA_9D='D' ' then hea9d=-2;
IF HEA_9D='DID NOT OCCUR' ' then hea9d=-3;
IF HEA_9D='EAR' ' then hea9d=15;
IF HEA_9D='ENDOMETRIUM' ' then hea9d=18;
IF HEA_9D='ESOPHAGUS' ' then hea9d= 9;
IF HEA_9D='FACE' ' then hea9d=14;
IF HEA_9D='FACE, UNDER EYE' ' then hea9d=14;
IF HEA_9D='FALLOPIAN TUBE' ' then hea9d=18;
IF HEA_9D='FEMAL PART OF BODY' ' then hea9d=18;
IF HEA_9D='FEMALE' ' then hea9d=18;
IF HEA_9D='FEMALE ORGANS' ' then hea9d=18;
IF HEA_9D='FOREHEAD' ' then hea9d=15;
IF HEA_9D='GROIN' ' then hea9d=20;
IF HEA_9D='HEAD' ' then hea9d=15;
IF HEA_9D='HODGKINSON DISEASE' ' then hea9d=20;
IF HEA_9D='IN HER UTERUS' ' then hea9d=18;
IF HEA_9D='KIDNEY' ' then hea9d= 7;
IF HEA_9D='LARGE INTESTINE' ' then hea9d= 6;
IF HEA_9D='LEFT BREAST' ' then hea9d= 3;
IF HEA_9D='LEFT EYE' ' then hea9d=20;
```

```

IF HEA_9D='LEFT LEG' ' then hea9d=16;
IF HEA_9D='LEFT LUNG' ' then hea9d= 8;
IF HEA_9D='LEFT LUNG TEMPLE' ' then hea9d= 8;
IF HEA_9D='LEFT SIDE' ' then hea9d=20;
IF HEA_9D='LEG' ' then hea9d=16;
IF HEA_9D='LEGS' ' then hea9d=16;
IF HEA_9D='LIP' ' then hea9d=14;
IF HEA_9D='LUNG' ' then hea9d= 8;
IF HEA_9D='LUNGS' ' then hea9d= 8;
IF HEA_9D='LYMPH NODES' ' then hea9d=20;
IF HEA_9D='LYMPH NODES IN NECK' ' then hea9d=20;
IF HEA_9D='LYMPHOMA' ' then hea9d=20;
IF HEA_9D='MOUTH' ' then hea9d= 9;
IF HEA_9D='NECK' ' then hea9d=15;
IF HEA_9D='NECK, RIBS AND BREAS' ' then hea9d=15;
IF HEA_9D='NOSE' ' then hea9d=14;
IF HEA_9D='OPENING BIRTH CHANNE' ' then hea9d=19;
IF HEA_9D='OVARIAN' ' then hea9d=10;
IF HEA_9D='OVARIAN CANCER' ' then hea9d=10;
IF HEA_9D='OVARIES' ' then hea9d=10;
IF HEA_9D='OVARY' ' then hea9d=10;
IF HEA_9D='OVERIES' ' then hea9d=10;
IF HEA_9D='PANCREAS' ' then hea9d=11;
IF HEA_9D='R' ' then hea9d=-1;
IF HEA_9D='RECTAL TUMOR' ' then hea9d= 6;
IF HEA_9D='RIBS' ' then hea9d= 2;
IF HEA_9D='RIGHT BREAST' ' then hea9d= 3;
IF HEA_9D='RIGHT LEG-LOWER TIBI' ' then hea9d=16;
IF HEA_9D='RT BREAT' ' then hea9d= 3;
IF HEA_9D='SINUSES' ' then hea9d=15;
IF HEA_9D='SKIN' ' then hea9d=17;
IF HEA_9D='SKIN CANCER' ' then hea9d=17;
IF HEA_9D='SKIN CANCER MELANOMA' ' then hea9d=17;
IF HEA_9D='SKIN FACE' ' then hea9d=14;
IF HEA_9D='SKIN ON ARM' ' then hea9d=13;
IF HEA_9D='SKIN ON FOREHEAD' ' then hea9d=15;
IF HEA_9D='SOFT PALLET' ' then hea9d= 9;
IF HEA_9D='SPLEEN' ' then hea9d=20;
IF HEA_9D='STOMACH' ' then hea9d=20;
IF HEA_9D='THROAT' ' then hea9d= 9;
IF HEA_9D='THYROID' ' then hea9d=20;
IF HEA_9D='TONGUE' ' then hea9d= 9;
IF HEA_9D='UPPER BODY' ' then hea9d=20;
IF HEA_9D='UPPER RIGHT ARM' ' then hea9d=13;
IF HEA_9D='URINARY TRACT' ' then hea9d=20;
IF HEA_9D='UTERAS' ' then hea9d=18;
IF HEA_9D='UTERIAN' ' then hea9d=18;
IF HEA_9D='UTERINE' ' then hea9d=18;
IF HEA_9D='UTERINE AREA' ' then hea9d=18;
IF HEA_9D='UTERIS' ' then hea9d=18;
IF HEA_9D='UTEROUS' ' then hea9d=18;
IF HEA_9D='UTERUS' ' then hea9d=18;
IF HEA_9D='UTERUS AND BREAST' ' then hea9d=18;
IF HEA_9D='UTERUS, OVARIES' ' then hea9d=18;
IF HEA_9D='VAGINAL' ' then hea9d=19;
IF HEA_9D='VAGINAL TUMORS' ' then hea9d=19;
IF HEA_9D='WOMB' ' then hea9d=18;
IF HEA_9D='WOMB CANCER' ' then hea9d=18;

if par_56_1="BECAUSE BROTHER MANIPULATED FUNDS. " then par56=6;
if par_56_1="BECAUSE FATHER DISOWNED XXXXXXXX BECAUSE SHE LEFT HOME AND " then par56=7;
if par_56_1="BECAUSE HE CHOSE TO DO IT THAT WAY " then par56=8;
if par_56_1="BECAUSE I WAS THE ESCEUTOR OF THE ESTATE SO I RECEIVED MORE" then par56=6;
if par_56_1="BECAUSE MY SISTER WAS EXECUTRIX OF ESTATE " then par56=6;
if par_56_1="BECAUSE SHE WAS THE CARE GIVER " then par56=1;
if par_56_1="BIOLOGICAL CHILDREN RECEIVED MORE FROM ESTATE. " then par56=4;
if par_56_1="CHILD WAS ADOPTED & SHE THINKS SHE GOD A LITTLE LESS. " then par56=4;
if par_56_1="CONTESTING OF THE WILL AND LEGAL FEES " then par56=7;
if par_56_1="EUROPEAN MALE INHERITANCE " then par56=3;
if par_56_1="FAMILY DISAGREEMENT " then par56=7;
if par_56_1="HE DID NOT LEAVE ANYTHING BUT 100.00 TO ONE CHILD " then par56=8;
if par_56_1="HE HAD AN ACCOUNT IN MY NAME " then par56=8;

```

```

if par_56_1="HE PAID THE REAL ESTATE TAXES ON THE PROPERTY AND HE GOT IT " then par56=6;
if par_56_1="I DONT KNOW'' " then par56=-2;
if par_56_1="I LIVED THE HOUSE THAT I RECEIVED AS PART OF THE ESTATE " then par56=1;
if par_56_1="I TOOK CARE OF HIM " then par56=1;
if par_56_1="IT WAS DIVIDED AMONG THE 3 THAT HE STILL SUPPORTED. " then par56=2;
if par_56_1="LEFT TO PERSON WHO TOOK CARE OF THEM . DAUGHTER XXXXXX " then par56=1;
if par_56_1="MEDICAL AND HOSPITAL AND FINAL EXPENSES USED UP MAJORITY " then par56=8;
if par_56_1="MY BROTHER GOT THE HOUSE BECAUSE HE WAS LIVING THERE AT THE " then par56=1;
if par_56_1="MY SISTER LIVED WITH HIM, SO HOME LEFT TO HER " then par56=1;
if par_56_1="N " then par56=-3;
if par_56_1="NOBODY WANTED THE HOUSE SO HE GAVE IT TO ME " then par56=5;
if par_56_1="ONE SISTER HAD TAKEN CARE OF FATHER DURING ILLNESS " then par56=1;
if par_56_1="ONE TOOK CARE OF THE DAD " then par56=1;
if par_56_1="ONE WAS POORER THAN THE OTHER.MOST OF HIS MONEY WENT TO THE " then par56=2;
if par_56_1="OTHER PERSON DID NOT WANT ANY OF THE ESTATE " then par56=2;
if par_56_1="RESP BIOLOGICAL, OTHER NOT " then par56=4;
if par_56_1="SHE HAD BOUGHT HER BROTHERS SHARE AND HE HAD PASSED AWAY " then par56=8;
if par_56_1="SHE RECEIVED FATHERS ESTATE AND SISTER RECEIVED MOTHERS " then par56=3;
if par_56_1="SISTER LIVED WITH FATHER AND SHE WILL CONTINUE TO OCCUPY " then par56=1;
if par_56_1="SISTER'S FAMILY IS WEALTHY AND VOLUNTARILY TOOK HERSELF OUT " then par56=2;
if par_56_1="SP THE ONE WHO TOOK CARE OF PARENTS IN LATER YEARS " then par56=1;
if par_56_1="THE ONE SON WAS THE EXECUTOR " then par56=6;
if par_56_1="WE DECIDED THE OLDEST BROTHER NEEDED MORE HELP " then par56=2;

```

```

IF PAR_66_1="ALL THE OTHERS WERE MARRIED, EVERYTHING WENT TO THE SISTER " THEN PAR66=3;
IF PAR_66_1="BECAUSE BROTHER TOOK CARE OF HER FINANCIALLY AND WE DIDN'T " THEN PAR66=1;
IF PAR_66_1="BECAUSE I DID NOT WANT IT " THEN PAR66=8;
IF PAR_66_1="BECAUSE I WAS THE ONLY ONE TO TAKE CARE OF HER WHEN SHE WAS" THEN PAR66=1;
IF PAR_66_1="BECAUSE MY BROTHERR DISASOCIATED HIMSELF FROM THE FAMILY " THEN PAR66=7;
IF PAR_66_1="BECAUSE ONE DAUGHTER CARED FOR HER AND LIVED IN SAME HOUSE " THEN PAR66=1;
IF PAR_66_1="BECAUSE SOME OF THE CHILDREN HAD THEIR OWN CHILDREN TO " THEN PAR66=2;
IF PAR_66_1="BECAUSE THAT WAS HER WISHES " THEN PAR66=8;
IF PAR_66_1="BECAUSE THEY HAD CUT FAMILY TIES PRIOR TO HER DEATH " THEN PAR66=7;
IF PAR_66_1="BECAUSE XXXX CARED FOR HER MOTHER IN HER LATER YEARS " THEN PAR66=1;
IF PAR_66_1="BOTHER TOOK CARE OF HER " THEN PAR66=1;
IF PAR_66_1="BROTHER IN XXXXXXXX INHERITED ENTIRE ESTATE " THEN PAR66=3;
IF PAR_66_1="BROTHER RECEIVED FARM, ALL OTHER 9 CHILDREN RECEIVED MONEY " THEN PAR66=5;
IF PAR_66_1="BROTHER WORKED FARM " THEN PAR66=5;
IF PAR_66_1="CD'S WERE IN OTHER NAME " THEN PAR66=8;
IF PAR_66_1="DID NOT TREAT HER EQUALLY " THEN PAR66=7;
IF PAR_66_1="DIDN'T FEEL HER DAUGHTERS WERE CAPABLE OF HANDLING IT " THEN PAR66=7;
IF PAR_66_1="DONT KNOW " THEN PAR66=-2;
IF PAR_66_1="DON'T KNOW " THEN PAR66=-2;
IF PAR_66_1="FAMILY SITUATION-- " THEN PAR66=7;
IF PAR_66_1="FAVORED HER ONLY SON " THEN PAR66=3;
IF PAR_66_1="FAVORITISM " THEN PAR66=3;
IF PAR_66_1="HAVE ONE SISTER THAT IS SINGLE AND TOOK CARE OF MOTHER " THEN PAR66=1;
IF PAR_66_1="HE VWAS THE YOUNGEST IT WAS UNDERSTOOD " THEN PAR66=2;
IF PAR_66_1="HOUSE WASW LEFT TO ONE SISTER AND THE REST WAS DIVIDED UP " THEN PAR66=5;
IF PAR_66_1="I DON'T KNOW SINCE I NO LONGER LIVE IN XX. THE THREE WHO " THEN PAR66=8;
IF PAR_66_1="I DONT KNOW " THEN PAR66=-2;
IF PAR_66_1="I TOOK CARE OF HER WHEN SHE WAS SICK. " THEN PAR66=1;
IF PAR_66_1="JUST MOTHER'S CHOICE " THEN PAR66=8;
IF PAR_66_1="LAND COULD NOT BE DISTRIBUTED EQUALLY " THEN PAR66=5;
IF PAR_66_1="LEFT THE HOUSE TO HER SISITER WHO LIVED WITH HER " THEN PAR66=1;
IF PAR_66_1="LEFT THE HOUSE TO THE BABY GIRL. OTHERS WERE GROWN " THEN PAR66=5;
IF PAR_66_1="LEFT TO CARE FOR RETARTED CHILD " THEN PAR66=2;
IF PAR_66_1="LIVED ON A SON'S PROPERTY SO HE GOT HER MOBILE HOME " THEN PAR66=1;
IF PAR_66_1="MOTHER WANTED IT THAT WAY " THEN PAR66=8;
IF PAR_66_1="MY BROTHER GOT AN EQUAL AMOUNT ON THE PROPERTY PLUS A TRUCK" THEN PAR66=8;
IF PAR_66_1="MY MOTHER DIDN'T LIKE MY BORTHERS WIFE " THEN PAR66=7;
IF PAR_66_1="N " THEN PAR66=-3;
IF PAR_66_1="OLDER SISTER CARED FOR MOTHER " THEN PAR66=1;
IF PAR_66_1="OLDEST BOY GOT THE FARM " THEN PAR66=5;
IF PAR_66_1="ONE CHILD BORROWED MONEY AND NEVER PAID IT BACK " THEN PAR66=2;
IF PAR_66_1="ONE SISTER LIVED AT HOME AND PROVIDING A HOME FOR ANOTHER " THEN PAR66=1;
IF PAR_66_1="ONE SISTER LIVED WITH AND TOOK CARE OF MOTHER " THEN PAR66=1;
IF PAR_66_1="ONE SISTER LIVED WITH MOTHER AND THINGS WERE OWNED JOINTLY " THEN PAR66=1;
IF PAR_66_1="ONE SON THAT ACTUALLY LIVED ON AND WORKED THE FARM GOT " THEN PAR66=1;
IF PAR_66_1="PERSONAL REASONS " THEN PAR66=8;
IF PAR_66_1="PROPERTY WENT TO OLDEST CHILD/A BROTHER OF RESP. THEN WHEN " THEN PAR66=5;

```

```

IF PAR_66_1="SHE FELT THE SON SHOULD HAVE MORE TO HELP HIM" " THEN PAR66=2;
IF PAR_66_1="SHE GAVE A LAKE HOUSE TO ONE BROTHER" " THEN PAR66=5;
IF PAR_66_1="SHE GOT CASH AND BROTHER GOT PROPERTY" " THEN PAR66=5;
IF PAR_66_1="SHE HAD GIVEN ALREADY TO HIM IN HER LIFETIME" " THEN PAR66=2;
IF PAR_66_1="SHE LEFT ALL TO DAUGHTER BECAUSE SHE HAD HELPED SON ALL" " THEN PAR66=2;
IF PAR_66_1="SHE LIVED WITH HER" " THEN PAR66=1;
IF PAR_66_1="SHE WAS AN INVALID FOR 20 YEARS AND I TOOK CARE OF HER" " THEN PAR66=1;
IF PAR_66_1="SHE WAS MAD" " THEN PAR66=7;
IF PAR_66_1="SISTER RECEIVED MOTHER ESTATE AND RESPONDENT RECEIVED" " THEN PAR66=3;
IF PAR_66_1="STEP SON RECEIVED LESS" " THEN PAR66=4;
IF PAR_66_1="THE BOYS GOT EVERY THING THATS THE WAY IT IS" " THEN PAR66=3;
IF PAR_66_1="THEY PAID OFF A HOUSE FOR THEM" " THEN PAR66=2;
IF PAR_66_1="THEY TOOK CARE OF HER AND WAS LIVIN G THERE" " THEN PAR66=1;
IF PAR_66_1="THOSE CHILDREN WERE NOT CLOSE TO HER. ONE HAD ALREADY PASSE" THEN PAR66=7;
IF PAR_66_1="VARIED BY WHAT WAS OWED TO THE MOTHER BY THE CHILD AND THE " THEN PAR66=1;
IF PAR_66_1="WENT TO YOUNGER BROTHER BECAUSE HE TOOK CARE OF HER" " THEN PAR66=1;
IF PAR_66_1="WOULD LIKE TO KNOW WHY LEFT ONLY TO SON" " THEN PAR66=6;
IF PAR_66_1="YES BECAUSE MY YOUNGER SISTER WAS STILL ATTENDING SCHOOL AN" THEN PAR66=2;
IF PAR_66_1="YOU HAVE TO KNOW THE CHILDREN" " THEN PAR66=8;

```