```
%MACRO SMPEM(DATA=,S=,VAR=,NS=,COV=,NC=,STRATA=,PSU=,WGT=,LOI=,BEG=,END=,LOOP=);
 %DO MA=1 %TO &NC;
   %LET COV&MA=%SCAN(&COV,&MA,' ');

   PROC SORT DATA=&DATA;
     BY &&COV&MA;
   RUN;

   DATA TEMP&MA(KEEP=ID &&COV&MA);
     SET &DATA;
     BY &&COV&MA;
     IF FIRST.&&COV&MA;
   RUN;

   DATA _NULL_;
     SET TEMP&MA END=FINAL;
     N+1;
     IF FINAL THEN CALL SYMPUT("LC&MA", N);  *** LC: LEVEL OF COVARIATES 1,2,... ***;
   RUN;
 %END;
```

This section converts the set of covariate names into macro variables to be used later.

```
 DATA _NULL_;
   IF &NC>1 THEN CALL SYMPUT("CP",TRANSLATE("&COV",'*',' '));
 RUN;

 PROC SORT DATA=&DATA;
   BY ID AGE;
 RUN;

 DATA SEM;
   SET &DATA;
   BY ID AGE;
   IF FIRST.ID THEN DO;
     PREVAGE=.;
     PREVST=.;
     PREVWGT=.;
   END;
   OUTPUT;
   PREVAGE=AGE;
   PREVST=&VAR;
   PREVWGT=&WGT;
   RETAIN PREVAGE PREVST PREVWGT;
 RUN;

 DATA SEM1(DROP=MIDAGE AGE &VAR &WGT J PREVAGE PREVST PREVWGT RENAME=(NAGE=AGE HS=&VAR NWGT=&WG
   SET SEM;
   BY ID;
   IF FIRST.ID THEN DO;
     HS=&VAR;
     NAGE=AGE;
     NWGT=&WGT;
     OUTPUT;
   END;
```

```
  ELSE DO;
    IF &VAR NE &NS THEN DO;
      *IF MOD(GAP,2)=1 THEN MIDAGE=PREVAGE+FLOOR(GAP/2);
      *ELSE MIDAGE=PREVAGE+CEIL(GAP*RANUNI(ROUND(DATETIME())))-1;
      MIDAGE=PREVAGE+CEIL((AGE-PREVAGE)*RANUNI(ROUND(DATETIME())))-1;

      DO J=PREVAGE+1 TO AGE;
        IF J<=MIDAGE THEN DO;
          HS=PREVST;
          NWGT=PREVWGT+ROUND((&WGT-PREVWGT)/(AGE-PREVAGE));
        END;
        ELSE DO;
          HS=&VAR;
          NWGT=&WGT;
        END;
        NAGE=J;
        PREVWGT=NWGT;
        OUTPUT;
      END;
    END;
    ELSE DO;
      IF PREVAGE+1<AGE THEN DO;
        DO J=PREVAGE+1 TO AGE;
          IF J NE AGE THEN DO;
            HS=PREVST;
            NWGT=PREVWGT+ROUND((&WGT-PREVWGT)/(AGE-PREVAGE));
          END;
          ELSE DO;
            HS=&VAR;
            NWGT=&WGT;
          END;
          NAGE=J;
          PREVWGT=NWGT;
          OUTPUT;
        END;
      END;
      ELSE IF PREVAGE+1=AGE THEN DO;
        HS=&VAR;
        NAGE=AGE;
        NWGT=&WGT;
        OUTPUT;
      END;
    END;
  END;
RUN;
```

This and the above blocks convert the input data set, which is in person-year format with one interview observation per line of record, into the interval format with one pair of successive interview per line of record. For surveys with interview gaps wider than 1 year, this section also randomly assigns the occurrence of event, if there is one, to any of intermittent years. For example, if there are 4 years between the 1996 and 2000 interviews, then the event is assumed to occur in 1997, 1998 or 1999, each with a probability of 1/3.

```
DATA TMP0;
  SET SEM1;
  BY ID;

  IF FIRST.ID THEN DO;
    PREVST=.;
    TYEARS=0;
```

```
  END;
  IF PREVST>=0 AND &VAR NE PREVST THEN DO;
    CHANGE=1;
    DUR=TYEARS+1;
    TYEARS=0;
  END;
  ELSE IF PREVST>=0 & &VAR=PREVST THEN DO;
    CHANGE=0;
    TYEARS=TYEARS+1;
  END;
  OUTPUT;
  PREVST=&VAR;
  RETAIN PREVST TYEARS;
RUN;

DATA TMP1;
  SET TMP0;
  BY ID;
  IF CHANGE=1 OR LAST.ID;
  IF DUR=. THEN DUR=TYEARS;
  DAGE=AGE;
  AGE=FLOOR(AGE-DUR);
  RENAME PREVST=BEGST &VAR=ENDST;
RUN;

DATA TMP2;
  SET TMP1;
  BY ID;
  IF BEGST=. THEN DELETE;
  IF FIRST.ID & BEGST=ENDST THEN STYPE='IC';
  ELSE IF FIRST.ID & BEGST NE ENDST THEN STYPE='LC';
  ELSE STYPE='CC';
  DROP TYEARS CHANGE;
RUN;
```

This section rearrange the data set into another format by the episode of events. Each episode is defined as the change from current state to the next or right censoring. This format is useful for analysis of state-duration-dependent transition probabilities.

```
PROC SORT;
  BY ID AGE;
RUN;

DATA PR(KEEP=ID AGE);
  SET SEM1;
  BY ID;
  IF FIRST.ID;
RUN;

PROC SORT;
  BY ID AGE;
RUN;

DATA EPS;
  MERGE TMP2(IN=CHS) PR;
  BY ID AGE;
  IF CHS;
```

```
RUN;

*** CALCULATE THE PREVALENCES ***;
DATA SEM2;
  SET SEM1;
  BY ID;
  IF FIRST.ID;
RUN;

DATA PS;
  DO AGE=50 TO 95;
    CONTROL=1;
    OUTPUT;
  END;
RUN;

DATA PSBACK;
  SET PS SEM2(WHERE=(65<=AGE<=95));
  IF CONTROL=. THEN CONTROL=0;
RUN;

PROC LOGISTIC DATA=PSBACK DESCENDING NOPRINT;
  MODEL &VAR=AGE AGE*AGE;
  OUTPUT OUT=PREVS1 PREDPROBS=I;
RUN;

DATA PREVS1B;
  SET PREVS1;
RUN;

DATA HSPREV;
  SET PREVS1B(WHERE=(CONTROL=1));
  ARRAY P{%EVAL(&NS-1)};
  ARRAY IP{%EVAL(&NS-1)} IP_1-IP_%EVAL(&NS-1);
  DO I=1 TO &NS-1;
    P{I}=IP{I};
  END;
  KEEP AGE P1-P%EVAL(&NS-1);
RUN;
```

This section estimates the prevalence of health states at baseline.

```
DATA CP;
  SET SEM2;
  BY ID;
  IF 65<=AGE<=74 THEN AGE=65;
  ELSE IF 75<=AGE<=84 THEN AGE=75;
  ELSE IF AGE>=85 THEN AGE=85;
RUN;

PROC SORT;
  BY AGE &VAR;
RUN;

PROC FREQ DATA=CP NOPRINT;
```

This sections estimates the prevalence of covariates, by age and health status, at baseline.

```
  BY AGE &VAR;
  %IF &NC>1 %THEN %DO;
    TABLE &CP / OUT=PREVEST OUTPCT;
  %END;
  %ELSE %DO;
    TABLE &COV / OUT=PREVEST OUTPCT;
  %END;
  WEIGHT &WGT;
RUN;

DATA COVPREV;
  SET PREVEST;
  PCT=PERCENT/100;
  KEEP AGE &VAR &COV PCT;
RUN;


DATA LC IC REST;
  SET EPS;
  LENGTH AGE 3;
  BY ID;
  DUR=DAGE-AGE;

  IF STYPE='LC' THEN OUTPUT LC;
  ELSE IF STYPE='IC' THEN OUTPUT IC;
  ELSE OUTPUT REST;
RUN;

PROC SORT DATA=LC;
  BY ID;
RUN;

PROC SORT DATA=IC;
  BY ID;
RUN;

DATA S.BSLE&S;
  TLE=.;
RUN;

%DO M=1 %TO &LOOP;
  %IF &M=1 %THEN %DO;
    DATA CSPELLS;
      SET EPS(DROP=DUR);
      BY ID;
      DDUR=DAGE-AGE;
      IF FIRST.ID THEN DO;
        RMEAN=0;                    *** ASSUME R=0 ***;
        SDUR=RMEAN+DDUR;
      END;
      ELSE DO;
        RMEAN=0;
        SDUR=DDUR;
```

```
    END;
  RUN;

  DATA CSPELLSB;
    SET CSPELLS;
    BY ID;
    DO DUR=1 TO SDUR;
      IF DUR=SDUR AND (BEGST NE ENDST) THEN EVT=1;
      ELSE EVT=0;

      IF DUR>RMEAN THEN OUTPUT;
    END;
  RUN;
%END;
%ELSE %DO;
  DATA CSPELLS;
    %IF &M=&LOOP %THEN %DO;
      SET S.SEM&S;
    %END;
    %ELSE %DO;
      SET SEM&S;
    %END;
    DDUR=DAGE-AGE;
  RUN;

  DATA CSPELLSB;
    SET CSPELLS;

    DO DUR=1 TO DDUR;
      IF DUR=DDUR AND (BEGST NE ENDST) THEN EVT=1;
      ELSE EVT=0;

      IF DUR>RMEAN THEN OUTPUT;
    END;
  RUN;
%END;

DATA PR;
  SET CSPELLS;
  %IF &M=1 %THEN %DO;
    IF 65<=AGE<=69 AND STYPE NE 'CC' AND RMEAN NE .;
  %END;
  %ELSE %DO;
    IF 65<=OAGE<=69 AND STYPE NE 'CC' AND RMEAN NE .;
  %END;
  RMEAN=CEIL(RMEAN);
  IF RMEAN>=20 THEN RMEAN=20;
RUN;

PROC FREQ DATA=PR NOPRINT;
  %IF &NC>1 %THEN %DO;
    TABLE &CP*BEGST*RMEAN / OUT=A2 OUTPCT;
  %END;
```

```
  %ELSE %DO;
    TABLE &COV*BEGST*RMEAN / OUT=A2 OUTPCT;
  %END;
  WEIGHT &WGT;
RUN;

%IF &M>1 %THEN %DO;
  DATA A3;
    SET A2(KEEP=&COV BEGST RMEAN PCT_ROW);
    PCT_ROW=PCT_ROW/100;
    RENAME PCT_ROW=PERCENT;
  RUN;

  PROC SORT;
    BY &COV BEGST;
  RUN;

  DATA TSP;
    %DO MA=1 %TO &NC;
      DO &&COV&MA=1 TO &&LC&MA;
    %END;
        DO BEGST=1 TO &NS-1;
          RMEAN=0;
        END;
    %DO MA=1 %TO &NC;
      END;
    %END;
  RUN;

  PROC SORT;
    BY &COV BEGST;
  RUN;

  DATA B;
    UPDATE A3 TSP;
    BY &COV BEGST;
    IF PERCENT=. THEN PERCENT=1;
  RUN;
%END;
%ELSE %DO;
   DATA B;
    SET A2(KEEP=&COV BEGST RMEAN PCT_ROW);
    PCT_ROW=PCT_ROW/100;
    RENAME PCT_ROW=PERCENT;
  RUN;
%END;

PROC SORT DATA=B OUT=B2 NODUPKEY;
  BY RMEAN;
RUN;

DATA _NULL_;
  SET B2 END=FINAL;
```

```
  IF FINAL THEN CALL SYMPUT('RMAX',RMEAN);
RUN;

%MACRO MODEL(BEG=55,END=150);
  DATA SEST;
    %DO MA=1 %TO &NC;
      DO &&COV&MA=1 TO &&LC&MA;
    %END;
        DO AGE=&BEG-CEIL(&RMAX) TO &END;
          DO BEGST=1 TO &NS-1;
            DO DUR=1 TO 50;
              CONTROL=1;
              OUTPUT;
            END;
          END;
        END;
    %DO MA=1 %TO &NC;
      END;
    %END;
  RUN;

  DATA MODEL;
    SET SEST CSPELLSB;
    IF CONTROL=. THEN CONTROL=0;
    IF EVT=0 THEN ENDST=BEGST;
    KEEP &COV ID DAGE AGE BEGST ENDST DUR &WGT CONTROL;
  RUN;

  PROC LOGISTIC DESCENDING NOPRINT OUTEST=PARB;
    CLASS BEGST &COV;
    MODEL ENDST=AGE BEGST &COV DUR AGE*AGE / L=GLOGIT;
    WEIGHT &WGT;
    OUTPUT OUT=SPROBS PREDPROBS=I;
  RUN;

  DATA PARBB;
    SET PARB;
  RUN;

  DATA TRANPR;
    SET SPROBS(WHERE=(CONTROL=1));
    %DO U=1 %TO &NS;
      P&U=IP_&U;
    %END;
    KEEP &COV AGE DUR BEGST P1-P&NS;
  RUN;

  PROC SORT;
    BY &COV AGE DUR BEGST;
  RUN;
%MEND;

%MODEL;
```

This step estimates the state-duration-dependent transition probabilities. Since an iterative algorithm is used, it is not possible to evaluate the "goodness of fit" of the model before the EM algorithm has converged. My suggestion is to first find the minimum number of iterations needed for convergence plus 5~10 more iterations (min. &M), and then evaluate the HE estimates for different model specifications, each running min. &M iterations. And the &SIMSIZE can also be reduced to minimum to reduce computation time.

```
%INCLUDE 'C:\SPACE\SMPEM_SIMDUR.SAS';
```

```
%IF &M>%EVAL(&LOOP-1) %THEN %DO;
  DATA PLY4;
    COL1=.;
  RUN;

  %DO AGE=65 %TO 75 %BY 10;
    %SIMDUR(AGE=&AGE,SIMSIZE=100000);

    DATA PLY4;
      SET PLY4 PLY3;
      IF COL1=. THEN DELETE;
    RUN;
  %END;
```

In this example, 100K 65- and 75-year olds are simulated. The distribution of their health status, sex and race are estimated from the sample data.

```
  DATA PLY5(KEEP=ID &COV IAGE PR AGE &VAR);
    SET PLY4;
    LENGTH &COV AGE &VAR 3;
    ARRAY COL{100};
    ID=COL1;
    IAGE=COL2;
    %DO MA=1 %TO &NC;
      %LET COV&MA=%SCAN(&COV,&MA,' ');
      &&COV&MA=COL%EVAL(2+&MA);
    %END;
    PR=COL%EVAL(&NC+3);
    DO I=&NC+4 TO 100;
      AGE=IAGE+(I-&NC-4);
      &VAR=COL{I};
      IF &VAR NE 0 THEN OUTPUT;
    END;
  RUN;

  PROC SORT OUT=SIMREC;
    BY IAGE ID AGE;
  RUN;
/*
  DATA SIMEPS;
    SET SIMREC;
    LENGTH PREVST TYEARS CHANGE DUR 3;
    BY IAGE ID;

    IF FIRST.ID THEN DO;
      PREVST=.;
      TYEARS=PR;
    END;
    IF PREVST>=0 & &VAR NE PREVST THEN DO;
      CHANGE=1;
      IF &VAR=&NS THEN DUR=TYEARS+0.5;
      ELSE DUR=TYEARS+1;
      TYEARS=0;
```

Data PLY5 holds the original simulated data in person-year format. Further analysis of life table quantities are all based on rearrangement of this data set.

```
      END;
      ELSE IF PREVST>=0 & &VAR=PREVST THEN DO;
        CHANGE=0;
        TYEARS=TYEARS+1;
      END;
      OUTPUT;
      PREVST=&VAR;
      RETAIN PREVST TYEARS;
    RUN;

    DATA SIMEPS1;
      SET SIMEPS;
      LENGTH DAGE 3;
      BY ID;
      IF DUR=. THEN DUR=TYEARS;
      DAGE=AGE;
      AGE=FLOOR(AGE-DUR);
      IF (CHANGE=1 & DAGE>65) OR (LAST.ID & CHANGE=0 & DAGE>65);
      RENAME PREVST=BEGST &VAR=ENDST DUR=SIMDUR AGE=SAGE;
      KEEP ID DAGE AGE &VAR PREVST DUR;
    RUN;
*/
    DATA SIMREC2;
      SET SIMREC;
      BY IAGE ID;
      RENAME &VAR=ENDST;
      BEGST=LAG(&VAR);
      BAGE=LAG(AGE);
      IF FIRST.ID THEN DO;
        BEGST=0;
        BAGE=0;
        DELETE;
      END;
    RUN;

    DATA PHSC(KEEP=ID IAGE BH DAGE);
      SET SIMREC2;
      BY IAGE ID;

      RETAIN BH;
      IF FIRST.ID THEN BH=BEGST;

      IF LAST.ID THEN DO;
        DAGE=AGE;
        OUTPUT;
      END;
    RUN;

    DATA SIMREC3;
      MERGE SIMREC2(RENAME=(AGE=EAGE)) PHSC;
      BY IAGE ID;

      IF FIRST.ID THEN DO;
```

Data SIMEPS1 transforms PLY5 into episode format.

```
      YRLVED=0;
      YRINAH=0;
      YRINAD=0;
    END;

    IF BEGST=1 THEN DO;
      IF ENDST IN (1,&NS) THEN AH=1;
      ELSE IF ENDST=2 THEN DO;
        AH=0.5;
        AD=0.5;
      END;
    END;
    ELSE IF BEGST=2 THEN DO;
      IF ENDST IN (2,&NS) THEN AD=1;
      ELSE IF ENDST=1 THEN DO;
        AH=0.5;
        AD=0.5;
      END;
    END;

    YRLVED+1;
    YRINAH+AH;
    YRINAD+AD;

    IF LAST.ID THEN OUTPUT;
  RUN;

  PROC MEANS DATA=SIMREC3 MEAN P25 P50 P75 NOPRINT;
    CLASS IAGE &COV BH;
    VAR YRLVED YRINAH YRINAD;
    OUTPUT OUT=MEDLE MEAN=TLE ALE DLE
                     P25=TLY25 ALY25 DLY25
                     P50=TLY50 ALY50 DLY50
                     P75=TLY75 ALY75 DLY75;
  RUN;

  DATA MEDLE2;
    SET MEDLE;
    ITR=&M;
    IF BH=. THEN BH=0;
    RENAME BH=STATE;
    FORMAT TLE ALE DLE 5.1;
    DROP _TYPE_;
  RUN;

  DATA S.BSLE&S;
    SET S.BSLE&S MEDLE2;
    IF TLE NE .;
  RUN;

  PROC SORT DATA=S.BSLE&S;
    BY &COV IAGE STATE;
  RUN;
```

```
%END;

***** THIS PART SIMULATES THE HISTORY FOR 55-YR OLD COHORT *****;

%SIMDUR(AGE=55,SIMSIZE=10000);
```

```
DATA SIMREC(KEEP=ID &COV SAGE PR AGE &VAR);
  SET PLY3;
  LENGTH &COV SAGE PR AGE &VAR 3;
  IF COL15 IN (0,&NS) THEN DELETE;  *** COL15=AGE 65 ***;
  ARRAY COL{100};
  ID=COL1;
  SAGE=COL2;
  %DO MA=1 %TO &NC;
    %LET COV&MA=%SCAN(&COV,&MA,' ');
    &&COV&MA=COL%EVAL(2+&MA);
  %END;
  PR=COL%EVAL(3+&NC);
  DO I=3+&NC+1 TO 100;
    AGE=SAGE+(I-(3+&NC+1));
    &VAR=COL{I};
    IF &VAR NE 0 THEN OUTPUT;
  END;
RUN;

PROC SORT;
  BY ID AGE;
RUN;

DATA TEMP0;
  SET SIMREC;
  BY ID;

  IF FIRST.ID THEN DO;
    PREVST=.;
    TYEARS=PR;
  END;
  IF PREVST>=0 & &VAR NE PREVST THEN DO;
    CHANGE=1;
    IF &VAR=&NS THEN DUR=TYEARS+0.5;
    ELSE DUR=TYEARS+1;
    TYEARS=0;
  END;
  ELSE IF PREVST>=0 & &VAR=PREVST THEN DO;
    CHANGE=0;
    TYEARS=TYEARS+1;
  END;
  OUTPUT;
  PREVST=&VAR;
  RETAIN PREVST TYEARS;
RUN;

DATA TEMP1;
```

```
  SET TEMP0;
  BY ID;
  IF CHANGE=1 OR (LAST.ID & CHANGE=0);
  IF DUR=. THEN DUR=TYEARS;
  DAGE=AGE;
  AGE=FLOOR(AGE-DUR);
  RENAME PREVST=BEGST &VAR=ENDST DUR=SIMDUR AGE=SAGE;
  KEEP &COV DAGE AGE &VAR PREVST DUR;
RUN;


*** IMPUTE FOR LC ONLY ***;
PROC SQL;
  CREATE TABLE LCIMP AS
  SELECT A.ID, B.SIMDUR LENGTH=3
  FROM LC AS A, TEMP1 AS B
  WHERE %DO MA=1 %TO &NC; A.&&COV&MA=B.&&COV&MA & %END; A.DAGE=B.DAGE
        & A.ENDST=B.ENDST & A.BEGST=B.BEGST & A.DUR<=B.SIMDUR;
QUIT;

PROC SORT DATA=LCIMP;
  BY ID;
RUN;

DATA LCIMP2;
  SET LCIMP;
  BY ID;
  RETAIN NS 0;
  IF FIRST.ID THEN DO;
    N=0;
    NS=NS+1;
  END;
  N+1;
  IF LAST.ID THEN OUTPUT;
RUN;

DATA LCIMP3;
  MERGE LCIMP LCIMP2;
  LENGTH SIMDUR NS N 3;
  BY ID;
RUN;

PROC IML;
  USE LCIMP3;
  READ ALL VAR {ID SIMDUR} INTO X;
  CLOSE LCIMP3;

  USE LCIMP2;
  READ ALL VAR {N} INTO Z;
  CLOSE LCIMP2;

  CUMS=0;
  LC4=J(NROW(Z),2,0);
```

```
  DO I=1 TO NROW(Z);
    CUMS=CUMS+Z[I];
    RS=X[CUMS-Z[I]+1:CUMS,];
    LC4[I,]=RS[ROUND(RANUNI(ROUND(DATETIME()))*Z[I]+0.5),];
  END;

  VAR={"ID" "RMEAN"};
  CREATE LCIMP4 FROM LC4 [COLNAME=VAR];
  APPEND FROM LC4;
  CLOSE LCIMP4;

  FREE /;
QUIT;

DATA LCIMP6;
  UPDATE LC LCIMP4;
  BY ID;
  IF RMEAN NE . THEN AUGDUR=RMEAN;
  ELSE AUGDUR=DUR;
  IF RMEAN NE . THEN RMEAN=AUGDUR-DUR;  *** !!! ***;
  OAGE=AGE;
  AGE=DAGE-CEIL(AUGDUR);
RUN;

*** IMPUTE FOR IC ***;
PROC SQL;
  CREATE TABLE IC2 AS
  SELECT A.ID %DO MA=1 %TO &NC;, A.&&COV&MA %END; LENGTH=3, A.AGE LENGTH=3,
         A.AGE-B.SAGE AS RMEAN LENGTH=3
  FROM IC(KEEP=ID &COV AGE DAGE BEGST) AS A, TEMP1(RENAME=(DAGE=EAGE)) AS B
  WHERE %DO MA=1 %TO &NC; A.&&COV&MA=B.&&COV&MA & %END; A.BEGST=B.BEGST
        & B.SAGE<=A.AGE & B.EAGE>A.DAGE;
QUIT;

PROC SORT DATA=IC2;
  BY ID;
RUN;

DATA IC3;
  SET IC2;
  BY ID;
  IF FIRST.ID THEN N=0;
  N+1;
  IF LAST.ID THEN DO;
    IF AGE<=74 THEN AGEGRP=1;
    ELSE IF AGE<=84 THEN AGEGRP=2;
    ELSE IF AGE>=85 THEN AGEGRP=3;
    OUTPUT;
  END;
RUN;

PROC SORT DATA=IC3 OUT=IC3B NODUPKEY;
  BY AGEGRP;
```

```
RUN;

DATA _NULL_;
  SET IC3B END=EOF;
  IF EOF THEN CALL SYMPUT('NA',AGEGRP);
RUN;

DATA IC4;
  MERGE IC2 IC3;
  BY ID;
  LENGTH RMEAN N AGEGRP 3;
  DROP AGE;
RUN;

DATA IC6;
  ID=.;
  RMEAN=.;
RUN;

%DO I=1 %TO &NA;
  DATA _NULL_;
    SET IC3(WHERE=(AGEGRP=&I));
    BY ID;
    IF LAST.ID THEN CALL SYMPUT('NP',_N_);
  RUN;

  PROC IML;
    USE IC3 WHERE (AGEGRP=&I);
    READ ALL VAR {N} INTO ZIR;

    USE IC4 WHERE (AGEGRP=&I);
    READ ALL VAR {ID RMEAN} INTO XIR;

    CUMS=0;
    IC5=J(&NP,2,0);
    DO K=1 TO &NP;
      CUMS=CUMS+ZIR[K];
      RS=XIR[CUMS-ZIR[K]+1:CUMS,];
      IC5[K,]=RS[ROUND(RANUNI(ROUND(DATETIME()))*ZIR[K]+0.5),];
    END;

    VAR={"ID" "RMEAN"};
    CREATE IC5&I FROM IC5 [COLNAME=VAR];
    APPEND FROM IC5;
    CLOSE IC5&I;

    FREE /;
  QUIT;

  DATA IC6;
    SET IC6 IC5&I;
    IF ID=. THEN DELETE;
  RUN;
```

```
    %END;

    PROC SORT;
      BY ID;
    RUN;

    DATA ICIMP;
      UPDATE IC IC6;
      BY ID;
      AUGDUR=DUR+RMEAN;
      IF RMEAN=. THEN AUGDUR=DUR;
      OAGE=AGE;
      AGE=DAGE-CEIL(AUGDUR);
    RUN;


    DATA REST1;
      SET REST;
      AUGDUR=DUR;
      RMEAN=.;
      OAGE=AGE;
    RUN;

    *** COMBINE IMPUTED SPELLS WITH COMPLETE SPELLLS ***;
    %IF &M=%EVAL(&LOOP-1) %THEN %DO;
      DATA S.SEM&S;
    %END;
    %ELSE %DO;
      DATA SEM&S;
    %END;
      SET LCIMP6 ICIMP REST1;
      RENAME DUR=ODUR;
      ITER=&M;
    RUN;

    PROC SORT;
      BY ID AGE;
    RUN;
  %END;
%MEND;
```