

# Appendix 5: Income and Assets Variable Creation

## Contents

- Household Income and Assets
  - Section 1: Gross Household Income
  - Section 2: Household Net Worth
  - Section 3: Household Poverty Status
  - Section 4: Topcoding Income, Net Worth, and Poverty Ratio
- Participation in Government Programs
  - Section 1: Unemployment Compensation
  - Section 2: Workers Compensation
  - Section 3: Other Government Assistance

## Household Income and Assets

### Variables Created:

- CV\_HH\_NET\_WORTH\_Y
- CV\_INCOME\_GROSS\_YR
- CV\_HH\_POV\_RATIO

### Variables Used

Name in Program	Question Name on CD	Name in Program	Question Name on CD
Naamt241-Naamt252	AFDC_AMT_2000.01-.12	YAS4400	YAST-4400
Noamt241-Noamt252	OTHER_AMT_2000.01-.12	YAS4420	YAST-4420
Nuiamt241-Nuiamt252	UneMP_AMT_2000.01-.12	YAS4422	YAST-4422
Nwcamt241	WKCOMP_AMT_2000.01-.03	YAS4426	YAST-4426
YI_1400-YI_2700	YINC-1400 - YINC-2700	YAS4530	YAST-4530
YI_2900-YI_3100	YINC-2900 - YINC-3100	YAS4550	YAST-4550
YI_3300-YI_5400	YINC-3300 - YINC-5400	YAS4552	YAST-4552
YI_7600-YI_7800	YINC-7600 - YINC-7800	YAS4556	YAST-4556
YI_8700-YI_8900	YINC-8700 - YINC-8900	YAS4660	YAST-4660
YI_9200-YI_9400	YINC-9200 - YINC-9400	YAS4680	YAST-4680
YI_9600-YI_9900	YINC-9600 - YINC-9900	YAS4682	YAST-4682
YI_10100-YI_10400	YINC-10100 - YINC-10400	YAS4686	YAST-4686
YI_10800	YINC-10800	YAS4790	YAST-4790
I1110001-I1110015	YINC-11100.01-.15	YAS4810	YAST-4810
I1160001-I1160013	YINC-11600.01-.13	YAS4812	YAST-4812
I1170001-I1170013	YINC-11700.01-.13	YAS4816	YAST-4816
YAS50	YAST-050	YAS4840	YAST-4840
YAS1400	YAST-1400	YAS4842	YAST-4842
YAS1500	YAST-1500	YAS4846	YAST-4846
YAS1610	YAST-1610	YAS4870	YAST-4870
YAS1860	YAST-1860	YAS4880	YAST-4880
YAS1862	YAST-1862	YAS4900	YAST-4900
YAS1866	YAST-1866	YAS4902	YAST-4902
YAS2520	YAST-2520	YAS4906	YAST-4906
YAS2550	YAST-2550	YAS5040	YAST-5040

YAS2552	YAST-2552	YAS5060	YAST-5060
YAS2556	YAST-2556	YAS5062	YAST-5062
YAS2760	YAST-2760	A506601-A506620	YAST-5066.01-.20
YAS2762	YAST-2762	A507401-A507420	YAST-5074.01-.20
YAS2766	YAST-2766	A508001-A508020	YAST-5080.01-.20
YAS3310	YAST-3310	A508201-A508220	YAST-5082.01-.20
YAS3380	YAST-3380	YAS5130	YAST-5130
YAS3382	YAST-3382	YAS5150	YAST-5150
YAS3386	YAST-3386	YAS5152	YAST-5152
YAS3740	YAST-3740	A516401-A516420	YAST-5164.01-.20
YAS3750	YAST-3750	A517001-A517020	YAST-5170.01-.20
YAS3752	YAST-3752	A517201-A517220	YAST-5172.01-.20
YAS3756	YAST-3756	YAS5210	YAST-5210
YAS3790	YAST-3790	YAS5220	YAST-5220
YAS3810	YAST-3810	YAS5222	YAST-5222
YAS3812	YAST-3812	YAS5226	YAST-5226
YAS3840	YAST-3840	HIRELY01-HIRELY16HHI_REL.Y.01-.16	
YAS3860	YAST-3860	PUBID	PUBID
YAS3880	YAST-3880	YAS1864L, U	YAST-1864~000001, ~000002
YAS3882	YAST-3882	YAS2554L, U	YAST-2554~000001, ~000002
YAS3910	YAST-3910	YAS2764L, U	YAST-2764~000001, ~000002
YAS3920	YAST-3920	YAS3384L, U	YAST-3384~000001, ~000002
YAS3922	YAST-3922	YAS3754L, U	YAST-3754~000001, ~000002
YAS3950	YAST-3950	YAS3924L, U	YAST-3924~000001, ~000002
YAS3960	YAST-3960	YAS3964L, U	YAST-3964~000001, ~000002
YAS3962	YAST-3962	YAS4034L, U	YAST-4034~000001, ~000002
YAS3966	YAST-3966	YAS4164L, U	YAST-4164~000001, ~000002
YAS4010	YAST-4010	YAS4294L, U	YAST-4294~000001, ~000002
YAS4030	YAST-4030	YAS4424L, U	YAST-4424~000001, ~000002
YAS4032	YAST-4032	YAS4554L, U	YAST-4554~000001, ~000002
YAS4036	YAST-4036	YAS4684L, U	YAST-4684~000001, ~000002
YAS4140	YAST-4140	YAS4814L, U	YAST-4814~000001, ~000002
YAS4160	YAST-4160	YAS4844L, U	YAST-4844~000001, ~000002
YAS4162	YAST-4162	YAS4904L, U	YAST-4904~000001, ~000002
YAS4166	YAST-4166	YAS5224L, U	YAST-5224~000001, ~000002
YAS4270	YAST-4270	YAS25	YAST-025
YAS4290	YAST-4290	YAS60	YAST-060
YAS4292	YAST-4292	YAS70	YAST-070
YAS4296	YAST-4296	YAS105	YAST-105

This program creates the household net worth and gross household income variables. The household net worth variable is an actual number that results from adding the values of all assets and subtracting liabilities of the household. The gross household income variable includes total annual cash receipts before taxes from all sources. The program then creates a ratio comparing the household's total income to federal poverty guidelines based on the number of household residents and the number of members under age 18.

Researchers should note that, like many income and asset variables in the data set, these three variables are topcoded to protect respondent privacy. More information about topcoding is available in the *NLSY97 User's Guide*.

---

/\*\*\*/SECTION 1: GROSS HOUSEHOLD INCOME/\*\*\*/

/\*\* In income section, I flag three items involving most of the respondents:

- I. nonfarm employment income -- flag1400
- II. father's income -- flag8700
- III. mother's income -- flag9200

flag=0 indicating accurate income reported  
flag=1 indicating estimated income reported \*\*/

/\*Create the following components of the hh net worth and gross income, such as income from wages and salaries, interests, etc. \*\*/

/\*\* Income from non-farm employment (nfarmwgY) \*\*/

```
nfarmwgY=-4; flag1400=0;
if (YI_1400=1 and (YI_1700 ne -1 and YI_1700 ne -2 and YI_1700 ne -3)) or (YI_1400=-1 and
YI_1600=1 and (YI_1700 ne -1 and YI_1700 ne -2 and YI_1700 ne -3)) or (YI_1400=-2 and
YI_1500=1 and (YI_1700 ne -1 and YI_1700 ne -2 and YI_1700 ne -3)) or (YI_1400=-2 and YI_1500=-
1 and YI_1600=1 and (YI_1700 ne -1 and YI_1700 ne -2 and YI_1700 ne -3))
then nfarmwgY=YI_1700;
if (YI_1400=1 or (YI_1400=-1 and YI_1600=1) or (YI_1400=-2 and YI_1500=1) or (YI_1400=-2 and
YI_1500=-1 and YI_1600=1) or (YI_1400=-2 and YI_1500=-3 and YI_1600=1) or (YI_1400=-3 and
YI_1600=1) or (YI_1400=-3 and YI_1500=1)) and (YI_1700 eq -1 or YI_1700 eq -2 or YI_1700 eq -3)
then do;
  if YI_1800=1 then do; nfarmwgY=2500; flag1400=1; end;
  if YI_1800=2 then do; nfarmwgY=7500; flag1400=1; end;
  if YI_1800=3 then do; nfarmwgY=17500; flag1400=1; end;
  if YI_1800=4 then do; nfarmwgY=37500; flag1400=1; end;
  if YI_1800=5 then do; nfarmwgY=75000; flag1400=1; end;
  if YI_1800=6 then do; nfarmwgY=175000; flag1400=1; end;
  if YI_1800=7 then do; nfarmwgY=250001; flag1400=1; end;
end;
if YI_1400=0 or YI_1500=0 or YI_1600=0 then nfarmwgY=0;
if YI_1600=-1 or YI_1800=-1 then nfarmwgY=-1;
if YI_1500=-2 or YI_1800=-2 then nfarmwgY=-2;
if YI_1600=-3 or YI_1500=-3 or YI_1800=-3 then nfarmwgY=-3;
```

/\*\* For all the questions below, the youth must be INDEPENDENT (YI\_1900=1) \*\*/

/\*\* Income from farm and self-employment (farmwgY) \*\*/

```
farmwgY=-4;
if YI_2000=1 and YI_2100 ne -1 and YI_2100 ne -2 and YI_2100 ne -3 then farmwgY=YI_2100;
if YI_2000=1 and (YI_2100 eq -1 or YI_2100 eq -2 or YI_2100 eq -3) then do;
  if YI_2200=1 then do; farmwgY=-2; end;
  if YI_2200=2 then do; farmwgY=2500; end;
  if YI_2200=3 then do; farmwgY=7500; end;
  if YI_2200=4 then do; farmwgY=17500; end;
  if YI_2200=5 then do; farmwgY=37500; end;
  if YI_2200=6 then do; farmwgY=75000; end;
  if YI_2200=7 then do; farmwgY=175000; end;
  if YI_2200=8 then do; farmwgY=250001; end;
end;
```

```

if YI_2000=0 then farmwgY=0;
if YI_2000=-1 or YI_2200=-1 then farmwgY=-1;
if YI_2000=-2 or YI_2200=-2 then farmwgY=-2;
if YI_2000=-3 or YI_2200=-3 then farmwgY=-3;

/** Non-farm income from the spouse/partner (nfarmwgP and farmwgP) **/
nfarmwgP=-4;
if YI_2300=1 then do;
  if (YI_2400=1 and YI_2600 ne -1 and YI_2600 ne -2 and YI_2600 ne -3) or (YI_2400=-1 and
  YI_2500=1 and YI_2600 ne -1 and YI_2600 ne -2 and YI_2600 ne -3)
  then nfarmwgP=YI_2600;
  if (YI_2400=1 and (YI_2600 eq -1 or YI_2600 eq -2 or YI_2600 eq -3)) or (YI_2400=-1 and
  YI_2500=1 and (YI_2600 eq -1 or YI_2600 eq -2 or YI_2600 eq -3)) then do;
    if YI_2700=1 then do; nfarmwgP=2500; end;
    if YI_2700=2 then do; nfarmwgP=7500; end;
    if YI_2700=3 then do; nfarmwgP=17500; end;
    if YI_2700=4 then do; nfarmwgP=37500; end;
    if YI_2700=5 then do; nfarmwgP=75000; end;
    if YI_2700=6 then do; nfarmwgP=175000; end;
    if YI_2700=7 then do; nfarmwgP=250001; end;
  end;
end;
if YI_2300=0 or YI_2400=0 or YI_2500=0 then nfarmwgP=0;
if YI_2300=-1 or YI_2500=-1 or YI_2700=-1 then nfarmwgP=-1;
if YI_2300=-2 or YI_2400=-2 or YI_2500=-2 or YI_2700=-2 then nfarmwgP=-2;
if YI_2300=-3 or YI_2400=-3 or YI_2500=-3 or YI_2700=-3 then nfarmwgP=-3;

/** Farm-related income from the spouse/partner      **/
farmwgP=-4;
if YI_2300=1 then do;
  if YI_2900=1 and YI_3000 ne -1 and YI_3000 ne -2 and YI_3000 ne -3 then farmwgP=YI_3000;
  if YI_2900=1 and (YI_3000 eq -1 or YI_3000 eq -2 or YI_3000 eq -3) then do;
    if YI_3100=1 then do; farmwgP=-2; end;
    if YI_3100=2 then do; farmwgP=2500; end;
    if YI_3100=3 then do; farmwgP=7500; end;
    if YI_3100=4 then do; farmwgP=17500; end;
    if YI_3100=5 then do; farmwgP=37500; end;
    if YI_3100=6 then do; farmwgP=75000; end;
    if YI_3100=7 then do; farmwgP=175000; end;
    if YI_3100=8 then do; farmwgP=250001; end;
  end;
end;
if YI_2300=0 or YI_2900=0 then farmwgP=0;
if YI_2300=-1 or YI_2900=-1 or YI_3100=-1 then farmwgP=-1;
if YI_2300=-2 or YI_2900=-2 or YI_3100=-2 then farmwgP=-2;
if YI_2300=-3 or YI_2900=-3 or YI_3100=-3 then farmwgP=-3;

/** Child support payment (chiltsuY) **/
chiltsuY=-4;
if YI_3300=1 then do;
  if (YI_3900=1 and YI_4000=1 and (YI_4100 ne -1 and YI_4100 ne -2 and YI_4100 ne -3)) then

```

```

childsuY=YI_4100;
  if (YI_3900=1 and YI_4000=1 and (YI_4100 eq -1 or YI_4100 eq -2 or YI_4100 eq -3)) then do;
    if YI_4200=1 then do; childsuY=500; end;
    if YI_4200=2 then do; childsuY=1750; end;
    if YI_4200=3 then do; childsuY=3750; end;
    if YI_4200=4 then do; childsuY=7500; end;
    if YI_4200=5 then do; childsuY=17500; end;
    if YI_4200=6 then do; childsuY=37500; end;
    if YI_4200=7 then do; childsuY=50001; end;
  end;
end;
if YI_3300=0 or YI_3900=0 or YI_4000=0 then childsuY=0;
if YI_3300=-1 or YI_3900=-1 or YI_4000=-1 or YI_4200=-1 then childsuY=-1;
if YI_3300=-2 or YI_3900=-2 or YI_4000=-2 or YI_4200=-2 then childsuY=-2;
if YI_3300=-1 or YI_3900=-1 or YI_4000=-3 or YI_4100=-3 or YI_4200=-3 then childsuY=-3;

```

/\*\* Interest payments received by the youth (Y) and his/her partner/spouse. \*\*/

```

interesY=-4;
if YI_4300=1 then do;
  if (YI_4400 ne -1 and YI_4400 ne -2 and YI_4400 ne -3) then interesY=YI_4400;
  if (YI_4400 eq -1 or YI_4400 eq -2 or YI_4400 eq -3) then do;
    if YI_4500=1 then do; interesY=250; end;
    if YI_4500=2 then do; interesY=750; end;
    if YI_4500=3 then do; interesY=1750; end;
    if YI_4500=4 then do; interesY=3750; end;
    if YI_4500=5 then do; interesY=6250; end;
    if YI_4500=6 then do; interesY=8750; end;
    if YI_4500=7 then do; interesY=10001; end;
  end;
end;
if YI_4300=0 then interesY=0;
if YI_4300=-1 or YI_4500=-1 then interesY=-1;
if YI_4300=-2 or YI_4500=-2 then interesY=-2;
if YI_4300=-3 or YI_4400=-3 or YI_4500=-3 then interesY=-3;

```

/\*\* Dividends from stocks and mutual funds. \*\*/

```

dividend=-4;
if YI_4600=1 then do;
  if (YI_4700 ne -1 and YI_4700 ne -2 and YI_4700 ne -3) then dividend=YI_4700;
  if (YI_4700 eq -1 or YI_4700 eq -2 or YI_4700 eq -3) then do;
    if YI_4800=1 then do; dividend=250; end;
    if YI_4800=2 then do; dividend=750; end;
    if YI_4800=3 then do; dividend=1750; end;
    if YI_4800=4 then do; dividend=3750; end;
    if YI_4800=5 then do; dividend=6250; end;
    if YI_4800=6 then do; dividend=8750; end;
    if YI_4800=7 then do; dividend=10001; end;
  end;
end;
if YI_4600=0 then dividend=0;
if YI_4600=-1 or YI_4800=-1 then dividend=-1;

```

```
if YI_4600=-2 or YI_4800=-2 then dividend=-2;
if YI_4600=-3 or YI_4700=-3 or YI_4800=-3 then dividend=-3;
```

```
/** Rental income.      **/
```

```
rentalIY=-4;
if YI_4900=1 then do;
  if (YI_5000 ne -1 and YI_5000 ne -2 and YI_5000 ne -3) then rentalIY=YI_5000;
  if (YI_5000 eq -1 or YI_5000 eq -2 or YI_5000 eq -3) then do;
    if YI_5100=1 then do; rentalIY=500; end;
    if YI_5100=2 then do; rentalIY=1750; end;
    if YI_5100=3 then do; rentalIY=3750; end;
    if YI_5100=4 then do; rentalIY=7500; end;
    if YI_5100=5 then do; rentalIY=17500; end;
    if YI_5100=6 then do; rentalIY=37500; end;
    if YI_5100=7 then do; rentalIY=50001; end;
  end;
end;
if YI_4900=0 then rentalIY=0;
if YI_4900=-1 or YI_5100=-1 then rentalIY=-1;
if YI_4900=-2 or YI_5100=-2 then rentalIY=-2;
if YI_4900=-3 or YI_5000=-3 or YI_5100=-3 then rentalIY=-3;
```

```
/** Property or money from estates, trusts, annuities or inheritances.      **/
```

```
estatesY=-4;
if YI_5200=1 then do;
  if (YI_5300 ne -1 and YI_5300 ne -2 and YI_5300 ne -3) then estatesY=YI_5300;
  if (YI_5300 eq -1 or YI_5300 eq -2 or YI_5300 eq -3) then do;
    if YI_5400=1 then do; estatesY=2500; end;
    if YI_5400=2 then do; estatesY=7500; end;
    if YI_5400=3 then do; estatesY=17500; end;
    if YI_5400=4 then do; estatesY=37500; end;
    if YI_5400=5 then do; estatesY=75000; end;
    if YI_5400=6 then do; estatesY=175000; end;
    if YI_5400=7 then do; estatesY=250001; end;
  end;
end;
if YI_5200=0 then estatesY=0;
if YI_5200=-1 or YI_5400=-1 then estatesY=-1;
if YI_5200=-2 or YI_5400=-2 then estatesY=-2;
if YI_5200=-3 or YI_5300=-3 or YI_5400=-3 then estatesY=-3;
```

```
/** Income from other sources: SS payments, pension and retirement income, alimony, payments from
insurance policies, etc...      **/
```

```
pensionY=-4;
if YI_7600=1 then do;
  if (YI_7700 ne -1 and YI_7700 ne -2 and YI_7700 ne -3) then pensionY=YI_7700;
  if (YI_7700 eq -1 or YI_7700 eq -2 or YI_7700 eq -3) then do;
    if YI_7800=1 then do; pensionY=500; end;
    if YI_7800=2 then do; pensionY=1750; end;
    if YI_7800=3 then do; pensionY=3750; end;
    if YI_7800=4 then do; pensionY=7500; end;
  end;
end;
```

```

    if YI_7800=5 then do; pensionY=17500; end;
    if YI_7800=6 then do; pensionY=37500; end;
    if YI_7800=7 then do; pensionY=50001; end;
end;
end;
if YI_7600=0 then pensionY=0;
if YI_7600=-1 or YI_7800=-1 then pensionY=-1;
if YI_7600=-2 or YI_7800=-2 then pensionY=-2;
if YI_7600=-3 or YI_7700=-3 or YI_7800=-3 then pensionY=-3;

/** If the youth lived with the father, the father's income.    **/
flag8700=0; faincome=-4;
/** identifying the father is in the household.    **/
if (HIRELY01=4 or HIRELY02=4 or HIRELY03=4 or HIRELY04=4 or HIRELY05=4 or
HIRELY06=4 or HIRELY07=4 or HIRELY08=4 or HIRELY09=4 or HIRELY10=4 or HIRELY11=4
or HIRELY12=4 or HIRELY13=4 or HIRELY14=4) then do;
    if YI_8700=1 then do;
        if (YI_8800 ne -1 and YI_8800 ne -2 and YI_8800 ne -3) then faincome=YI_8800;
        if (YI_8800 eq -1 or YI_8800 eq -2 or YI_8800 eq -3) then do;
            if YI_8900=1 then do; faincome=2500; flag8700=1; end;
            if YI_8900=2 then do; faincome=7500; flag8700=1; end;
            if YI_8900=3 then do; faincome=17500; flag8700=1; end;
            if YI_8900=4 then do; faincome=37500; flag8700=1; end;
            if YI_8900=5 then do; faincome=75000; flag8700=1; end;
            if YI_8900=6 then do; faincome=175000; flag8700=1; end;
            if YI_8900=7 then do; faincome=250001; flag8700=1; end;
        end;
    end;
    if YI_8700=0 then faincome=0;
    if YI_8700=-1 or YI_8900=-1 then faincome=-1;
    if YI_8700=-2 or YI_8900=-2 then faincome=-2;
    if YI_8700=-3 or YI_8800=-3 or YI_8900=-3 then faincome=-3;
end;

/** If the youth lived with the mother, the mother's income.    **/
flag9200=0;
maincome=-4;
/** identify the mother is in the household.    **/
if (HIRELY01=3 or HIRELY02=3 or HIRELY03=3 or HIRELY04=3 or HIRELY05=3 or
HIRELY06=3 or HIRELY07=3 or HIRELY08=3 or HIRELY09=3 or HIRELY10=3 or HIRELY11=3
or HIRELY12=3 or HIRELY13=3 or HIRELY14=3) then do;
    if YI_9200=1 then do;
        if (YI_9300 ne -1 and YI_9300 ne -2 and YI_9300 ne -3) then maincome=YI_9300;
        if (YI_9300 eq -1 or YI_9300 eq -2 or YI_9300 eq -3) then do;
            if YI_9400=1 then do; maincome=2500; flag9200=1; end;
            if YI_9400=2 then do; maincome=7500; flag9200=1; end;
            if YI_9400=3 then do; maincome=17500; flag9200=1; end;
            if YI_9400=4 then do; maincome=37500; flag9200=1; end;
            if YI_9400=5 then do; maincome=75000; flag9200=1; end;
            if YI_9400=6 then do; maincome=175000; flag9200=1; end;
            if YI_9400=7 then do; maincome=250001; flag9200=1; end;
        end;
    end;
end;

```

```

end;
end;
if YI_9200=0 then maincome=0;
if YI_9200=-1 or YI_9400=-1 then maincome=-1;
if YI_9200=-2 or YI_9400=-2 then maincome=-2;
if YI_9200=-3 or YI_9400=-3 then maincome=-3;
end;

/** If the youth lives with the male guardian. **/
mgincome=-4;
if YI_9600=1 then do;
  if (YI_9700=1 and (YI_9800 ne -1 and YI_9800 ne -2 and YI_9800 ne -3)) then mgincome=YI_9800;
  if (YI_9700=1 and (YI_9800 eq -1 or YI_9800 eq -2 or YI_9800 eq -3)) then do;
    if YI_9900=1 then do; mgincome=2500; end;
    if YI_9900=2 then do; mgincome=7500; end;
    if YI_9900=3 then do; mgincome=17500; end;
    if YI_9900=4 then do; mgincome=37500; end;
    if YI_9900=5 then do; mgincome=75000; end;
    if YI_9900=6 then do; mgincome=175000; end;
    if YI_9900=7 then do; mgincome=250001; end;
  end;
end;
if YI_9700=0 then mgincome=0;
if YI_9600=-1 or YI_9700=-1 or YI_9900=-1 then mgincome=-1;
if YI_9600=-2 or YI_9700=-2 or YI_9900=-2 then mgincome=-2;
if YI_9600=-3 or YI_9700=-3 or YI_9800=-3 or YI_9900=-3 then mgincome=-3;

/** If the youth lives with female guardian. **/
fgincome=-4;
if YI_10100=1 then do;
  if (YI_10200=1 and (YI_10300 ne -1 and YI_10300 ne -2 and YI_10300 ne -3)) then
fgincome=YI_10300;
  if (YI_10100=1 and YI_10200=1 and (YI_10300 eq -1 or YI_10300 eq -2 or YI_10300 eq -3)) then
do;
  if YI_10400=1 then do; fgincome=2500; end;
  if YI_10400=2 then do; fgincome=7500; end;
  if YI_10400=3 then do; fgincome=17500; end;
  if YI_10400=4 then do; fgincome=37500; end;
  if YI_10400=5 then do; fgincome=75000; end;
  if YI_10400=6 then do; fgincome=175000; end;
  if YI_10400=7 then do; fgincome=250001; end;
end;
end;
if YI_10200=0 then fgincome=0;
if YI_10100=-1 or YI_10200=-1 or YI_10400=-1 then fgincome=-1;
if YI_10100=-2 or YI_10200=-2 or YI_10400=-2 then fgincome=-2;
if YI_10100=-3 or YI_10200=-3 or YI_10300=-3 or YI_10400=-3 then fgincome=-3;

/** Income from household members of 14 or older other than spouse/partner. **/
array offamI offamI01-otfamI14;
array I11100 I1110001-I1110014;

```



```

array I11600 I1160001-I1160014;
array I11700 I1170001-I1170014;
do I=1 to 14;
  otfamI(I)=-4;
  if YI_10800=1 then do;
    if I11100(I)=0 and I11600(I) ne -1 and I11600(I) ne -2 and I11600(I) ne -3 and I11600(I) ne -4 then
otfamI(I)=I11600(I);
    if I11100(I)=0 and (I11600(I) eq -1 or I11600(I) eq -2 or I11600(I) eq -3) then do;
      if I11700(I)=1 then do; otfamI(I)=2500; end;
      if I11700(I)=2 then do; otfamI(I)=7500; end;
      if I11700(I)=3 then do; otfamI(I)=17500; end;
      if I11700(I)=4 then do; otfamI(I)=37500; end;
      if I11700(I)=5 then do; otfamI(I)=75000; end;
      if I11700(I)=6 then do; otfamI(I)=175000; end;
      if I11700(I)=7 then do; otfamI(I)=250001; end;
    end;
    if I11100(I)=-1 or I11700(I)=-1 then otfamI(I)=-1;
    if I11100(I)=-2 or I11700(I)=-2 then otfamI(I)=-2;
    if I11100(I)=-3 or I11600(I)=-3 or I11700(I)=-3 then otfamI(I)=-3;
  end;
end;

```

```

/** We now create gross hh income according to the youth.      **/

```

```

groshhIY=0;
if YI_1900=1 then do;
  if nfarmwgY NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+nfarmwgY;
  if farmwgY NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+farmwgY;
  if nfarmwgP NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+nfarmwgP;
  if farmwgP NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+farmwgP;
  if childsuY NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+childsuY;
  if interesY NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+interesY;
  if dividend NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+dividend;
  if rentalIY NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+rentalIY;
  if estatesY NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+estatesY;
  if pensionY NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+pensionY;
  if faincome NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+faincome;
  if maincome NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+maincome;
  if mgincome NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+mgincome;
  if fgincome NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+fgincome;
  if prgamt NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+prgamt;
  do I=1 to 14;
    if otfamI[I] NOT in (-1,-2,-3,-4) then groshhIY=groshhIY+otfamI[I];
  end;
end;

```

```

if YI_1900=-1 or YI_8500=-1 or (nfarmwgY=-1 or farmwgY=-1 or nfarmwgP=-1 or farmwgP=-1 or
childsuY=-1 or interesY=-1 or dividend=-1 or rentalIY=-1 or estatesY=-1 or pensionY=-1 or
faincome=-1 or maincome=-1 or mgincome=-1 or fgincome=-1 or prgamt=-1 or otfamI01=-1 or
otfamI02=-1 or otfamI03=-1 or otfamI04=-1 or otfamI05=-1 or otfamI06=-1 or otfamI07=-1 or
otfamI08=-1 or otfamI09=-1 or otfamI10=-1 or otfamI11=-1 or otfamI12=-1 or otfamI13=-1 or
otfamI14=-1) then do;

```

```
groshhIY=-1;
invskpGIY=1;
end;
```

```
if YI_1900=-2 or YI_8500=-2 or (nfarmwgY=-2 or farmwgY=-2 or nfarmwgP=-2 or farmwgP=-2 or
childsuY=-2 or interesY=-2 or dividend=-2 or rentalIY=-2 or pensionY=-2 or estatesY=-2 or
faincome=-2 or maincome=-2 or mgincome=-2 or fgincome=-2 or prgamt=-2 or offamI01=-2 or
otfamI02=-2 or offamI03=-2 or offamI04=-2 or offamI05=-2 or offamI06=-2 or offamI07=-2 or
otfamI08=-2 or offamI09=-2 or offamI10=-2 or offamI11=-2 or offamI12=-2 or offamI13=-2 or
otfamI14=-2) then do;
  groshhIY=-2;
  invskpGIY=2;
end;
```

```
if YI_1900=-3 or YI_8500=-3 or (nfarmwgY=-3 or farmwgY=-3 or nfarmwgP=-3 or farmwgP=-3 or
childsuY=-3 or interesY=-3 or dividend=-3 or rentalIY=-3 or pensionY=-3 or estatesY=-3 or
faincome=-3 or maincome=-3 or mgincome=-3 or fgincome=-3 or prgamt=-3 or offamI01=-3 or
otfamI02=-3 or offamI03=-3 or offamI04=-3 or offamI05=-3 or offamI06=-3 or offamI07=-3 or
otfamI08=-3 or offamI09=-3 or offamI10=-3 or offamI11=-3 or offamI12=-3 or offamI13=-3 or
otfamI14=-3) then do;
  groshhIY=-3;
  invskpGIY=3;
end;
```

```
if YI_1900=0 or YI_1900=-4 then do; groshhIY=-4; vskpGIY=1; end;
```

```
/** give the people who were not interviewed in round 5 value -5. **/
if YAS50=-5 then do; groshhIY=-5; nonintGIY=1; end;
```

---

**/\*\* \*\*\*\*\* SECTION 2: HOUSEHOLD NET WORTH \*\*\*\*\* \*\*\*/**

```
/** In asset section, I flag top three items involved the most respondents:
```

```
  I. savings -- flag4400
```

```
  II. present value of vehicles owned -- flag4790  III. money still owed on these vehicles -- flag4840
```

```
  flag=0 indicating accurate income reported flag=1 indicating estimated income reported.  **/
```

```
/** If the youth OWNS some land, ITS PRESENT VLAUE.  **/
```

```
pvranch=0;
```

```
if YAS1400=1 then do;
```

```
/** If the youth reported owning all of it.  **/
```

```
  if YAS1610=1 or YAS1610=2 then do;
```

```
    if YAS1860=1 then pvranch=YAS1862;
```

```
    if YAS1860=2 then pvranch=YAS1864L+(YAS1864U-YAS1864L)/2;
```

```
    if ((YAS1860 ne 1 and YAS1860 ne 2) or (YAS1862=-1 or YAS1862=-2)) then do;
```

```
      if YAS1866=1 then do; pvranch=12500; end;
```

```
      if YAS1866=2 then do; pvranch=37500; end;
```

```
      if YAS1866=3 then do; pvranch=75000; end;
```

```
      if YAS1866=4 then do; pvranch=175000; end;
```

```
      if YAS1866=5 then do; pvranch=375000; end;
```

```

    if YAS1866=6 then do; pvranch=750000; end;
    if YAS1866=7 then do; pvranch=1000001; end;
end;
end;
/** If the youth reported owning only part of it      **/
if (YAS1610=3 or YAS1610=4) and YAS2120=100 then do;
  if YAS2140=1 then pvranch=YAS2142;
  if YAS2140=2 then pvranch=(YAS2144L+(YAS2144U-YAS2144L)/2);
  if ((YAS2140 ne 1 and YAS2140 ne 2) or (YAS2142=-1 or YAS2142=-2)) then do;
    if YAS2146=1 then do; pvranch=12500; end;
    if YAS2146=2 then do; pvranch=37500; end;
    if YAS2146=3 then do; pvranch=75000; end;
    if YAS2146=4 then do; pvranch=175000; end;
    if YAS2146=5 then do; pvranch=375000; end;
    if YAS2146=6 then do; pvranch=750000; end;
    if YAS2146=7 then do; pvranch=1000001; end;
  end;
end;
if (YAS1610=3 or YAS1610=4) and YAS2120 ne 100 then do;
  if YAS2170=1 then pvranch=YAS2172;
  if YAS2170=2 then pvranch=(YAS2174L+(YAS2174U-YAS2174L)/2);
  if ((YAS2170 ne 1 and YAS2170 ne 2) or (YAS2172=-1 or YAS2172=-2)) then do;
    if YAS2176=1 then do; pvranch=12500; end;
    if YAS2176=2 then do; pvranch=37500; end;
    if YAS2176=3 then do; pvranch=75000; end;
    if YAS2176=4 then do; pvranch=175000; end;
    if YAS2176=5 then do; pvranch=375000; end;
    if YAS2176=6 then do; pvranch=750000; end;
    if YAS2176=7 then do; pvranch=1000001; end;
  end;
end;
end;
end;
if YAS1400=-1 or YAS1610=-1 or YAS1864L=-1 or YAS1864U=-1 or YAS1866=-1 or YAS2144L=-1
or YAS2144U=-1 or YAS2146=-1 or YAS2174L=-1 or YAS2174U=-1 or YAS2176=-1
then pvranch=-1;
if YAS1400=-2 or YAS1610=-2 or YAS1864L=-2 or YAS1864U=-2 or YAS1866=-2 or YAS2144L=-2
or YAS2144U=-2 or YAS2146=-2 or YAS2174L=-2 or YAS2174U=-2 or YAS2176=-2
then pvranch=-2;
if YAS1400=-3 or YAS1610=-3 or YAS1864L=-3 or YAS1864U=-3 or YAS1866=-3 or YAS2144L=-3
or YAS2144U=-3 or YAS2146=-3 or YAS2174L=-3 or YAS2174U=-3 or YAS2176=-3
then pvranch=-3;

/** If the youth OWNS the mobile home and the site, ITS PRESENT VALUE.      **/
pvmbst=0;
if YAS1500=1 then do;
  if (YAS2520=1 or YAS2520=2) and YAS2550=1 then pvmbst=YAS2552;
  if (YAS2520=1 or YAS2520=2) and YAS2550=2 then pvmbst=YAS2554L+(YAS2554U-
YAS2554L)/2;
  if (YAS2520=1 or YAS2520=2) and ((YAS2550 ne 1 and YAS2550 ne 2) or (YAS2552=-1 or
YAS2552=-2)) then do;
    if YAS2556=1 then do; pvmbst=500; end;
  end;
end;

```

```

    if YAS2556=2 then do; pvmbst=1750; end;
    if YAS2556=3 then do; pvmbst=3750; end;
    if YAS2556=4 then do; pvmbst=7500; end;
    if YAS2556=5 then do; pvmbst=17500; end;
    if YAS2556=6 then do; pvmbst=37500; end;
    if YAS2556=7 then do; pvmbst=50001; end;
end;
end;
if YAS1500=-1 or YAS2520=-1 or YAS2554L=-1 or YAS2554U=-1 or YAS2556=-1 then pvmbst=-1;
if YAS1500=-2 or YAS2520=-2 or YAS2554L=-2 or YAS2554U=-2 or YAS2556=-2 then pvmbst=-2;
if YAS1500=-3 or YAS2520=-3 or YAS2552=-3 or YAS2554L=-3 or YAS2554U=-3 or YAS2556=-3
then pvmbst=-3;

/** If the youth owns only the mobile home, IT'S PRESENT VALUE.   **/
pvmb=0;
if YAS1500=1 then do;
    if (YAS2520=3 or YAS2520=4) and YAS2760=1 then pvmb=YAS2762;
    if (YAS2520=3 or YAS2520=4) and YAS2760=2 then pvmb=(YAS2764L+(YAS2764U-
YAS2764L)/2);
    if (YAS2520=3 or YAS2520=4) and ((YAS2760 ne 1 and YAS2760 ne 2) or (YAS2762=-1 or
YAS2762=-2)) then do;
        if YAS2766=1 then do; pvmb=12500; end;
        if YAS2766=2 then do; pvmb=37500; end;
        if YAS2766=3 then do; pvmb=75000; end;
        if YAS2766=4 then do; pvmb=175000; end;
        if YAS2766=5 then do; pvmb=375000; end;
        if YAS2766=6 then do; pvmb=750000; end;
        if YAS2766=7 then do; pvmb=1000001; end;
    end;
end;
if YAS1500=-1 or YAS2520=-1 or YAS2764L=-1 or YAS2764U=-1 or YAS2766=-1 then pvmb=-1;
if YAS1500=-2 or YAS2520=-2 or YAS2764L=-2 or YAS2764U=-2 or YAS2766=-2 then pvmb=-2;
if YAS1500=-3 or YAS2520=-3 or YAS2762=-3 or YAS2764L=-3 or YAS2764U=-3 or YAS2766=-3
then pvmb=-3;

/** If the youth owns only the mobile home site, ITS PRESENT VALUE.   **/
pvst=0;
if YAS1500=1 then do;
    if (YAS2520=5 or YAS2520=6) and YAS3010=1 then pvst=YAS3012;
    if (YAS2520=5 or YAS2520=6) and YAS3010=2 then pvst=(YAS3014L+(YAS3014U-
YAS3014L)/2);
    if (YAS2520=5 or YAS2520=6) and ((YAS3010 ne 1 and YAS3010 ne 2) or (YAS3012=-1 or
YAS3012=-2)) then do;
        if YAS3016=1 then do; pvst=12500; end;
        if YAS3016=2 then do; pvst=57500; end;
        if YAS3016=3 then do; pvst=75000; end;
        if YAS3016=4 then do; pvst=175000; end;
        if YAS3016=5 then do; pvst=575000; end;
        if YAS3016=6 then do; pvst=750000; end;
        if YAS3016=7 then do; pvst=1000001; end;
    end;
end;

```

```

end;
if YAS2520=-1 or YAS3014L=-1 or YAS3014U=-1 or YAS3016=-1 or YAS1500=-1 then pvst=-1;
if YAS2520=-2 or YAS3014L=-2 or YAS3014U=-2 or YAS3016=-2 or YAS1500=-2 then pvst=-2;
if YAS2520=-3 or YAS3012=-3 or YAS3014L=-3 or YAS3014U=-3 or YAS3016=-3 or YAS1500=-3
then pvst=-3;

/** If the respondent owns the apartment, its value.   **/
pvapthm=0;
if YAS3310=1 or YAS3310=2 then do;
  if YAS3380=1 then pvapthm=YAS3382;
  if YAS3380=2 then pvapthm=(YAS3384L+(YAS3384U-YAS3384L)/2);
  if ((YAS3380 ne 1 and YAS3380 ne 2) or (YAS3382=-1 or YAS3382=-2)) then do;
    if YAS3386=1 then do; pvapthm=500; end;
    if YAS3386=2 then do; pvapthm=1750; end;
    if YAS3386=3 then do; pvapthm=3750; end;
    if YAS3386=4 then do; pvapthm=7500; end;
    if YAS3386=5 then do; pvapthm=17500; end;
    if YAS3386=6 then do; pvapthm=37500; end;
    if YAS3386=7 then do; pvapthm=50001; end;
  end;
end;
if YAS3310=-1 or YAS3384L=-1 or YAS3384U=-1 or YAS3386=-1 then pvapthm=-1;
if YAS3310=-2 or YAS3384L=-2 or YAS3384U=-2 or YAS3386=-2 then pvapthm=-2;
if YAS3310=-3 or YAS3382=-3 or YAS3384L=-3 or YAS3384U=-3 or YAS3386=-3 then pvapthm=-3;

/** Mortgage or land contract on land or property.   **/
mortgagY=0;
if YAS3740=1 or YAS3740=2 then do;
  if YAS3750=1 and YAS3752 ge 0 then mortgagY=YAS3752;
  if YAS3750=2 and YAS3754U ge 0 and YAS3754L ge 0 then mortgagY=(YAS3754L+(YAS3754U-
YAS3754L)/2);
  if ((YAS3750 ne 1 and YAS3750 ne 2) or (YAS3752=-1 or YAS3752=-2)) then do;
    if YAS3756=1 then do; mortgagY=500; end;
    if YAS3756=2 then do; mortgagY=1750; end;
    if YAS3756=3 then do; mortgagY=3750; end;
    if YAS3756=4 then do; mortgagY=7500; end;
    if YAS3756=5 then do; mortgagY=17500; end;
    if YAS3756=6 then do; mortgagY=37500; end;
    if YAS3756=7 then do; mortgagY=50001; end;
  end;
end;
if YAS3740=-1 or YAS3754L=-1 or YAS3754U=-1 or YAS3756=-1 then mortgagY=-1;
if YAS3740=-2 or YAS3754L=-2 or YAS3754U=-2 or YAS3756=-2 then mortgagY=-2;
if YAS3740=-3 or YAS3752=-3 or YAS3754L=-3 or YAS3754U=-3 or YAS3756=-3 then mortgagY=-
3;

/** Loans from relatives or friends when respondent bought or remodeled the residence. **/
loanowed=0;
if YAS3790=1 then do;
  if YAS3810=1 and YAS3812 ge 0 then loanowed=YAS3812;
  if YAS3810=2 and YAS3814U ge 0 and YAS3814L ge 0 then loanowed=(YAS3814L+(YAS3814U-

```

```

YAS3814L)/2);
  if ((YAS3810 ne 1 and YAS3810 ne 2) or (YAS3812=-1 or YAS3812=-2)) then do;
    if YAS3816=1 then do; loanowed=12500; end;
    if YAS3816=2 then do; loanowed=37500; end;
    if YAS3816=3 then do; loanowed=75000; end;
    if YAS3816=4 then do; loanowed=175000; end;
    if YAS3816=5 then do; loanowed=375000; end;
    if YAS3816=6 then do; loanowed=750000; end;
    if YAS3816=7 then do; loanowed=1000001; end;
  end;
end;
if YAS3810=-1 or YAS3790=-1 or YAS3814L=-1 or YAS3814U=-1 or YAS3816=-1 then loanowed=-1;
if YAS3810=-2 or YAS3790=-2 or YAS3814L=-2 or YAS3814U=-2 or YAS3816=-2 then loanowed=-2;
if YAS3810=-3 or YAS3790=-3 or YAS3812=-3 or YAS3814L=-3 or YAS3814U=-3 or YAS3816=-3 then loanowed=-3;

/** Remaining loans from round 4 still unpaid, total amount owed **/
stilowed=0;
if (YAS3840=1 and YAS3850=1) or YAS3860=1 then do;
  if YAS3880=1 and YAS3882 ge 0 then stilowed=YAS3882;
  if YAS3880=2 and YAS3884U ge 0 and YAS3884L ge 0 then stilowed=(YAS3884L+(YAS3884U-YAS3884L)/2);
  if ((YAS3880 ne 1 and YAS3880 ne 2) or (YAS3882=-1 or YAS3882=-2)) then do;
    if YAS3886=1 then do; stilowed=500; end;
    if YAS3886=2 then do; stilowed=1750; end;
    if YAS3886=3 then do; stilowed=3750; end;
    if YAS3886=4 then do; stilowed=7500; end;
    if YAS3886=5 then do; stilowed=17500; end;
    if YAS3886=6 then do; stilowed=37500; end;
    if YAS3886=7 then do;stilowed=50001; end;
  end;
end;
if YAS3860=-1 or YAS3884L=-1 or YAS3884U=-1 or YAS3886=-1 then stilowed=-1;
if YAS3860=-2 or YAS3884L=-2 or YAS3884U=-2 or YAS3886=-2 then stilowed=-2;
if YAS3860=-3 or YAS3882=-3 or YAS3884L=-3 or YAS3884U=-3 or YAS3886=-3 then stilowed=-3;

/** Second mortgages. **/
secmortY=0;
if YAS3910=1 then do;
  if YAS3920=1 and YAS3922 ge 0 then secmortY=YAS3922;
  if YAS3920=2 and YAS3924U ge 0 and YAS3924L ge 0 then secmortY=(YAS3924L+(YAS3924U-YAS3924L)/2);
  if ((YAS3920 ne 1 and YAS3920 ne 2) or (YAS3922=-1 or YAS3922=-2)) then do;
    if YAS3926=1 then do; secmortY=500; end;
    if YAS3926=2 then do; secmortY=1750; end;
    if YAS3926=3 then do; secmortY=3750; end;
    if YAS3926=4 then do; secmortY=7500; end;
    if YAS3926=5 then do; secmortY=17500; end;
    if YAS3926=6 then do; secmortY=37500; end;
  end;
end;

```

```

    if YAS3926=7 then do; secmortY=50001; end;
end;
end;
if YAS3910=-1 or YAS3924L=-1 or YAS3924U=-1 or YAS3926=-1 then secmortY=-1;
if YAS3910=-2 or YAS3924L=-2 or YAS3924U=-2 or YAS3926=-2 then secmortY=-2;
if YAS3910=-3 or YAS3922=-3 or YAS3924L=-3 or YAS3924U=-3 or YAS3926=-3 then secmortY=-3;

```

```

/** Taxes on the property to be paid.      **/

```

```

proptaxY=0;
if YAS3950=1 then do;
    if YAS3960=1 and YAS3962 ge 0 then proptaxY=YAS3962;
    if YAS3960=2 and YAS3964U ge 0 and YAS3964L ge 0 then proptaxY=(YAS3964L+(YAS3964U-YAS3964L)/2);
    if ((YAS3960 ne 1 and YAS3960 ne 2) or (YAS3962=-1 or YAS3962=-2)) then do;
        if YAS3966=1 then do; proptaxY=500; end;
        if YAS3966=2 then do; proptaxY=1750; end;
        if YAS3966=3 then do; proptaxY=3750; end;
        if YAS3966=4 then do; proptaxY=7500; end;
        if YAS3966=5 then do; proptaxY=17500; end;
        if YAS3966=6 then do; proptaxY=37500; end;
        if YAS3966=7 then do; proptaxY=50001; end;
    end;
end;
if YAS3950=-1 or YAS3964L=-1 or YAS3964U=-1 or YAS3966=-1 then proptaxY=-1;
if YAS3950=-2 or YAS3964L=-2 or YAS3964U=-2 or YAS3966=-2 then proptaxY=-2;
if YAS3950=-3 or YAS3962=-3 or YAS3964L=-3 or YAS3964U=-3 or YAS3966=-3 then proptaxY=-3;

```

```

/** Own a business, partnership or professional practice.      **/

```

```

pvbuss=0;
if YAS4010=1 or YAS4010=2 then do;
    if YAS4030=1 then pvbuss=YAS4032;
    if YAS4030=2 then pvbuss=(YAS4034L+(YAS4034U-YAS4034L)/2);
    if ((YAS4030 ne 1 and YAS4030 ne 2) or (YAS4032=-1 or YAS4032=-2)) then do;
        if YAS4036=1 then do; pvbuss=12500; end;
        if YAS4036=2 then do; pvbuss=37500; end;
        if YAS4036=3 then do; pvbuss=75000; end;
        if YAS4036=4 then do; pvbuss=175000; end;
        if YAS4036=5 then do; pvbuss=375000; end;
        if YAS4036=6 then do; pvbuss=750000; end;
        if YAS4036=7 then do; pvbuss=1000001; end;
    end;
end;
if YAS4010=-1 or YAS4034L=-1 or YAS4034U=-1 or YAS4036=-1 then pvbuss=-1;
if YAS4010=-2 or YAS4034L=-2 or YAS4034U=-2 or YAS4036=-2 then pvbuss=-2;
if YAS4010=-3 or YAS4032=-3 or YAS4034L=-3 or YAS4034U=-3 or YAS4036=-3 then pvbuss=-3;

```

```

/** Second real estate owned.      **/

```

```

secrestY=0;
if YAS4140=1 or YAS4140=2 then do;

```

```

if YAS4160=1 then secrestY=YAS4162;
if YAS4160=2 then secrestY=(YAS4164L+(YAS4164U-YAS4164L)/2);
if ((YAS4160 ne 1 and YAS4160 ne 2) or (YAS4162=-1 or YAS4162=-2)) then do;
  if YAS4166=1 then do; secrestY=12500; end;
  if YAS4166=2 then do; secrestY=37500; end;
  if YAS4166=3 then do; secrestY=75000; end;
  if YAS4166=4 then do; secrestY=175000; end;
  if YAS4166=5 then do; secrestY=375000; end;
  if YAS4166=6 then do; secrestY=750000; end;
  if YAS4166=7 then do; secrestY=1000001; end;
end;
end;
if YAS4140=-1 or YAS4164L=-1 or YAS4164U=-1 or YAS4166=-1 then secrestY=-1;
if YAS4140=-2 or YAS4164L=-2 or YAS4164U=-2 or YAS4166=-2 then secrestY=-2;
if YAS4140=-3 or YAS4162=-3 or YAS4164L=-3 or YAS4164U=-3 or YAS4166=-3 then secrestY=-3;

```

/\* Retirement plans or pensions \*/

```

retireY=0;
if YAS4270=1 or YAS4270=2 or YAS4270=3 then do;
  if YAS4290=1 and YAS4292 ge 0 then retireY=YAS4292;
  if YAS4290=2 and YAS4294U ge 0 and YAS4294L ge 0 then retireY=(YAS4294L+(YAS4294U-
YAS4294L)/2);
  if ((YAS4290 ne 1 and YAS4290 ne 2) or (YAS4292=-1 or YAS4292=-2)) then do;
    if YAS4296=1 then do; retireY=2500; end;
    if YAS4296=2 then do; retireY=7500; end;
    if YAS4296=3 then do; retireY=17500; end;
    if YAS4296=4 then do; retireY=37500; end;
    if YAS4296=5 then do; retireY=75000; end;
    if YAS4296=6 then do; retireY=175000; end;
    if YAS4296=7 then do; retireY=250001; end;
  end;
end;
if YAS4270=-1 or YAS4294L=-1 or YAS4294U=-1 or YAS4296=-1 then retireY=-1;
if YAS4270=-2 or YAS4294L=-2 or YAS4294U=-2 or YAS4296=-2 then retireY=-2;
if YAS4270=-3 or YAS4292=-3 or YAS4294L=-3 or YAS4294U=-3 or YAS4296=-3 then retireY=-3;

```

/\*\* Savings in saving accounts, money market, funds, trusts,... \*/

```

savingsY=0; flag4400=0;
if YAS4400=1 or YAS4400=2 or YAS4400=3 then do;
  if YAS4420=1 and YAS4422 ge 0 then savingsY=YAS4422;
  if YAS4420=2 and YAS4424U ge 0 and YAS4424L ge 0 then savingsY=(YAS4424L+(YAS4424U-
YAS4424L)/2);
  if ((YAS4420 ne 1 and YAS4420 ne 2) or (YAS4422=-1 or YAS4422=-2)) then do;
    if YAS4426=1 then do; savingsY=500; flag4400=1; end;
    if YAS4426=2 then do; savingsY=1750; flag4400=1; end;
    if YAS4426=3 then do; savingsY=3750; flag4400=1; end;
    if YAS4426=4 then do; savingsY=7500; flag4400=1; end;
    if YAS4426=5 then do; savingsY=17500; flag4400=1; end;
    if YAS4426=6 then do; savingsY=37500; flag4400=1; end;
    if YAS4426=7 then do; savingsY=50001; flag4400=1; end;
  end;
end;

```



```

end;
if YAS4400=-1 or YAS4424L=-1 or YAS4424U=-2 or YAS4426=-1 then savingsY=-1;
if YAS4400=-2 or YAS4424L=-2 or YAS4424U=-2 or YAS4426=-2 then savingsY=-2;
if YAS4400=-3 or YAS4422=-3 or YAS4424L=-3 or YAS4424U=-3 or YAS4426=-3 then savingsY=-3;

/** Other savings in bonds or CDs. **/
othsavY=0;
if YAS4530=1 or YAS4530=2 or YAS4530=3 then do;
  if YAS4550=1 and YAS4552 ge 0 then othsavY=YAS4552;
  if YAS4550=2 and YAS4554U ge 0 and YAS4554L ge 0 then othsavY=(YAS4554L+(YAS4554U-YAS4554L)/2);
  if ((YAS4550 ne 1 and YAS4550 ne 2) or (YAS4552=-1 or YAS4552=-2)) then do;
    if YAS4556=1 then do; othsavY=500; end;
    if YAS4556=2 then do; othsavY=1750; end;
    if YAS4556=3 then do; othsavY=3750; end;
    if YAS4556=4 then do; othsavY=7500; end;
    if YAS4556=5 then do; othsavY=17500; end;
    if YAS4556=6 then do; othsavY=37500; end;
    if YAS4556=7 then do; othsavY=50001; end;
  end;
end;
if YAS4530=-1 or YAS4554L=-1 or YAS4554U=-1 or YAS4556=-1 then othsavY=-1;
if YAS4530=-2 or YAS4554L=-2 or YAS4554U=-2 or YAS4556=-2 then othsavY=-2;
if YAS4530=-3 or YAS4552=-3 or YAS4554L=-3 or YAS4554U=-3 or YAS4556=-3 then othsavY=-3;

/** Stocks, mutual funds. **/
stockY=0;
if YAS4660=1 or YAS4660=2 or YAS4660=3 then do;
  if YAS4680=1 then stockY=YAS4682;
  if YAS4680=2 then stockY=(YAS4684L+(YAS4684U-YAS4684L)/2);
  if ((YAS4680 ne 1 and YAS4680 ne 2) or (YAS4682=-1 or YAS4682=-2)) then do;
    if YAS4686=1 then do; stockY=500; end;
    if YAS4686=2 then do; stockY=1750; end;
    if YAS4686=3 then do; stockY=3750; end;
    if YAS4686=4 then do; stockY=7500; end;
    if YAS4686=5 then do; stockY=17500; end;
    if YAS4686=6 then do; stockY=37500; end;
    if YAS4686=7 then do; stockY=50001; end;
  end;
end;
if YAS4660=-1 or YAS4684L=-1 or YAS4684U=-1 or YAS4686=-1 then stockY=-1;
if YAS4660=-2 or YAS4684L=-2 or YAS4684U=-2 or YAS4686=-2 then stockY=-2;
if YAS4660=-3 or YAS4682=-3 or YAS4684L=-3 or YAS4684U=-3 or YAS4686=-3 then stockY=-3;

/** Present value of vehicles owned. **/
pvcarsY=0;
flag4790=0;
if YAS4790=1 or YAS4790=2 or YAS4790=3 then do;
  if YAS4810=1 then pvcarsY=YAS4812;
  if YAS4810=2 then pvcarsY=(YAS4814L+(YAS4814U-YAS4814L)/2);

```

```

if ((YAS4810 ne 1 and YAS4810 ne 2) or (YAS4812=-1 or YAS4812=-2)) then do;
  if YAS4816=1 then do; pvcarsY=2500; flag4790=1; end;
  if YAS4816=2 then do; pvcarsY=7500; flag4790=1; end;
  if YAS4816=3 then do; pvcarsY=17500; flag4790=1;end;
  if YAS4816=4 then do; pvcarsY=37500; flag4790=1; end;
  if YAS4816=5 then do; pvcarsY=75000; flag4790=1; end;
  if YAS4816=6 then do; pvcarsY=175000; flag4790=1; end;
  if YAS4816=7 then do; pvcarsY=250001; flag4790=1; end;
end;
end;
if YAS4790=-1 or YAS4814L=-1 or YAS4814U=-1 or YAS4816=-1 then pvcarsY=-1;
if YAS4790=-2 or YAS4814L=-2 or YAS4814U=-2 or YAS4816=-2 then pvcarsY=-2;
if YAS4790=-3 or YAS4812=-3 or YAS4814L=-3 or YAS4814U=-3 or YAS4816=-3 then pvcarsY=-3;

/** Money still owed on these vehicles. **/
cardebt=0;
flag4840=0;
if YAS4840=1 then cardebt=YAS4842;
if YAS4840=2 then cardebt=(YAS4844L+(YAS4844U-YAS4844L)/2);
if ((YAS4840 ne 1 and YAS4840 ne 2) or (YAS4842=-1 or YAS4842=-2)) then do;
  if YAS4846=1 then do; cardebt=2500; flag4840=1; end;
  if YAS4846=2 then do; cardebt=7500; flag4840=1; end;
  if YAS4846=3 then do; cardebt=17500; flag4840=1; end;
  if YAS4846=4 then do; cardebt=37500; flag4840=1; end;
  if YAS4846=5 then do; cardebt=75000; flag4840=1; end;
  if YAS4846=6 then do; cardebt=175000; flag4840=1; end;
  if YAS4846=7 then do; cardebt=250001; flag4840=1; end;
end;
end;
if YAS4790=-1 or YAS4844L=-1 or YAS4844U=-1 or YAS4846=-1 then cardebt=-1;
if YAS4790=-2 or YAS4844L=-2 or YAS4844U=-2 or YAS4846=-2 then cardebt=-2;
if YAS4790=-3 or YAS4842=-3 or YAS4844L=-3 or YAS4844U=-3 or YAS4846=-3 then cardebt=-3;

/** Present value of owned furniture **/
pvfurntY=0;
if YAS4870=1 then do; pvfurntY=2500; end;
if YAS4870=2 then do; pvfurntY=7500; end;
if YAS4870=3 then do; pvfurntY=17500; end;
if YAS4870=4 then do; pvfurntY=37500; end;
if YAS4870=5 then do; pvfurntY=75000; end;
if YAS4870=6 then do; pvfurntY=175000; end;
if YAS4870=7 then do; pvfurntY=250001; end;
if YAS4870=-1 then pvfurntY=-1;
if YAS4870=-2 then pvfurntY=-2;
if YAS4870=-3 then pvfurntY=-3;

/** Other assets not being mentioned before. **/
otassetY=0;
if YAS4880=1 or YAS4880=2 or YAS4880=3 then do;
  if YAS4900=1 then otassetY=YAS4902;
  if YAS4900=2 then otassetY=(YAS4904L+(YAS4904U-YAS4904L)/2);
  if ((YAS4900 ne 1 and YAS4900 ne 2) or (YAS4902=-1 or YAS4902=-2)) then do;

```

```

if YAS4906=1 then do; otassetY=2500; end;
if YAS4906=2 then do; otassetY=7500; end;
if YAS4906=3 then do; otassetY=17500; end;
if YAS4906=4 then do; otassetY=37500; end;
if YAS4906=5 then do; otassetY=75000; end;
if YAS4906=6 then do; otassetY=175000; end;
if YAS4906=7 then do; otassetY=250001; end;
end;
end;
if YAS4880=-1 or YAS4904L=-1 or YAS4904U=-1 or YAS4906=-1 then otassetY=-1;
if YAS4880=-2 or YAS4904L=-2 or YAS4904U=-2 or YAS4906=-2 then otassetY=-2;
if YAS4880=-3 or YAS4902=-3 or YAS4904L=-3 or YAS4904U=-3 or YAS4906=-3 then otassetY=-3;

/* Loans still owed to family or relatives */
array rloan rloan01 rloan02 rloan03 rloan04 rloan05 rloan06 rloan07 rloan08 rloan09 rloan10 rloan11
rloan12 rloan13 rloan14 rloan15 rloan16 rloan17 rloan18 rloan19 rloan20;
array A5080 A508001 A508002 A508003 A508004 A508005 A508006 A508007 A508008 A508009
A508010 A508011 A508012 A508013 A508014 A508015 A508016 A508017 A508018 A508019
A508020;
array A5082 A508201 A508202 A508203 A508204 A508205 A508206 A508207 A508208 A508209
A508210 A508211 A508212 A508213 A508214 A508215 A508216 A508217 A508218 A508219
A508220;
array A5084L A5084L01 A5084L02 A5084L03 A5084L04 A5084L05 A5084L06 A5084L07
A5084L08 A5084L09 A5084L10 A5084L11 A5084L12 A5084L13 A5084L14 A5084L15 A5084L16
A5084L17 A5084L18 A5084L19 A5084L20;
array A5084U A5084U01 A5084U02 A5084U03 A5084U04 A5084U05 A5084U06 A5084U07
A5084U08 A5084U09 A5084U10 A5084U11 A5084U12 A5084U13 A5084U14 A5084U15
A5084U16 A5084U17 A5084U18 A5084U19 A5084U20;
array A5086 A508601 A508602 A508603 A508604 A508605 A508606 A508607 A508608 A508609
A508610 A508611 A508612 A508613 A508614 A508615 A508616 A508617 A508618 A508619
A508620;
array A5170 A517001 A517002 A517003 A517004 A517005 A517006 A517007 A517008 A517009
A517010 A517011 A517012 A517013 A517014 A517015 A517016 A517017 A517018 A517019
A517020;
array A5172 A517201 A517202 A517203 A517204 A517205 A517206 A517207 A517208 A517209
A517210 A517211 A517212 A517213 A517214 A517215 A517216 A517217 A517218 A517219
A517220;
array A5174L A5174L01 A5174L02 A5174L03 A5174L04 A5174L05 A5174L06 A5174L07
A5174L08 A5174L09 A5174L10 A5174L11 A5174L12 A5174L13 A5174L14 A5174L15 A5174L16
A5174L17 A5174L18 A5174L19 A5174L20;
array A5174U A5174U01 A5174U02 A5174U03 A5174U04 A5174U05 A5174U06 A5174U07
A5174U08 A5174U09 A5174U10 A5174U11 A5174U12 A5174U13 A5174U14 A5174U15
A5174U16 A5174U17 A5174U18 A5174U19 A5174U20;
array A5176 A517601 A517602 A517603 A517604 A517605 A517606 A517607 A517608 A517609
A517610 A517611 A517612 A517613 A517614 A517615 A517616 A517617 A517618 A517619
A517620;
array A5074 A507401-A507420;
array A5066 A506601-A506620;
array A5164 A516401-A516420;

do I=1 to 20;

```

```

rloan(I)=0;
if YAS5040=1 and YAS5060 ne 1 and YAS5062 ne 1 then do;
  if A5080(I)=1 then rloan(I)=A5082(I);
  if A5080(I)=2 then rloan(I)=(A5084L(I)+(A5084U(I)-A5084L(I))/2);
  if ((A5080(I) ne 1 and A5080(I) ne 2) or (A5082(I)=-1 or A5082(I)=-2)) then do;
    if A5086(I)=1 then do; rloan(I)=500; end;
    if A5086(I)=2 then do; rloan(I)=1750; end;
    if A5086(I)=3 then do; rloan(I)=3750; end;
    if A5086(I)=4 then do; rloan(I)=7500; end;
    if A5086(I)=5 then do; rloan(I)=17500; end;
    if A5086(I)=6 then do; rloan(I)=37500; end;
    if A5086(I)=7 then do; rloan(I)=50001; end;
  end;
end;
if YAS5040=-1 or A5074(I)=-1 or A5086(I)=-1 or A5084L[I]=-1 or A5084U[I]=-1 then rloan(I)=-1;
if YAS5040=-2 or A5074(I)=-2 or A5086(I)=-2 or A5084L[I]=-2 or A5084U[I]=-2 then rloan(I)=-2;
if YAS5040=-3 or A5082(I)=-3 or A5086(I)=-3 or A5084L[I]=-3 or A5084U[I]=-3 then rloan(I)=-3;

if YAS5130=1 and YAS5150 ne 1 and YAS5152 ne 1 then do;
  if A5170(I)=1 then rloan(I)=A5172(I);
  if A5170(I)=2 then rloan(I)=(A5174L(I)+(A5174U(I)-A5174L(I))/2);
  if ((A5170(I) ne 1 and A5170(I) ne 2) or (A5172(I)=-1 or A5172(I)=-2)) then do;
    if A5176(I)=1 then do; rloan(I)=500; end;
    if A5176(I)=2 then do; rloan(I)=1750; end;
    if A5176(I)=3 then do; rloan(I)=3750; end;
    if A5176(I)=4 then do; rloan(I)=7500; end;
    if A5176(I)=5 then do; rloan(I)=17500; end;
    if A5176(I)=6 then do; rloan(I)=37500; end;
    if A5176(I)=7 then do; rloan(I)=50001; end;
  end;
end;
if YAS5130=-1 or A5164(I)=-1 or A5176(I)=-1 or A5174L[I]=-1 or A5174U[I]=-1 then rloan(I)=-1;
if YAS5130=-2 or A5164(I)=-2 or A5176(I)=-2 or A5174L[I]=-2 or A5174U[I]=-2 then rloan(I)=-2;
if YAS5130=-3 or A5172(I)=-3 or A5176(I)=-3 or A5174L[I]=-3 or A5174U[I]=-3 then rloan(I)=-3;
end;

/** Other debts from loans, credit cards, etc... */
othdebtY=0;
if YAS5210=1 then do;
  if YAS5220=1 then othdebtY=YAS5222;
  if YAS5220=2 then othdebtY=(YAS5224L+(YAS5224U-YAS5224L)/2);
  if ((YAS5220 ne 1 and YAS5220 ne 2) or (YAS5222=-1 or YAS5222=-2)) then do;
    if YAS5226=1 then do; othdebtY=500; end;
    if YAS5226=2 then do; othdebtY=1750; end;
    if YAS5226=3 then do; othdebtY=3750; end;
    if YAS5226=4 then do; othdebtY=7500; end;
    if YAS5226=5 then do; othdebtY=17500; end;
    if YAS5226=6 then do; othdebtY=37500; end;
    if YAS5226=7 then do; othdebtY=50001; end;
  end;
end;
end;

```

if YAS5210=-1 or YAS5224L=-1 or YAS5224U=-1 or YAS5226=-1 then othdebtY=-1;  
if YAS5210=-2 or YAS5224L=-2 or YAS5224U=-2 or YAS5226=-2 then othdebtY=-2;  
if YAS5210=-3 or YAS5222=-3 or YAS5224L=-3 or YAS5224U=-3 or YAS5226=-3 then othdebtY=-3;

/\*\* We now calculate the household net worth according to the youth: hhworthY=assets-liabilities.  
\*\*/

/\*\* Till Round 3 we have used YAST-050 as the question determining whether respondents are asked the assets question but in this round YAST-105 does that. So we switched to using YAST-105 instead of YAST-050 as well. \*\*/

if (YAS105=0 or YAS105=1) then do;

hhworthY=0;  
if pvranch NOT in (-1, -2, -3) then hhworthY=hhworthY+pvranch;  
if pvmbst NOT in (-1, -2, -3) then hhworthY=hhworthY+pvmbst;  
if pvmb NOT in (-1, -2, -3) then hhworthY=hhworthY+pvmb;  
if pvst NOT in (-1, -2, -3) then hhworthY=hhworthY+pvst;  
if pvapthm NOT in (-1, -2, -3) then hhworthY=hhworthY+pvapthm;  
if pvbuss NOT in (-1, -2, -3) then hhworthY=hhworthY+pvbuss;  
if secrestY NOT in (-1, -2, -3) then hhworthY=hhworthY+secrestY;  
if retireY NOT in (-1, -2, -3) then hhworthY=hhworthY+retireY;  
if savingsY NOT in (-1, -2, -3) then hhworthY=hhworthY+savingsY;  
if othsavY NOT in (-1, -2, -3) then hhworthY=hhworthY+othsavY;  
if stockY NOT in (-1, -2, -3) then hhworthY=hhworthY+stockY;  
if pvcarsY NOT in (-1, -2, -3) then hhworthY=hhworthY+pvcarsY;  
if pvfurntY NOT in (-1, -2, -3) then hhworthY=hhworthY+pvfurntY;  
if otassetY NOT in (-1, -2, -3) then hhworthY=hhworthY+otassetY;  
if mortgagY NOT in (-1, -2, -3) then hhworthY=hhworthY-mortgagY;  
if loanowed NOT in (-1, -2, -3) then hhworthY=hhworthY-loanowed;  
if stilowed NOT in (-1, -2, -3) then hhworthY=hhworthY-stilowed;  
if secmortY NOT in (-1, -2, -3) then hhworthY=hhworthY-secmortY;  
if cardebt NOT in (-1, -2, -3) then hhworthY=hhworthY-cardebt;  
if othdebtY NOT in (-1, -2, -3) then hhworthY=hhworthY-othdebtY;  
do I=1 to 20; if rloan[I] NOT in (-1, -2, -3) then hhworthY=hhworthY-rloan[I]; end;  
end;

if YAS105=-4 then do; hhworthY=-4; VskpHHY=1; end;

if YAS105=-1 or (pvranch=-1 or pvmbst=-1 or pvmb=-1 or pvst=-1 or pvapthm=-1 or pvbuss=-1 or secrestY=-1 or retireY=-1 or savingsY=-1 or othsavY=-1 or stockY=-1 or pvcarsY=-1 or pvfurntY=-1 or otassetY=-1 or mortgagY=-1 or loanowed=-1 or stilowed=-1 or secmortY=-1 or cardebt=-1 or rloan01=-1 or rloan02=-1 or rloan03=-1 or rloan04=-1 or rloan05=-1 or rloan06=-1 or rloan07=-1 or rloan08=-1 or rloan09=-1 or rloan10=-1 or rloan11=-1 or rloan12=-1 or rloan13=-1 or rloan14=-1 or rloan15=-1 or rloan16=-1 or rloan17=-1 or rloan18=-1 or rloan19=-1 or rloan20=-1 or othdebtY=-1) then do;

hhworthY=-1;  
InvskpHHY=1;  
end;

if YAS105=-2 or (pvranch=-2 or pvmbst=-2 or pvmb=-2 or pvst=-2 or pvapthm=-2 or pvbuss=-2 or secrestY=-2 or retireY=-2 or savingsY=-2 or othsavY=-2 or stockY=-2 or pvcarsY=-2 or pvfurntY=-2 or otassetY=-2 or mortgagY=-2 or loanowed=-2 or stilowed=-2 or secmortY=-2 or cardebt=-2 or rloan01=-2 or rloan02=-2 or rloan03=-2 or rloan04=-2 or rloan05=-2 or rloan06=-2 or rloan07=-2 or

```

rloan08=-2 or rloan09=-2 or rloan10=-2 or rloan11=-2 or rloan12=-2 or rloan13=-2 or rloan14=-2 or
rloan15=-2 or rloan16=-2 or rloan17=-2 or rloan18=-2 or rloan19=-2 or rloan20=-2 or othdebtY=-2)
then do;
  hhworthY=-2;
  InvskpHHY=2;
end;
if YAS105=-3 or (pvranch=-3 or pvmbst=-3 or pvmb=-3 or pvst=-3 or pvapthm=-3 or pvbuss=-3 or
secrestY=-3 or retireY=-3 or savingsY=-3 or othsavY=-3 or stockY=-3 or pvcarsY=-3 or pvfurntY=-3
or otassetY=-3 or mortgagY=-3 or loanowed=-3 or stilowed=-3 or secmortY=-3 or cardebt=-3 or
rloan01=-3 or rloan02=-3 or rloan03=-3 or rloan04=-3 or rloan05=-3 or rloan06=-3 or rloan07=-3 or
rloan08=-3 or rloan09=-3 or rloan10=-3 or rloan11=-3 or rloan12=-3 or rloan13=-3 or rloan14=-3 or
rloan15=-3 or rloan16=-3 or rloan17=-3 or rloan18=-3 or rloan19=-3 or rloan20=-3 or othdebtY=-3)
then do;
  hhworthY=-3;
  InvskpHHY=3;
end;

/** HANDEDIT: These respondents give a positive value for the lower limit but ZERO for the upper
limit for certain assets. So we give them a -3. */
if pubid=430 or pubid=432 or pubid=614 or pubid=1829 or pubid=3065 or pubid=7785 or pubid=7983
or pubid=8164 or pubid=8441 then do;
  hhworthY=-3; FLAG_RANGE=1; InvskpHHY=3; end;

/** give the people who were not interviewed in round 5 value -5. */
if YAS105=-5 then do; hhworthY=-5; nonintHHY=1; end;

hhworthY=ROUND(hhworthY,1);

```

---

**\*\*\*\*\* SECTION 3: HOUSEHOLD POVERTY STATUS \*\*\*\*\***  
**\*\*\*\*\***

```

/** Create the poverty thresholds (taken from U.S.Census' website -
http://www.census.gov/hhes/poverty/threshld/thresh00.html) */

povert=-4;

if HHSIZE=-1 or under18=-1 then povert=-1;
if HHSIZE=-2 or under18=-2 then povert=-2;
if HHSIZE=-3 or under18=-3 then povert=-3;

if HHSIZE=1 then povert=8959;

if HHSIZE=2 then do;
  if under18=0 then povert=11531;
  if under18=1 then povert=11869;
  if under18=2 then povert=11869;
end;

if HHSIZE=3 then do;
  if under18=0 then povert=13470;

```

```
if under18=1 then povert=13861;
if under18=2 then povert=13874;
if under18=3 then povert=13874;
end;
```

```
if HHSIZE=4 then do;
  if under18=0 then povert=17761;
  if under18=1 then povert=18052;
  if under18=2 then povert=17463;
  if under18=3 then povert=17524;
  if under18=4 then povert=17524;
end;
```

```
if HHSIZE=5 then do;
  if under18=0 then povert=21419;
  if under18=1 then povert=21731;
  if under18=2 then povert=21065;
  if under18=3 then povert=20550;
  if under18=4 then povert=20236;
  if under18=5 then povert=20236;
end;
```

```
if HHSIZE=6 then do;
  if under18=0 then povert=24636;
  if under18=1 then povert=24734;
  if under18=2 then povert=24224;
  if under18=3 then povert=23736;
  if under18=4 then povert=23009;
  if under18=5 then povert=22579;
  if under18=6 then povert=22579;
end;
```

```
if HHSIZE=7 then do;
  if under18=0 then povert=28347;
  if under18=1 then povert=28524;
  if under18=2 then povert=27914;
  if under18=3 then povert=27489;
  if under18=4 then povert=26696;
  if under18=5 then povert=25772;
  if under18=6 then povert=24758;
  if under18=7 then povert=24758;
end;
```

```
if HHSIZE=8 then do;
  if under18=0 then povert=31704;
  if under18=1 then povert=31984;
  if under18=2 then povert=31408;
  if under18=3 then povert=30904;
  if under18=4 then povert=30188;
  if under18=5 then povert=29279;
  if under18=6 then povert=28334;
```

```
if under18=7 then povert=28093;
if under18=8 then povert=28093;
end;
```

```
if HHSIZE>=9 then do;
  if under18= 0 then povert=38138;
  if under18= 1 then povert=38322;
  if under18= 2 then povert=37813;
  if under18= 3 then povert=37385;
  if under18= 4 then povert=36682;
  if under18= 5 then povert=35716;
  if under18= 6 then povert=34841;
  if under18= 7 then povert=34625;
  if under18 ge 8 then povert=33291;
end;
```

```
povthr=-4;
if groshhIY ge 0 and povert gt 0 then povthr=ROUND(((groshhIY*100)/povert),1);
if groshhIY eq -1 or povert=-1 then do; povthr=-1; invskpPOV=1; end;
if groshhIY eq -2 or povert=-2 then do; povthr=-2; invskpPOV=2; end;
if groshhIY eq -3 or povert=-3 then do; povthr=-3; invskpPOV=3; end;
if groshhIY eq -4 then do; povthr=-4; vskpPOV=1; end;
```

```
/** Repondents who were not interviewed in round 5    **/
if YAS50=-5 then do; povthr=-5; nonintPOV=1; end;
```

```
/** Changing all invalid skips to -3.    **/
if groshhIY=-1 then groshhIY=-3; if groshhIY=-2 then groshhIY=-3;
if hhworthY=-1 then hhworthY=-3; if hhworthY=-2 then hhworthY=-3;
if povthr=-1 then povthr=-3; if povthr=-2 then povthr=-3;
```

```
pvcomb=0;
if pvcarsy>0 and pvfurntY>0 then pvcomb=(pvcarsy+pvfurntY);
```

---

**/\*\*\*\*\*\*\* SECTION 4: TOPCODING INCOME, NET WORTH, AND POVERTY RATIO  
\*\*\*\*\*\*/**

```
/** Topcoding the groshhIY, hhworthY and povthr variables.    **/
```

```
topcode_GIY=191001;
topcode_HHY=183900;
topcode_POV=1151;
```

```
mean_topcode_GIY=285805;
mean_topcode_HHY=1111342;
mean_topcode_POV=1720;
```

```
if groshhIY ge topcode_GIY then do; groshhIY=mean_topcode_GIY; topcode1=1; end;
if hhworthY ge topcode_HHY then do; hhworthY=mean_topcode_HHY; topcode2=1; end;
if povthr ge topcode_POV then do; povthr=mean_topcode_POV; topcode3=1; end;
```



if topcode1=1 and topcode2=. and topcode3=. then topcode=1;  
 if topcode1=. and topcode2=1 and topcode3=. then topcode=2;  
 if topcode1=. and topcode2=. and topcode3=1 then topcode=3;

if topcode1=1 and topcode2=1 and topcode3=. then topcode=12;  
 if topcode1=1 and topcode2=. and topcode3=1 then topcode=13;  
 if topcode1=. and topcode2=1 and topcode3=1 then topcode=23;

if topcode1=1 and topcode2=1 and topcode3=1 then topcode=123;

endsas;

## Participation in Government Programs

### Variables Created:

- CV\_AMT\_GOVNT\_PGM\_PCY.80 - CV\_AMT\_GOVNT\_PGM\_PCY.00
- CV\_GOVNT\_PGM\_EVER
- CV\_GOVNT\_PGM\_YR.80 - CV\_GOVNT\_PGM\_YR.00

### Variables Used

Name in Program	Question Name on CD	Name in Program	Question Name on CD
<b>Unemployment Compensation</b>			
p010	YPRG-010	p50001	YPRG-5000.01
p1210	YPRG-1210	p52001	YPRG-5200.01
ps1830	YPRG-1830_UPD	p59001, p59002	YPRG-5900.01, .02
ps18501	YPRG-1850_UPD	yint_d, yint_m, yint_y	SYMBOL!CURDATE~D, ~M, ~Y
ps1897	YPRG-1897_UPD	bdate_d, bdate_m, bdate_y	KEY!BDATE_D, _M, _Y
ps19301	YPRG-1930_UPD	uia1total80-uia1total98	CV_AMT_UI_YR.80-.98
ps40001M, ps40001Y	YPRG-4000_UPD.01~M, ~Y	int97_d, int97_m, int97_y	CV_INTERVIEW_DATE_D, _M, _Y
ps40002M, ps40002Y	YPRG-4000_UPD.02~M, ~Y	ui1_spells80-ui1_spells98	CV_UI_SPELLS_YR.80-.98
ps40003M, ps40003Y	YPRG-4000_UPD.03~M, ~Y	ui1_ever	CV_UI_WKS_EVER
ps42001-ps42002	YPRG-4200_UPD.01, .02	ui1total80-ui1total98	CV_UI_WKS_YR.80-.98
ps43001-ps43002	YPRG-4300_UPD.01, .02	uia2total97-uia2total99	CV_AMT_UI_YR.97-.99
ps44001-ps44002	YPRG-4400_UPD.01, .02	int98_d, int98_m, int98_y	CV_INTERVIEW_DATE~D, ~M, ~Y
ps45001-ps45002	YPRG-4500_UPD.01, .02	ui2_spells80-ui2_spells99	CV_UI_SPELLS_YR.80-.99
ps47001M, ps47001Y	YPRG-4700_UPD.01~M, ~Y	ui2_spells99	CV_UI_SPELLS_YR.99
ps47002M, ps47002Y	YPRG-4700_UPD.02~M, ~Y	ui2_ever	CV_UI_EVER
ps47003M, ps47003Y	YPRG-4700_UPD.03~M, ~Y	ui2total80-ui2total99	CV_UI_YR.80-.99
ps49001-ps49002	YPRG-4900_UPD.01, .02	uia3total00	CV_AMT_UI_YR.00
ps50001-ps50002	YPRG-5000_UPD.01, .02	uia3total90	CV_AMT_UI_YR.90
ps52001-ps52003	YPRG-5200_UPD.01-.03	uia3total98	CV_AMT_UI_YR.98
ps59001-ps59003	YPRG-5900_UPD.01-.03	uia3total99	CV_AMT_UI_YR.99
ps60001	YPRG-6000_UPD.01	int99_d, int99_m, int99_y	CV_INTERVIEW_DATE~D, ~M, ~Y

ps6200B1-ps6200B3 p35001	YPRG-6200B_UPD.01-.03 YPRG-3500	ui3_spells80-ui3_spells00 ui3_ever	CV_UI_SPELLS_YR.80-.00 CV_UI_EVER
p40001M, p40001Y	YPRG-4000.01~M, ~Y	ui3total80-ui3total00	CV_UI_YR.80-.00
p40002M, p40002Y	YPRG-4000.02~M, ~Y	PUBID	PUBID
p42001, p42002	YPRG-4200.01, .02	uia4total97-uia4total01	CV_AMT_UI_YR.97-.01
p43001	YPRG-4300.01	int00_d, int00_m, int00_y	CV_INTERVIEW_DATE~D, ~M, ~Y
p44001, p44002	YPRG-4400.01, .02	ui4_spells80-ui4_spells00	CV_UI_SPELLS_YR.80-.00
p45001, p45002	YPRG-4500.01, .02	ui4_ever	CV_UI_EVER
p47001M, p47001Y	YPRG-4700.01~M, ~Y	ui4total80-ui4total01	CV_UI_YR.80-.01
p49001	YPRG-4900.01		

**Worker's Compensation**

p010	YPRG-010	ps124001	YPRG-12400_UPD.01
p1210	YPRG-1210	ps12500A1	YPRG-12500A_UPD.01
ps9780	YPRG-9780_UPD	ps126001	YPRG-12600_UPD.01
ps9790	YPRG-9790_UPD	bdate_d, bdate_m, bdate_y	KEY!BDATE_D, _M, _Y
ps9800	YPRG-9800_UPD	wca1total80-wca1total98	CV_AMT_WC_YR.80-.98
ps9810	YPRG-9810_UPD	int97_D, int97_M, int97_Y	CV_INTERVIEW_DATE_D, _M, _Y
ps9820	YPRG-9820_UPD	wca2total81-wca2total99	CV_AMT_WC_YR.81-.99
ps9830	YPRG-9830_UPD	int98_D, int98_M, int98_Y	CV_INTERVIEW_DATE~D, ~M, ~Y
ps9840	YPRG-9840_UPD	wca3total96-wca3total00	CV_AMT_WC_YR.96-.00
ps9850	YPRG-9850_UPD	int99_D, int99_M, int99_Y	CV_INTERVIEW_DATE~D, ~M, ~Y
ps9860	YPRG-9860_UPD	wca4total81-wca4total01	CV_AMT_WC_YR.81-.00
ps9865	YPRG-9865_UPD	int00_D, int00_M, int00_Y	CV_INTERVIEW_DATE~D, ~M, ~Y
ps9867	YPRG-9867_UPD	p2000	YPRG-2000
ps9870	YPRG-9870_UPD	p9550_ELIG_CHECK	YPRG-9550-ELIG-CHECK
ps9875	YPRG-9875_UPD	p9600	YPRG-9600
ps9880M, ps9880Y	YPRG-9880_UPD~M, ~Y	p9700	YPRG-9700
ps9890	YPRG-9890_UPD	p102001M, p102001Y	YPRG-10200.01~M, ~Y
ps9900	YPRG-9900_UPD	p103001	YPRG-10300.01
ps10100X1	YPRG-10100X_UPD.01	p104001	YPRG-10400.01
ps10100A1	YPRG-10100A_UPD.01	p105001	YPRG-10500.01
ps102001M, ps102001Y	YPRG-10200_UPD.01~M	p106001	YPRG-10600.01
ps103001	YPRG-10300_UPD.01	p107001	YPRG-10700.01
ps104001	YPRG-10400_UPD.01	p107901	YPRG-10790.01
ps105001	YPRG-10500_UPD.01	p109001M, p109001Y	YPRG-10900.01~M, ~Y
ps106001	YPRG-10600_UPD.01	p110001	YPRG-11000.01
ps107001	YPRG-10700_UPD.01	p111001	YPRG-11100.01
ps109001M, ps109001Y	YPRG-10900_UPD.01~M	p112001	YPRG-11200.01
ps110001	YPRG-11000_UPD.01	p114001	YPRG-11400.01
ps111001	YPRG-11100_UPD.01	p116001	YPRG-11600.01
ps112001	YPRG-11200_UPD.01	p117001	YPRG-11700.01
ps113001	YPRG-11300_UPD.01	p118001	YPRG-11800.01
ps114001	YPRG-11400_UPD.01	p121001	YPRG-12100.01
ps116001	YPRG-11600_UPD.01	p123001	YPRG-12300.01
ps117001	YPRG-11700_UPD.01	p124001	YPRG-12400.01
ps118001	YPRG-11800_UPD.01	p126001	YPRG-12600.01
ps121001	YPRG-12100_UPD.01	p12700_LOOP_end_1	YPRG-12700-LOOP-end.01
ps122001	YPRG-12200_UPD.01	PUBID	PUBID
ps123001	YPRG-12300_UPD.01	YINT_D, YINT_M, YINT_Y	SYMBOL!CURDATE~D, ~M, ~Y

**Government Assistance**

ps21395	YPRG-21395_UPD	ps2350011-ps2350015	YPRG-23500_UPD.01~000001- ~000005
ps21397	YPRG-21397_UPD	ps2350021-ps2350025	YPRG-23500_UPD.02~000001- ~000005
ps21420	YPRG-21420_UPD	ps2350031-ps2350035	YPRG-23500_UPD.03~000001- ~000005
ps21430	YPRG-21430_UPD	p2350011-p2350015	YPRG-23500.01~000001~000005

ps21900A1	YPRG-21900A_UPD.01	p2350021-p2350025	YPRG-23500.02~000001--000005
ps220001M-ps220001Y	YPRG-22000_UPD.01~M, ~Y	p2350031-p2350035	YPRG-23500.03~000001--000005
ps220002M-ps220002Y	YPRG-22000_UPD.02~M, ~Y	p2350041-p2350045	YPRG-23500.04~000001--000005
ps220003M-ps220003Y	YPRG-22000_UPD.03~M, ~Y	ps2090011-ps2090015	YPRG-20900_UPD.01~000001- ~000005
ps222001 p-222003	YPRG-22200_UPD.01-.03	ps2090021-ps2090025	YPRG-20900_UPD.02~000001- ~000005
ps223001-ps223003	YPRG-22300_UPD.01-.03	ps2090031-ps2090035	YPRG-20900_UPD.03~000001- ~000005
ps226001M-ps226001Y	YPRG-22600_UPD.01~M, ~Y	p2090011-p2090015	YPRG-20900.01~000001--000005
ps226002M-ps226002Y	YPRG-22600_UPD.02~M, ~Y	ps3250011-ps3250015	YPRG-32500_UPD.01~000001- ~000005
ps228001-ps228002	YPRG-22800_UPD.01, .02	ps3250021-ps3250025	YPRG-32500_UPD.02~000001- ~000005
ps233001-ps233003	YPRG-23300_UPD.01-.03	p3250011-p3250015	YPRG-32500.01~000001--000005
ps234001-ps234002	YPRG-23400_UPD.01, .02	p3250021-p3250025	YPRG-32500.02~000001--000005
ps23700A1-ps23700A3	YPRG-23700A_UPD.01- .03	ps3592011-ps3592015	YPRG-35920_UPD.01~000001- ~000005
ps238001-ps238002	YPRG-23800_UPD.01, .02	ps3592021-ps3592025	YPRG-35920_UPD.02~000001- ~000005
p21300_ELIG_CHECK	YPRG-21300-ELIG- CHECK	ps3592031-ps3592035	YPRG-35920_UPD.03~000001- ~000005
p21400	YPRG-21400	p1830011-p1830015	YPRG-18300.01~000001--000005
p21500	YPRG-21500	p1830021-p1830025	YPRG-18300.02~000001--000005
p220001M-p220001Y	YPRG-22000.01~M, ~Y	YINT_D Y,INT_M, YINT_Y	SYMBOL!CURDATE~D, ~M, ~Y
p220002M-p220002Y	YPRG-22000.02~M, ~Y	BDATE_D, BDATE_M, BDATE_Y	KEY!BDATE_D, _M, _Y
p220003M-p220003Y	YPRG-22000.03~M, ~Y	ga1total80-ga1total98	CV_AMT_GOVNT_PGM_PCY.80-.98
p220004M-p220004Y	YPRG-22000.04~M, ~Y	G1PRG_EVER	CV_GOVNT_PRG_EVER
p222001-p222004	YPRG-22200.01-.04	g1total80-g1total98	CV_GOVNT_PRG_YR.80-.98
p223001-p223004	YPRG-22300.01-.04	INT97_D, INT97_M, INT97_Y	CV_INTERVIEW_DATE_D, _M, _Y
p223901-p223904	YPRG-22390.01-.04	ga2total80-ga2total99	CV_AMT_GOVNT_PGM_PCY.80-.99
p226001M-p226001Y	YPRG-22600.01~M, ~Y	G2PRG_EVER	CV_GOVNT_PRG_EVER
p226002M-p226002Y	YPRG-22600.02~M, ~Y	g2total80-g2total99	CV_GOVNT_PRG_YR.80-.99
p226003M-p226003Y	YPRG-22600.03~M, ~Y	INT98_D, INT98_M, INT98_Y	CV_INTERVIEW_DATE~D, ~M, ~Y
p226004M-p226004Y	YPRG-22600.04~M, ~Y	ga3total80-ga3total100	CV_AMT_GOVNT_PGM_PCY.80-.100
p228001-p228004	YPRG-22800.01-.04	G3PRG_EVER	CV_GOVNT_PRG_EVER
p233001-p233004	YPRG-23300.01-.04	g3total80-g3total100	CV_GOVNT_PRG_YR.80-.100
p234001-p234004	YPRG-23400.01-.04	INT99_D, INT99_M, INT99_Y	CV_INTERVIEW_DATE~D, ~M, ~Y
p236001-p236004	YPRG-23600.01-.04	ga4total80-ga4total101	CV_AMT_GOVNT_PGM_PCY.80-.101
p238001-p238004	YPRG-23800.01-.04	G4PRG_EVER	CV_GOVNT_PRG_EVER
ps16375	YPRG-16375_UPD	g4total80-g4total101	CV_GOVNT_PRG_YR.80-.101
ps16377	YPRG-16377_UPD	INT00_D, _M, _Y	CV_INTERVIEW_DATE~D, ~M, ~Y
ps16410	YPRG-16410_UPD	p18700_ELIG_CHECK	YPRG-18700-ELIG-CHECK
ps19063A1	YPRG-19063A_UPD.01	p30500	YPRG-30500
ps191001M-ps191001Y	YPRG-19100_UPD.01~M, ~Y	p310001M-p310001Y	YPRG-31000.01~M, ~Y
ps191002M-ps191002Y	YPRG-19100_UPD.02~M, ~Y	p310002M-p310002Y	YPRG-31000.02~M, ~Y
ps191003M-ps191003Y	YPRG-19100_UPD.03~M, ~Y	p312001-p312002	YPRG-31200.01, .02
ps191401-ps191403	YPRG-19140_UPD.01-.03	p313001-p313002	YPRG-31300.01, .02
ps197001-ps197003	YPRG-19700_UPD.01-.03	p313901-p313902	YPRG-31390.01, .02
ps200001M-ps200001Y	YPRG-20000_UPD.01~M, ~Y	p316001M-p316001Y	YPRG-31600.01~M, ~Y
ps200002M-ps200002Y	YPRG-20000_UPD.02~M, ~Y	p318001	YPRG-31800.01
ps202001-ps202002	YPRG-20200_UPD.01-.02	p323001-p323002	YPRG-32300.01, .02

ps207001-ps207003	YPRG-20700_UPD.01-.03	p324001	YPRG-32400.01
ps208001-ps208002	YPRG-20800_UPD.01-.02	p329001-p329002	YPRG-32900.01, .02
ps21100A1-ps21100A3	YPRG-21100A_UPD.01-.03	p010	YPRG-010
ps212001-ps212002	YPRG-21200_UPD.01-.02	p1210	YPRG-1210
p2500	YPRG-2500	ps35707	YPRG-35707_UPD
p16010_ELIG_CHECK	YPRG-16010-ELIG-CHECK	ps35740	YPRG-35740_UPD
p18800	YPRG-18800	ps357901	YPRG-35790_UPD.01
p18900	YPRG-18900	ps358001M-ps358001Y	YPRG-35800_UPD.01~M, ~Y
p194001M-p194001Y	YPRG-19400.01~M, ~Y	ps358002M-ps358002Y	YPRG-35800_UPD.02~M, ~Y
p196001	YPRG-19600.01	ps358003M-ps358003Y	YPRG-35800_UPD.03~M, ~Y
p197001	YPRG-19700.01	ps358201-ps358202	YPRG-35820_UPD.01-.02
p197901	YPRG-19790.01	ps358301-ps358302	YPRG-35830_UPD.01-.02
p200001M-p200001Y	YPRG-20000.01~M, ~Y	ps358401M-ps358401Y	YPRG-35840_UPD.01~M, ~Y
p202001	YPRG-20200.01	ps358402M-ps358402Y	YPRG-35840_UPD.02~M, ~Y
p210001	YPRG-21000.01	ps358403M-ps358403Y	YPRG-35840_UPD.03~M, ~Y
ps36087	YPRG-36087_UPD	ps358601-ps358603	YPRG-35860_UPD.01-.03
ps36120	YPRG-36120_UPD	ps359001-ps359003	YPRG-35900_UPD.01-.03
ps30900A1	YPRG-30900A_UPD.01	ps359101-ps359102	YPRG-35910_UPD.01, .02
ps310001M-ps310001Y	YPRG-31000_UPD.01~M, ~Y	ps359501-ps359503	YPRG-35950_UPD.01-.03
ps310002M-ps310002Y	YPRG-31000_UPD.02~M, ~Y	ps359601-ps359603	YPRG-35960_UPD.01-.03
ps312001-ps312002	YPRG-31200_UPD.01, .02	p16200	YPRG-16200
ps313001-ps313002	YPRG-31300_UPD.01, .02	p167001M-p167001Y	YPRG-16700.01~M, ~Y
ps316001M-ps316001Y	YPRG-31600_UPD.01~M, ~Y	p167002M-p167002Y	YPRG-16700.02~M, ~Y
ps316002M-ps316002Y	YPRG-31600_UPD.02~M, ~Y	p169001-p69002	YPRG-16900.01-.02
ps318001	YPRG-31800_UPD.01	p170001-p170002	YPRG-17000.01-.02
ps323001-ps323002	YPRG-32300_UPD.01, .02	p170901-p170902	YPRG-17090.01-.02
ps324001	YPRG-32400_UPD.01	p172001M-p172001Y	YPRG-17200.01~M, ~Y
ps32700A1-ps32700A2	YPRG-32700A_UPD.01, .02	p175001	YPRG-17500.01
ps329001-ps329002	YPRG-32900_UPD.01, .02	p181001-p181002	YPRG-18100.01-.02
PUBID	PUBID	p186001	YPRG-18600.01

This program creates several variables describing the respondent's participation in government programs for the economically disadvantaged. During the interview, respondents report amounts received and months of participation in Aid to Families with Dependent Children (AFDC); food stamps; and Women, Infants, and Children (WIC). There is also an "other assistance" question to capture information about any other government program from which respondents may have received assistance. The interview also records amounts received from unemployment compensation and worker's compensation in separate question series; this program also creates variables describing respondents' participation in unemployment or worker's compensation.

The program to create these variables first creates a month-by-month participation array for five of the six categories (AFDC, food stamps, WIC, other programs, and unemployment compensation; the array is not created for worker's compensation). These month-by-month variables constitute part of the event history array for program participation; see [appendix 7](#) for more information. After all five arrays are created, the program merges data to create the summary variables.

Because worker's compensation can be reported either as a lump sum received on a single day or as payments over the period of time that the respondent was not working, the month-by-month array variables are no longer created for this category of program participation. The total benefits received in each calendar year are calculated, but the weeks and months that compensation were received are not

created.

---

**\*\*\*\*\* SECTION 1: UNEMPLOYMENT COMPENSATION \*\*\*\*\***

**/\*\* HANDEDITS - 1 \*\*/**

**/\*\* This person reports no stop date or currently receiving (ps19301=1). But in a followup question (ps45001) about whether there was any period of two weeks or more that this person had not received, this person answers no. This implies that this person is currently receiving.\*\*/**  
**if pubid=7523 then do; ps19301=1; CURR\_REC\_PROB=1; end;**

**/\*\* All the arrays used in this program. \*\*/**

**array intd (4) int00\_d int99\_d int98\_d int97\_d;**  
**array intm (4) int00\_m int99\_m int98\_m int97\_m;**  
**array inty (4) int00\_y int99\_y int98\_y int97\_y;**

**array uitotalpr (4,23) ui4total80-ui4total99 ui4total00 ui4total01 ui4total02**  
**ui3total80-ui3total99 ui3total00 ui3total01 ui3total02**  
**ui2total80-ui2total99 ui2total00 ui2total01 ui2total02**  
**ui1total80-ui1total99 ui1total00 ui1total01 ui1total02;**

**array uiatotalpr (4,23) uia4total80-uia4total99 uia4total00 uia4total01 uia4total02**  
**uia3total80-uia3total99 uia3total00 uia3total01 uia3total02**  
**uia2total80-uia2total99 uia2total00 uia2total01 uia2total02**  
**uia1total80-uia1total99 uia1total00 uia1total01 uia1total02;**

**array ui\_spellspr (4,23) ui4\_spells80-ui4\_spells99 ui4\_spells00 ui4\_spells01 ui4\_spells02**  
**ui3\_spells80-ui3\_spells99 ui3\_spells00 ui3\_spells01 ui3\_spells02**  
**ui2\_spells80-ui2\_spells99 ui2\_spells00 ui2\_spells01 ui2\_spells02**  
**ui1\_spells80-ui1\_spells99 ui1\_spells00 ui1\_spells01 ui1\_spells02;**

**array uitotal (23) uitotal80-uitotal99 uitotal00 uitotal01 uitotal02;**  
**array uiatotal (23) uiatotal80-uiatotal99 uiatotal00 uiatotal01 uiatotal02;**  
**array ui\_spells (23) ui\_spells80-ui\_spells99 ui\_spells00 ui\_spells01 ui\_spells02;**  
**array nu (23) nu80-nu99 nu00 nu01 nu02;**

**array olduitotal (23) olduitotal80-olduitotal99 olduitotal00 olduitotal01 olduitotal02;**  
**array olduiatotal (23) olduiatotal80-olduiatotal99 olduiatotal00 olduiatotal01 olduiatotal02;**  
**array oldui\_spells (23) oldui\_spells80-oldui\_spells99 oldui\_spells00 oldui\_spells01 old\_uispells02;**  
**array CURRnu (23) CURRnu80-CURRnu99 CURRnu00 CURRnu01 CURRnu02;**

**array um (23) um80-um99 um00 um01 um02;**  
**array uamt (23) uamt80-uamt99 uamt00 uamt01 uamt02;**  
**array ua (23) ua80-ua99 ua00 ua01 ua02;**  
**array un (23) un80-un99 un00 un01 un02;**  
**array ui (269) ui001-ui269;**  
**array uiamt (269) uiamt001-uiamt269;**  
**array nms (2,3) nmsu11-nmsu13 nmsu21-nmsu23;**  
**array nme (2,3) nmeu11-nmeu13 nmeu21-nmeu23;**  
**array yms (2,3) ymsu11-ymsu13 ymsu21-ymsu23;**  
**array csm (2,3) csmu11-csmu13 csmu21-csmu23;**

```

array yme      (2,3) ymeu11-ymeu1 ymeu21-ymeu23;
array cem      (2,3) cem11-cemu13 cemu21-cemu23;
array s_day    (2,3) sdayu11-sdayu13 sdayu21-sdayu23;
array e_day    (2,3) edayu11-edayu13 edayu21-edayu23;
array eflag    (2,3) eflagu11-eflagu13 eflagu21-eflagu23;
array aflag    (2,3) aflagu11-aflagu13 aflagu21-aflagu23;
array mos      (2,3) mosu11-mosu13 mosu21-mosu23;
array agefl    (2,3) ageflu11-ageflu13 ageflu21-ageflu23;
array ymsmo    (2,3) ymsmo11-ymsmo13 ymsmo21-ymsmo23;
array ymemo    (2,3) ymemo11-ymemo13 ymemo21-ymemo23;
array newms    (2,3) newms11-newms13 newms21-newms23;
array newme    (2,3) newme11-newme13 newme21-newme23;

```

```

/** start YEAR, MONTH, ACTUAL DAY and ESTIMATED DAY
information.          **/

```

```

array ys      (2,3) ps40001Y ps40002Y ps40003Y p40001Y p40002Y p40003Y;
array ms      (2,3) ps40001M ps40002M ps40003M p40001M p40002M p40003M;
array ds      (2,3) ps42001 ps42002 ps42003 p42001 p42002 p42003;
array es      (2,3) ps43001 ps43002 ps43003 p43001 p43002 p43003;

```

```

/** end YEAR, MONTH, ACTUAL DAY and ESTIMATED DAY
information.          **/

```

```

array ye      (2,3) ps47001Y ps47002Y ps47003Y p47001Y p47002Y p47003Y;
array me      (2,3) ps47001M ps47002M ps47003M p47001M p47002M p47003M;
array de      (2,3) ps49001 ps49002 ps49003 p49001 p49002 p49003;
array ee      (2,3) ps50001 ps50002 ps50003 p50001 p50002 p50003;

```

```

/** ESTIMATED WEEKS - unable to answer start or stop date questions.          **/
array wks     (2,3) ps52001 ps52002 ps52003 p52001 p52002 p52003;

```

```

/** CURRENTLY receiving information:

```

For SDLI sections:

If start data (ps40001m etc.) reported is a valid date then currently receiving information questions (p44001 etc.) are asked. However if the start date is Refused/Dont Know or missing then it is not asked for that LOOP. This is specific to each LOOP. However currently receiving information question ps19301 is asked to all those who say they have received Unemp SDLI.

For NEVER PRIOR sections:

Since respondents are routed through different branches some are asked the currently receiving information in PS1897 and others in p35001. (The information from ps1897 and p35001 are collected into p35001 later in the program). Then they all get routed to the start date questions (p40001m etc.) and then if they answer this question with some valid dates they later get routed to the currently receiving information question for that LOOP (p44001 etc.) otherwise not. Again this is also LOOP specific.

```

**/

```

```

array cur     (2,3) ps44001 ps44002 ps44003 p44001 p44002 p44003;
array cwks    (2,3) ps19301 ps19302 ps19303 p35001 p35002 p35003;

```

```

/** Receiving at DLI - using ps18501 instead of ps18301 as the latter is a computer check from last
interview and the former is the follow up qestion that verifies that information.          **/

```

```

array dli      (2,3) ps18501  ps18502  ps18503  p18501  p18502  p18503;

/** AMOUNT RECEIVED - actual & estimated.                                **/
array inc      (2,3) ps59001  ps59002  ps59003  p59001  p59002  p59003;
array einc     (2,3) ps60001  ps60002  ps60003  p60001  p60002  p60003;

** Part 1. Creating certain basic variables
*****;
/** change weekly ui figures to daily amount.                            **/

do K=1 to DIM(inc,1);
  do J=1 to DIM(inc,2);
    if inc(K,J) gt 0 then inc(K,J)=inc(K,J)/7;
  end;
end;

/** change previous round variables to account for noninterviews -
  If r4 interview, use r4 created var. If noninterview in r4, then use r3 variable.
  If noninterview in r3, then use r2 variable. If noninterview in r2, then use r1 variable
  If variable not in r1, r2, r3 or r4 then set to 0.                        **/

do Q=1 to DIM(inty);
  if inty(Q) ne -5 then do;
    do I=1 to DIM(uitotal);
      uitotal(I)=uitotalpr(Q,I);
      uiatotal(I)=uiatotalpr(Q,I);
      ui_spells(I)=ui_spellspr(Q,I);
    end;
    Q=DIM(inty);
  end;
end;

do I=1 to DIM(uitotal);
  olduitotal(I)=uitotal(I);
  olduiatotal(I)=uiatotal(I);
  oldui_spells(I)=ui_spells(I);
  nu(I)=0;
  if uitotal(I)=. then do;
    olduitotal(I)=-4;
    uitotal(I)=0;
  end;
  if uiatotal(I)=. then do;
    olduiatotal(I)=-4;
    uiatotal(I)=0;
  end;
  if ui_spells(I)=. then do;
    oldui_spells(I)=-4;
    ui_spells(I)=0;
  end;
end;
end;

```

```

ui_ever=ui4_ever;
if int00_y=-5 then do;
  ui_ever=ui3_ever;
  if int99_y=-5 then do;
    ui_ever=ui2_ever;
    if int98_y=-5 then do;
      ui_ever=ui1_ever;
    end;
  end;
end;
if ui_ever=-4 then do;
  ui_ever=0;
end;

/** Initialize the following counters to zero.                **/
/** Counter for UIAMT ge 0 (receiving positive amounts) but UI=0 (not receiving).                **/
/** Counter for UI gt 0 (receiving) but UIAMT is missing (amount receiving info not available) **/
/** Total number of months ever received.                    **/
/** Month counter for correction of problematic amounts reported.                **/

uick=0;
uickm=0;
uickmm=0;
ttl=0;
uimnths=0;
SAMEMONTH=0;

/** Creating variables relating to date of birth, age, age 14 etc.                **/

cm14=((bdate_y+14)-1980)*12+bdate_m;
ym14=(bdate_y*100+bdate_m)+1400;
cmb=((bdate_y)-1980)*12+bdate_m;
doicm=(yint_y-1980)*12+yint_m;
iym=yint_y*100+yint_m;
aaicm=(yint_y-bdate_y)*12+yint_m-bdate_m;
array mon (12) mon1-mon12;
array uiday (12) uiday1-uiday12;

do Z=1 to 12; mon(Z)=Z; end;

do Z=1 to 12;
  if mon(Z)=1 OR mon(Z)=3 OR mon(Z)=5 OR mon(Z)=7 OR mon(Z)=8 OR mon(Z)=10 OR mon(Z)
=12 then do;
    uiday(Z)=31;
  end;
  if mon(Z)=2 then do;
    uiday(Z)=28;
  end;
  if mon(Z)=4 OR mon(Z)=6 OR mon(Z)=9 OR mon(Z)=11 then do;
    uiday(Z)=30;
  end;
end;

```



```
end;  
end;
```

```
/** Creating variables pertaining to the date of last interview.
```

```
*/
```

```
if p1210=1 then do;  
  do Q=1 to DIM(inty);  
    if inty[Q] ne -5 then do;  
      if intm[Q] lt 12 then dliym=(inty[Q]*100)+(intm[Q]+1);  
      if intm[Q]=12 then dliym=((inty[Q]+1)*100)+1;  
      dlicm=((inty[Q]-1980)*12)+(intm[Q]+1);  
      dli_d=intd[Q];  
      dli_m=dliym-round(dliym,100);  
      dli_y=round(dliym,100)/100;  
      Q=DIM(inty);  
    end;  
  end;  
end;
```

```
if p1210 ne 1 then do;  
  dliym=ym14;  
  dli_d=bdate_d;  
  dli_m=bdate_m;  
  dli_y=bdate_y;  
  dlicm=cm14;  
end;
```

```
do Z=1 to 12;  
  if yint_m=mon(Z) then intday=uiday(Z)+1-yint_d;  
  if dli_m=mon(Z) then ddliday=uiday(Z)+1-dli_d;  
end;
```

```
if dli_m=1 OR dli_m=8 then do;  
  dliday=(31-dli_d+31);  
  dliday1=31-dli_d;  
  dliday2=31;  
end;
```

```
if dli_m=3 then do;  
  dliday=(28-dli_d+31);  
  dliday1=28-dli_d;  
  dliday2=31;  
end;
```

```
if dli_m=5 OR dli_m=7 OR dli_m=10 OR dli_m=12 then do;  
  dliday=(30-dli_d+31);  
  dliday1=30-dli_d;  
  dliday2=31;  
end;
```

```
if dli_m=4 OR dli_m=6 OR dli_m=9 OR dli_m=11 then do;  
  dliday=(31-dli_d+30);  
  dliday1=31-dli_d;  
  dliday2=30;
```

```

end;
if dli_m=2 then do;
  dliday=(31-dli_d+28);
  dliday1=31-dli_d;
  dliday2=28;
end;

```

```

ps19302=ps19301;
ps19303=ps19301;
p35002=p35001;
p35003=p35001;

```

**\*\* Part 2. Creating variables using Round 5 information**

\*\*\*\*\*;

```

if p1210>-4 then do; /** [B1] **/

```

```

  /** Combining currently receiving information for the NEVER PRIOR sections. **/

```

```

  if ps1897>-3 then do;
    p35001=ps1897;
    p35002=p35001;
    p35003=p35001;
  end;

```

```

  /** Initializing all ui from the month of birth to the date of current interview to zero. **/

```

```

  do L=1 to DIM(ui);
    if cmb le L le doicm then do;
      ui(L)=0;
    end;
  end;

```

/\*\* This portion of the SAS program defines the start date and end dates. If the respondent reports still receiving, the interview date is used as the temporary end date for the last loop reported. In the next survey round, the respondent will be asked if he or she is still receiving and, if not, a permanent end date equivalent to the interview date of the previous round will be assigned. Users will be able to tell which method was used by looking at the following participation flag variable created during the program. The categories are the following:

- 1=respondent reported participation dates
- 2=start month imputed
- 3=start month and year imputed
- 4=stop month imputed
- 5=stop month and year imputed
- 6=start and stop dates imputed
- 7=error in data due to round 2 programming error \*\*\*\*\*/

```

do K=1 to DIM(csm,1); /** [B2] **/
  do J=1 to DIM(csm,2); /** [B3] **/

```

```

/** 1. If start and stop dates are all valid **/
  if ys(K,J)>0 and ms(K,J)>0 then do;
    yms(K,J)=(ys(K,J)*100)+ms(K,J);

```

```

/** set start day, if given.                                **/
if ds(K,J)>0 OR es(K,J)>0 then do;
  if ms(K,J)=1 OR ms(K,J)=3 OR ms(K,J)=5 OR ms(K,J)=7 OR ms(K,J)=8 OR ms(K,J)=10 OR
ms(K,J)=12
  then do;
    if ds(K,J)>0 and ds(K,J) ne . then s_day(K,J)=(32-ds(K,J));
    if es(K,J)=1 then s_day(K,J)=31;
    if es(K,J)=2 then s_day(K,J)=21;
    if es(K,J)=3 then s_day(K,J)=11;
  end;
  if ms(K,J)=4 OR ms(K,J)=6 OR ms(K,J)=9 OR ms(K,J)=11 then do;
    if ds(K,J)>0 and ds(K,J) ne . then s_day(K,J)=(31-ds(K,J));
    if es(K,J)=1 then s_day(K,J)=30;
    if es(K,J)=2 then s_day(K,J)=20;
    if es(K,J)=3 then s_day(K,J)=10;
  end;
  if ms(K,J)=2 then do;
    if ds(K,J)>0 and ds(K,J) ne . then s_day(K,J)=(29-ds(K,J));
    if es(K,J)=1 then s_day(K,J)=28;
    if es(K,J)=2 then s_day(K,J)=18;
    if es(K,J)=3 then s_day(K,J)=8;
  end;
end;
end;

```

```

/** set start date to DLI if receiving at DLI.              **/
if dli(K,J)=1 then do;
  yms(K,J)=dliym;
  s_day(K,J)=dliday;
  if ps19301=1 then do;
    yme(K,J)=iym;
    e_day(K,J)=yint_d;
    NOGAPS=1;
  end;
end;
end;

```

```

/** set start date to DLI if start date is before DLI.     **/
if yms(K,J) ne . and yms(K,J)<dliym then do;
  yms(K,J)=dliym;
  s_day(K,J)=dliday;
end;

```

```

/** set end date to current int date if receiving now.     **/
if cur(K,J)=1 then do;
  yme(K,J)=iym;
  e_day(K,J)=yint_d;
end;

```

```

/** set end date if both end month and year are valid.     **/
if ye(K,J)>0 and me(K,J)>0 then do;
  yme(K,J)=(ye(K,J)*100)+me(K,J);

```

```

/** set end day, if given.                                **/
if de(K,J)>0 and de(K,J) ne . then do;
  e_day(K,J)=de(K,J);
end;

```

```

/** end day unknown, estimate given.                      **/
if de(K,J)<0 and ee(K,J)>0 and ee(K,J) ne . then do;
  if ee(K,J)=1 then e_day(K,J)=9;
  if ee(K,J)=2 then e_day(K,J)=19;
  if ee(K,J)=3 then e_day(K,J)=28;
end;
end;

```

```

/** if end month greater than int month, constrain by int month. **/
if yme(K,J)>iym then do;
  yme(K,J)=iym;
  e_day(K,J)=yint_d;
end;
if yms(K,J)>0 and yme(K,J) ge dliym then eflag(K,J)=1;

```

```

/** determine months from estimated weeks variable.      **/
if wks(K,J) ge 0 and (ys(K,J)<0 OR ms(K,J)<0 OR ye(K,J)<0 OR me(K,J)<0) then do;
  mos(K,J)=floor(wks(K,J)/4.3);
end;
agefl(K,J)=0;
if ys(K,J)>0 and ys(K,J)<1990 then agefl(K,J)=1;

```

```

/** 2. If start year is known and month is unknown
Note: if wks ge -3 then it means that stop date is not known. **/
if ys(K,J)>0 and -3 le ms(K,J) le -1 then do;

```

/\*\* 2.a If weeks are known and currently receiving, then count backwards by the number of weeks from the

current interview date. If the number of weeks falls short of the start year, the start month is December of that year. If the number of weeks is past the start year, then the start month is January of that year. \*\*/

```

if (wks(K,J) ge 0 and cwks(K,J)=1) then do;
  yme(K,J)=iym;
  e_day(K,J)=yint_d;
  if mos(K,J) le yint_m then yms(K,J)=yme(K,J)- mos(K,J) ;
  if (yint_m+ 0) le mos(K,J) le (yint_m+ 12) then yms(K,J)=yme(K,J)- 100-(mos(K,J)- 12);
  if (yint_m+12) le mos(K,J) le (yint_m+ 24) then yms(K,J)=yme(K,J)- 200-(mos(K,J)- 24);
  if (yint_m+24) le mos(K,J) le (yint_m+ 36) then yms(K,J)=yme(K,J)- 300-(mos(K,J)- 36);
  if (yint_m+36) le mos(K,J) le (yint_m+ 48) then yms(K,J)=yme(K,J)- 400-(mos(K,J)- 48);
  if (yint_m+48) le mos(K,J) le (yint_m+ 60) then yms(K,J)=yme(K,J)- 500-(mos(K,J)- 60);
  if (yint_m+60) le mos(K,J) le (yint_m+ 72) then yms(K,J)=yme(K,J)- 600-(mos(K,J)- 72);
  if (yint_m+72) le mos(K,J) le (yint_m+ 84) then yms(K,J)=yme(K,J)- 700-(mos(K,J)- 84);
  if (yint_m+84) le mos(K,J) le (yint_m+ 96) then yms(K,J)=yme(K,J)- 800-(mos(K,J)- 96);
  if (yint_m+96) le mos(K,J) le (yint_m+108) then yms(K,J)=yme(K,J)- 900-(mos(K,J)-108);
  if (yint_m+108) le mos(K,J) le (yint_m+120) then yms(K,J)=yme(K,J)-1000-(mos(K,J)-120);

```

```

do Z=1 to 12;
  if (yms(K,J)-(round(yms(K,J), 100)))=mon(Z) then s_day(K,J)=uiday(Z);
end;

if yms(K,J)<((ys(K,J)*100)+01) then do;
  yms(K,J)=((ys(K,J)*100)+01);
  s_day(K,J)=31;
end;

if yms(K,J)<dliym then do;
  yms(K,J)=dliym;
  s_day(K,J)=ddliday;
end;

if yms(K,J)>((ys(K,J)*100)+12) then do;
  yms(K,J)=((ys(K,J)*100)+12);
  s_day(K,J)=31;
end;
end;

```

```

/** 2.b weeks missing and currently receiving - set end date to current interivew date and start date
to December of start year and constrain start date by Current Interview Date and DLI.          **/
else if (-3 le wks(K,J) lt 0 and cwks(K,J)=1) then do;

```

```

  yme(K,J)=iym;
  e_day(K,J)=yint_d;
  yms(K,J)=((ys(K,J)*100)+12);
  s_day(K,J)=31;

```

```

  if yms(K,J)>iym then do;
    yms(K,J)=iym;
    s_day(K,J)=intday;
  end;

```

```

  if yms(K,J)<dliym then do;
    yms(K,J)=dliym;
    s_day(K,J)=ddliday;
  end;

```

```

end;

```

```

/** 2.c weeks missing and not currently receiving - set start date to January of start year and end date
to December of start year. Constrain start date by DLI and end date by Current Interview Date.
**/

```

```

else if (-3 le wks(K,J) lt 0 and cwks(K,J)=0) then do;

```

```

  yms(K,J)=((ys(K,J)*100)+01);
  s_day(K,J)=31;
  yme(K,J)=((ys(K,J)*100)+12);
  e_day(K,J)=31;

```

```

if yms(K,J)<dliym then do;
  yms(K,J)=dliym;
  s_day(K,J)=ddliday;
end;

if yme(K,J)>iym then do;
  yme(K,J)=iym;
  e_day(K,J)=yint_d;
end;
end;

```

/\*\* 2.d If weeks are known and not currently receiving, then count forward by the number of weeks from January of the start year. Constrain start date by DLI. If the count exceeds the interview date then stop counting at the interview date. \*/

```

else if (wks(K,J) ge 0 and cwks(K,J)=0) then do;

```

```

  yms(K,J)=((ys(K,J)*100)+01);
  s_day(K,J)=31;
  if yms(K,J)<dliym then do;
    yms(K,J)=dliym;
    s_day(K,J)=ddliday;
  end;

```

```

  ymsmo(K,J)=(yms(K,J)-round(yms(K,J),100))+mos(K,J);
  if 0 le ymsmo(K,J) le 12 then yme(K,J)=yms(K,J)+ 0+mos(K,J)- 0;
  if 13 le ymsmo(K,J) le 24 then yme(K,J)=yms(K,J)+100+mos(K,J)- 12;
  if 25 le ymsmo(K,J) le 36 then yme(K,J)=yms(K,J)+200+mos(K,J)- 24;
  if 37 le ymsmo(K,J) le 48 then yme(K,J)=yms(K,J)+300+mos(K,J)- 36;
  if 49 le ymsmo(K,J) le 60 then yme(K,J)=yms(K,J)+400+mos(K,J)- 48;
  if 61 le ymsmo(K,J) le 72 then yme(K,J)=yms(K,J)+500+mos(K,J)- 60;
  if 73 le ymsmo(K,J) le 84 then yme(K,J)=yms(K,J)+600+mos(K,J)- 72;
  if 85 le ymsmo(K,J) le 96 then yme(K,J)=yms(K,J)+700+mos(K,J)- 84;
  if 97 le ymsmo(K,J) le 108 then yme(K,J)=yms(K,J)+800+mos(K,J)- 96;
  if 109 le ymsmo(K,J) le 120 then yme(K,J)=yms(K,J)+900+mos(K,J)-108;

```

```

  do Z=1 to 12;
    if (yme(K,J)-(round(yme(K,J), 100)))=mon(Z) then e_day(K,J)=uiday(Z);
  end;

```

```

  if yme(K,J)>iym then do;
    yme(K,J)=iym;
    e_day(K,J)=yint_d;
  end;
end;

```

```

if yme(K,J)>0 and CWKS(K,J)=0 then do;
  yms(K,J)=((ys(K,J)*100)+01);
  s_day(K,J)=31;
end;

```

```
if yms(K,J)<dliym then do;
  yms(K,J)=dliym;
  s_day(K,J)=ddliday;
end;
```

```
eflag(K,J)=2;
end;
```

```
/** 3. If start year is unknown. **/
if -3 le ys(K,J) le -1 then do;
```

```
/** 3.a If currently receiving and weeks are known then count back from interview date receiving.
**/
```

```
if (wks(K,J) ge 0 and cwks(K,J)=1) then do;
  if 0 le mos(K,J) le (yint_m+ 0) then yms(K,J)=iym- 0-(mos(K,J)- 0);
  if (yint_m+ 0) le mos(K,J) le (yint_m+ 12) then yms(K,J)=iym- 100-(mos(K,J)- 12);
  if (yint_m+ 12) le mos(K,J) le (yint_m+ 24) then yms(K,J)=iym- 200-(mos(K,J)- 24);
  if (yint_m+ 24) le mos(K,J) le (yint_m+ 36) then yms(K,J)=iym- 300-(mos(K,J)- 36);
  if (yint_m+ 36) le mos(K,J) le (yint_m+ 48) then yms(K,J)=iym- 400-(mos(K,J)- 48);
  if (yint_m+ 48) le mos(K,J) le (yint_m+ 60) then yms(K,J)=iym- 500-(mos(K,J)- 60);
  if (yint_m+ 60) le mos(K,J) le (yint_m+ 72) then yms(K,J)=iym- 600-(mos(K,J)- 72);
  if (yint_m+ 72) le mos(K,J) le (yint_m+ 84) then yms(K,J)=iym- 700-(mos(K,J)- 84);
  if (yint_m+ 84) le mos(K,J) le (yint_m+ 96) then yms(K,J)=iym- 800-(mos(K,J)- 96);
  if (yint_m+ 96) le mos(K,J) le (yint_m+108) then yms(K,J)=iym- 900-(mos(K,J)-108);
  if (yint_m+108) le mos(K,J) le (yint_m+120) then yms(K,J)=iym-1000-(mos(K,J)-120);
```

```
do Z=1 to 12;
  if yms(K,J)-(round(yms(K,J),100))=mon(Z) then s_day(K,J)=uiday(Z);
end;
```

```
yme(K,J)=iym;
e_day(K,J)=yint_d;
if yms(K,J)<dliym then do;
  yms(K,J)=dliym;
  s_day(K,J)=ddliday;
end;
end;
```

```
/** 3.a If currently receiving and weeks are unknown then take DLI as start date and Current Interview
Date
```

```
as stop date. **/
if (-3 le wks(K,J) lt 0 and cwks(K,J)=1) then do;
  yms(K,J)=dliym;
  yme(K,J)=iym;
  s_day(K,J)=ddliday;
  e_day(K,J)=yint_d;
end;
```

```
/** 3.c If weeks are known, then count back from interview date to find the most recent year the
respondent
could have begun receiving and receive for that number of months. Set start date as January of that
```

year. If currently not receiving then count forward the number of months from January of that year and set the stop date. If currently receiving then just set the Current Interview date as the stop date. Constrain start and stop dates by DLI and Current Interview Date, respectively. \*\*/

```

if (wks(K,J) ge 0 and cwks(K,J)=0) then do;
  if 0 le mos(K,J) le yint_m then do;
    yms(K,J)=(yint_y*100)+01;
    yme(K,J)=(yint_y*100)+01+mos(K,J);
  end;
  if (yint_m+01) le mos(K,J) le (yint_m+12) then do;
    yms(K,J)=((yint_y-1)*100)+01;
    if mos(K,J) le 12 then yme(K,J)=((yint_y-1)*100)+01+mos(K,J);
    if mos(K,J) ge 13 then yme(K,J)=(yint_y*100)+01+mos(K,J)-12;
  end;
  if (yint_m+13) le mos(K,J) le (yint_m+24) then do;
    yms(K,J)=((yint_y-2)*100)+01;
    if mos(K,J) le 24 then yme(K,J)=((yint_y-1)*100)+01+mos(K,J)-12;
    if mos(K,J) ge 25 then yme(K,J)=(yint_y*100)+01+mos(K,J)-24;
  end;
  if (yint_m+25) le mos(K,J) le (yint_m+36) then do;
    yms(K,J)=((yint_y-3)*100)+01;
    if mos(K,J) le 36 then yme(K,J)=((yint_y-1)*100)+01+mos(K,J)-24;
    if mos(K,J) ge 37 then yme(K,J)=(yint_y*100)+01+mos(K,J)-36;
  end;
  if (yint_m+37) le mos(K,J) le (yint_m+48) then do;
    yms(K,J)=((yint_y-4)*100)+01;
    if mos(K,J) le 48 then yme(K,J)=((yint_y-1)*100)+01+mos(K,J)-36;
    if mos(K,J) ge 49 then yme(K,J)=(yint_y*100)+01+mos(K,J)-48;
  end;

  s_day(K,J)=31;
  do Z=1 to 12;
    if (yme(K,J)-(round(yms(K,J),100)))=mon(Z) then e_day(K,J)=uiday(Z);
  end;

  if yme(K,J)>iym then do;
    yme(K,J)=iym;
    e_day(K,J)=yint_d;
  end;
end;

eflag(K,J)=3;
end;

if yme(K,J)>0 and yms(K,J)=. then do;
  yms(K,J)=dliym;
  s_day(K,J)=ddliday;
  if j=1 then eflag(K,J)=3;
  else if j>1 then eflag(K,J)=7;
end;

```

/\*\* Note: If either stop month or year is ge -3 then it implies that this unemp spell has ended. Even if



the respondent reports currently receiving that would imply another loop. \*\*/

/\*\* 4. If stop year is known and stop month is unknown. \*\*/  
if yms(K,J)>0 and ye(K,J)>0 and -3 le me(K,J) le -1 then do;

/\*\* 4.a If weeks are known, then count forward from start year. If the number of months falls short of the

stop year, then use January of the end year as the stop date. If the number of months exceeds the stop year, then end the array in the December of the stop year. Constrain stop date by Current Interview Date. If currently receiving then use interview date as the stop date. \*\*/

```
if wks(K,J) ge 0 then do;
  ymsmo(K,J)=yms(K,J)- round (yms(K,J),100);
  if 01 le (ymsmo(K,J)+mos(K,J)) le 12 then yme(K,J)=yms(K,J)+mos(K,J);
  if 13 le (ymsmo(K,J)+mos(K,J)) le 24 then yme(K,J)=yms(K,J)+100+mos(K,J)-12;
  if 25 le (ymsmo(K,J)+mos(K,J)) le 36 then yme(K,J)=yms(K,J)+200+mos(K,J)-24;
  if 37 le (ymsmo(K,J)+mos(K,J)) le 48 then yme(K,J)=yms(K,J)+300+mos(K,J)-36;
  if 49 le (ymsmo(K,J)+mos(K,J)) le 60 then yme(K,J)=yms(K,J)+400+mos(K,J)-48;
  if 61 le (ymsmo(K,J)+mos(K,J)) le 72 then yme(K,J)=yms(K,J)+500+mos(K,J)-60;
  if 73 le (ymsmo(K,J)+mos(K,J)) le 84 then yme(K,J)=yms(K,J)+600+mos(K,J)-72;
```

```
do Z=1 to 12;
  if yme(K,J)-round(yme(K,J),100)=mon(Z) then e_day(K,J)=uiday(Z);
end;
```

```
if yme(K,J)>((ye(K,J)*100)+12) then do;
  yme(K,J)=((ye(K,J)*100)+12);
  e_day(K,J)=31;
end;
```

```
if yme(K,J)<((ye(K,J)*100)+01) then do;
  yme(K,J)=((ye(K,J)*100)+01);
  e_day(K,J)=31;
end;
```

```
if yme(K,J)>iym then do;
  yme(K,J)=iym;
  e_day(K,J)=yint_d;
end;
end;
```

/\*\* 4.b If weeks are unknown, then set December of stop year as the stop month. \*\*/

```
if -3 le wks(K,J) lt 0 then do;
  yme(K,J)=((ye(K,J)*100)+12);
  e_day(K,J)=31;
end;
```

```
if yme(K,J)>iym then do;
  yme(K,J)=iym;
  e_day(K,J)=yint_d;
end;
eflag(K,J)=4;
```

```

end;

/** 5. If stop year is unknown. **/
if -3 le ye(K,J) le -1 then do;

/** 5.a If weeks are known then count forward by the number of weeks from the start date and constrain
stop
date by Current Interview date. **/
if yms(K,J)>0 and wks(K,J) ge 0 then do;
  ymsmo(K,J)=yms(K,J)- round (yms(K,J),100);
  if 01 le (ymsmo(K,J)+mos(K,J)) le 12 then yme(K,J)=yms(K,J)+mos(K,J);
  if 13 lt (ymsmo(K,J)+mos(K,J)) le 24 then yme(K,J)=yms(K,J)+100+mos(K,J)-12;
  if 25 lt (ymsmo(K,J)+mos(K,J)) le 36 then yme(K,J)=yms(K,J)+200+mos(K,J)-24;
  if 37 le (ymsmo(K,J)+mos(K,J)) le 48 then yme(K,J)=yms(K,J)+300+mos(K,J)-36;
  if 49 le (ymsmo(K,J)+mos(K,J)) le 60 then yme(K,J)=yms(K,J)+400+mos(K,J)-48;
  if 61 le (ymsmo(K,J)+mos(K,J)) le 72 then yme(K,J)=yms(K,J)+500+mos(K,J)-60;
  if 73 le (ymsmo(K,J)+mos(K,J)) le 84 then yme(K,J)=yms(K,J)+600+mos(K,J)-72;

do Z=1 to 12;
  if yme(K,J)-round(yme(K,J),100)=mon(Z) then e_day(K,J)=uiday(Z);
end;

if yme(K,J)>iym then do;
  yme(K,J)=iym;
  e_day(K,J)=yint_d;
end;
end;

/** 5.b If weeks are unknown then set stop date as December of start year. **/
if yms(K,J)>0 and -3 le wks(K,J) lt 0 then do;
  yme(K,J)=(round (yms(K,J),100))+12;
  e_day(K,J)=31;
end;

if yme(K,J)>iym then do;
  yme(K,J)=iym;
  e_day(K,J)=yint_d;
end;

eflag(K,J)=5;
end;

/** 6. If the start and stop years are unknown and the weeks are unknown, use current and last
interview date. **/
if -3 le ys(K,J) le -1 and cwks(K,J)=0 and -3 le wks(K,J) lt 0 then do;
  yme(K,J)=iym;
  e_day(K,J)=yint_d;
  yms(K,J)=dliym;
  s_day(K,J)=ddliday;

if j=1 then eflag(K,J)=6;

```

```
    else if j>1 then eflag(K,J)=7;
end;
```

```
/** This portion calculates the number of days receiving if start and end year and months are the same
**/
```

```
    newms(K,J)=(yms(K,J)- round ( yms(K,J),100));
    newme(K,J)=(yme(K,J)- round ( yme(K,J),100));
    if (yms(K,J)=yme(K,J) and yms(K,J) gt 0 and yms(K,J) ne . and newms(K,J) gt 0 and newms
```

```
(K,J) ne .
    and e_day(K,J) ne . and e_day(K,J) ge 0 and s_day(K,J) ne . and s_day(K,J) ge 0) then do;
    SAMEMONTH=1;
```

```
    if (newms(K,J)=1 OR newms(K,J)=3 OR newms(K,J)=5 OR newms(K,J)=7
    OR newms(K,J)=8 OR newms(K,J)=10 OR newms(K,J)=12) then do;
```

```
        if e_day(K,J) ge (32-s_day(K,J)) then do;
            s_day(K,J)=e_day(K,J)-(32-s_day(K,J))+1;
            e_day(K,J)=0;
```

```
        end;
        else if e_day(K,J) lt (32-s_day(K,J)) then do;
            s_day(K,J)=0;
            e_day(K,J)=0;
            HANDEDIT=1;
```

```
        end;
    end;
```

```
    if (newms(K,J)=4 OR newms(K,J)=6 OR newms(K,J)=9 OR newms(K,J)=11) then do;
```

```
        if e_day(K,J) ge (31-s_day(K,J)) then do;
            s_day(K,J)=e_day(K,J)-(31-s_day(K,J))+1;
            e_day(K,J)=0;
```

```
        end;
        else if e_day(K,J) lt (31-s_day(K,J)) then do;
            s_day(K,J)=0;
            e_day(K,J)=0;
            HANDEDIT=1;
```

```
        end;
    end;
```

```
    if newms(K,J)=2 then do;
```

```
        if e_day(K,J) ge (29-s_day(K,J)) then do;
            s_day(K,J)=e_day(K,J)-(29-s_day(K,J))+1;
            e_day(K,J)=0;
```

```
        end;
        else if e_day(K,J) lt (29-s_day(K,J)) then do;
            s_day(K,J)=0;
            e_day(K,J)=0;
            HANDEDIT=1;
```

```
        end;
    end;
end;
```

```
/** This portion of the program uses the category reported by the respondent to create an estimated amount. The estimated amount is the midpoint rounded down. Note that the 12th category lists $1251
```

as the amount. This amount was chosen since the category is unbounded - the number represents one dollar above the lower bound.\*\*/

```

if -3 le inc(K,J) le -1 and einc(K,J)=1 then inc(K,J)=(50/7);
if -3 le inc(K,J) le -1 and einc(K,J)=2 then inc(K,J)=(125/7);
if -3 le inc(K,J) le -1 and einc(K,J)=3 then inc(K,J)=(175/7);
if -3 le inc(K,J) le -1 and einc(K,J)=4 then inc(K,J)=(225/7);
if -3 le inc(K,J) le -1 and einc(K,J)=5 then inc(K,J)=(275/7);
if -3 le inc(K,J) le -1 and einc(K,J)=6 then inc(K,J)=(351/7);
if inc(K,J)>-4 then aflag(K,J)=0;

```

```

csm(K,J)=(round(yms(K,J),100)-198000)*.12+(yms(K,J)-round(yms(K,J),100));
cem(K,J)=(round(yme(K,J),100)-198000)*.12+(yme(K,J)-round(yme(K,J),100));
cdli=(round(dliym,100)-198000)*.12+(dliym-round(dliym,100))-1;

```

/\*\* added to zero out people who back report. \*\*/

```

end; /** [E3] **/

```

```

end; /** [E2] **/

```

/\*\* This portion counts the number of Unemployment Spells in each year \*\*/

```

do K=1 to DIM(csm,1);

```

```

do J=1 to DIM(csm,2);

```

```

if (csm(K,J) gt dlicm) OR (csm(K,J) le dlicm and dli(K,J)=0) then do;

```

```

if 1 le csm(K,J) le 12 then nu80=nu80+1;

```

```

if 13 le csm(K,J) le 24 then nu81=nu81+1;

```

```

if 25 le csm(K,J) le 36 then nu82=nu82+1;

```

```

if 37 le csm(K,J) le 48 then nu83=nu83+1;

```

```

if 49 le csm(K,J) le 60 then nu84=nu84+1;

```

```

if 61 le csm(K,J) le 72 then nu85=nu85+1;

```

```

if 73 le csm(K,J) le 84 then nu86=nu86+1;

```

```

if 85 le csm(K,J) le 96 then nu87=nu87+1;

```

```

if 97 le csm(K,J) le 108 then nu88=nu88+1;

```

```

if 109 le csm(K,J) le 120 then nu89=nu89+1;

```

```

if 121 le csm(K,J) le 132 then nu90=nu90+1;

```

```

if 133 le csm(K,J) le 144 then nu91=nu91+1;

```

```

if 145 le csm(K,J) le 156 then nu92=nu92+1;

```

```

if 157 le csm(K,J) le 168 then nu93=nu93+1;

```

```

if 169 le csm(K,J) le 180 then nu94=nu94+1;

```

```

if 181 le csm(K,J) le 192 then nu95=nu95+1;

```

```

if 193 le csm(K,J) le 204 then nu96=nu96+1;

```

```

if 205 le csm(K,J) le 216 then nu97=nu97+1;

```

```

if 217 le csm(K,J) le 228 then nu98=nu98+1;

```

```

if 229 le csm(K,J) le 240 then nu99=nu99+1;

```

```

if 241 le csm(K,J) le 252 then nu00=nu00+1;

```

```

if 253 le csm(K,J) le 264 then nu01=nu01+1;

```

```

if 265 le csm(K,J) le 269 then nu02=nu02+1;

```

```

end;

```

```

end;

```

```

end;

```

```

do I=1 to DIM(nu);

```

```

CURRnu(I)=nu(I);
end;

/** HANDEDITS - 2 */
/** This person reports receiving UC from Feb 8, 2002 to Feb 4, 2002. So we switched the start and end
dates which makes the number of days this person received UC as 5. */
if pubid=2126 then do; sdayu22=5; edayu22=0; DAYS_PROB=1; end;

/** This person reports receiving UC from May 11,2001 to May 7,2001 . So we switched the start and
end dates which makes the number of days this person received UC as 5.*/
if pubid=2595 then do; sdayu11=5; edayu11=0; DAYS_PROB=1; end;

/** This portion creates the month receiving indicator (UI) and the amount receiving each month
(UIAMT) */

do K=1 to DIM(csm,1);
  do J=1 to DIM(csm,2);
    C=0;
    do L=1 to DIM(ui);
      C=C+1;
      if 0 le L le doicm then do;
        if csm(K,J) le C le cem(K,J) then do;
          if (C=csm(K,J) and inc(K,J) ge 0) then do;
            ui(L)=eflag(K,J);
            if s_day(K,J) ne . then uiamt(L)= round ((inc(K,J)*s_day(K,J)), 1);
          end;

          else if (C=cem(K,J) and inc(K,J) ge 0) then do;
            ui(L)=eflag(K,J);
            if e_day(K,J) ne . then uiamt(L)= round ((inc(K,J)*e_day(K,J)), 1);
          end;

          else if csm(K,J)<C and C<cem(K,J) and inc(K,J) ge 0 then do;
            ui(L)=eflag(K,J);
            if C in (169,171,173,175,176,178,180,181,183,185,187,188,
190,192,193,195,197,199,200,202,204,205,207,209,
211,212,214,216,217,219,221,223,224,226,228,229,
231,233,235,236,238,240,241,243,245,247,248,250,
252,253,255,257,259,260,262,264,265,267,269) then do;
              uiamt(L)= round ((inc(K,J)*31),1);
            end;
            else if C in (172,174,177,179,184,186,189,191,196,198,201,203,
208,210,213,215,220,222,225,227,232,234,237,239,
244,246,249,251,256,258,261,263,268) then do;
              uiamt(L)=round ((inc(K,J)*30),1);
            end;
            else if C in (194,242) then do;
              uiamt(L)=round ((inc(K,J)*29),1);
            end;
            else if C in (170,182,206,218,230,254,266) then do;
              uiamt(L)= round ((inc(K,J)*28),1);
            end;
          end;
        end;
      end;
    end;
  end;
end;

```

```

    end;
end;

    else if inc(K,J)<0 then do;
        ui(L)=eflag(K,J);
        uiamt(L)=inc(K,J);
    end;

    if uiamt(L) le 10000 and (uiodd=0 OR uiodd=.) then do; uiodd=0; end;
    if uiamt(L)>10000 then do; uiodd=uiamt(L); end;
    if ui(L)>0 and uiamt(L)=. then do; uick=uick+1; end;
    if (uiamt(L)>0 OR (-3 le uiamt(L) lt 0)) and ui(L)=0 then do; uickm=uickm+1; end;
    if ui(L)=-3 and uiamt(L) ne -3 then do; uickmm=uickmm+1; end;
    if uiamt(L)=0 then dummyui=1;
end;
end;

end;
end;
end;

```

```

/** This portion deletes spells that started before DLI **/

```

```

do K=1 to DIM(csm,1);
do J=1 to DIM(csm,2);
C=0;
do L=1 to DIM(ui);
C=C+1;
if csm(K,J) le cdli and p010 eq 1 and csm(K,J) ne . then do;
if csm(K,J) le C le cdli then do;
ui(L)=0;
uiamt(L)=.;
end;
end;
end;
end;
end;
end;

```

```

end;
/** [E1] **/

```

```

/** This portion creates the months receiving in each year variables (UMXX) and the amount receiving
each year variables (UAMTXX). **/

```

```

do I=1 to DIM(um);
if ui_ever>-4 OR p1210>-4 OR p1210=-5 then do;
um(I)=0;
uamt(I)=0;
ua(I)=0;
un(I)=0;
end;
end;
end;

```

```
/** HANDEDITS - 3 **/
```

```
/** These persons say that they are currently receiving but give a stop date that is different from the current interview date so we give them a -3 for all the months between the DLI and current interview date.**/
```

```
if pubid=5547 then do;  
  STP_DT_PROB=1;  
  do L=1 to 269;  
    if cdli le L le doicm then do;  
      ui(L)=-3;  
      uiamt(L)=-3;  
    end;  
  end;  
  UM01=-3;  
  UM02=-3;  
end;
```

```
if pubid=6921 then do;  
  STP_DT_PROB=1;  
  do L=1 to 269;  
    if cdli le L le doicm then do;  
      ui(L)=-3;  
      uiamt(L)=-3;  
    end;  
  end;  
  UM01=-3;  
end;
```

```
if ui_ever>-4 OR p1210>-4 OR p1210=-5 then do;
```

```
/** [B4] **/
```

```
/** calculate total no. months to divide the problematic amts over.
```

```
**/
```

```
do L=1 to DIM(ui);  
  if ui(L)>0 and uiamt(L)>10000 and uiodd>0 then do;  
    uimnths=uimnths+1;  
  end;  
end;
```

```
/** divide the amount by the number of months.
```

```
**/
```

```
do L=1 to DIM(ui);  
  if uiamt(L)>10000 and uimnths>0 and ui(L)>0 then do;  
    uiamt(L)=round(uiamt(L)/uimnths, 1);  
  end;  
end;
```

```
do L=1 to DIM(ui);
```

```
/** [B5] **/
```

```
  if 1 le L le 12 then do;  
    if um80 ge 0 and ui(L)>0 then do;  
      um80=um80+1;  
    end;  
    if ui(L)>1 then um80=-3;  
    if -3 le uiamt(L) le -1 then do;  
      un80=un80+1;
```

```
    uamt80=uiamt(L);
end;
if uiamt(L) ge 0 then do;
    ua80=ua80+1;
    if ua80=1 and un80=0 then uamt80=uiamt(L);
    if ua80>1 and un80=0 then uamt80=uamt80+uiamt(L);
    if um80=-3 then uamt80=-3;
end;
end;
```

```
if 13 le L le 24 then do;
    if um81 ge 0 and ui(L)>0 then do;
        um81=um81+1;
    end;
    if ui(L)>1 then um81=-3;
    if -3 le uiamt(L) le -1 then do;
        un81=un81+1;
        uamt81=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua81=ua81+1;
        if ua81=1 and un81=0 then uamt81=uiamt(L);
        if ua81>1 and un81=0 then uamt81=uamt81+uiamt(L);
        if um81=-3 then uamt81=-3;
    end;
end;
```

```
if 25 le L le 36 then do;
    if um82 ge 0 and ui(L)>0 then do;
        um82=um82+1;
    end;
    if ui(L)>1 then um82=-3;
    if -3 le uiamt(L) le -1 then do;
        un82=un82+1;
        uamt82=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua82=ua82+1;
        if ua82=1 and un82=0 then uamt82=uiamt(L);
        if ua82>1 and un82=0 then uamt82=uamt82+uiamt(L);
        if um82=-3 then uamt82=-3;
    end;
end;
```

```
if 37 le L le 48 then do;
    if um83 ge 0 and ui(L)>0 then do;
        um83=um83+1;
    end;
    if ui(L)>1 then um83=-3;
    if -3 le uiamt(L) le -1 then do;
        un83=un83+1;
    end;
end;
```



```
    uamt83=uiamt(L);
end;
if uiamt(L) ge 0 then do;
    ua83=ua83+1;
    if ua83=1 and un83=0 then uamt83=uiamt(L);
    if ua83>1 and un83=0 then uamt83=uamt83+uiamt(L);
    if um83=-3 then uamt83=-3;
end;
end;
```

```
if 49 le L le 60 then do;
    if um84 ge 0 and ui(L)>0 then do;
        um84=um84+1;
    end;
    if ui(L)>1 then um84=-3;
    if -3 le uiamt(L) le -1 then do;
        un84=un84+1;
        uamt84=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua84=ua84+1;
        if ua84=1 and un84=0 then uamt84=uiamt(L);
        if ua84>1 and un84=0 then uamt84=uamt84+uiamt(L);
        if um84=-3 then uamt84=-3;
    end;
end;
```

```
if 61 le L le 72 then do;
    if um85 ge 0 and ui(L)>0 then do;
        um85=um85+1;
    end;
    if ui(L)>1 then um85=-3;
    if -3 le uiamt(L) le -1 then do;
        un85=un85+1;
        uamt85=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua85=ua85+1;
        if ua85=1 and un85=0 then uamt85=uiamt(L);
        if ua85>1 and un85=0 then uamt85=uamt85+uiamt(L);
        if um85=-3 then uamt85=-3;
    end;
end;
```

```
if 73 le L le 84 then do;
    if um86 ge 0 and ui(L)>0 then do;
        um86=um86+1;
    end;
    if ui(L)>1 then um86=-3;
    if -3 le uiamt(L) le -1 then do;
        un86=un86+1;
    end;
end;
```

```
    uamt86=uiamt(L);
end;
if uiamt(L) ge 0 then do;
    ua86=ua86+1;
    if ua86=1 and un86=0 then uamt86=uiamt(L);
    if ua86>1 and un86=0 then uamt86=uamt86+uiamt(L);
    if um86=-3 then uamt86=-3;
end;
end;
```

```
if 85 le L le 96 then do;
    if um87 ge 0 and ui(L)>0 then do;
        um87=um87+1;
    end;
    if ui(L)>1 then um87=-3;
    if -3 le uiamt(L) le -1 then do;
        un87=un87+1;
        uamt87=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua87=ua87+1;
        if ua87=1 and un87=0 then uamt87=uiamt(L);
        if ua87>1 and un87=0 then uamt87=uamt87+uiamt(L);
        if um87=-3 then uamt87=-3;
    end;
end;
```

```
if 97 le L le 108 then do;
    if um88 ge 0 and ui(L)>0 then do;
        um88=um88+1;
    end;
    if ui(L)>1 then um88=-3;
    if -3 le uiamt(L) le -1 then do;
        un88=un88+1;
        uamt88=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua88=ua88+1;
        if ua88=1 and un88=0 then uamt88=uiamt(L);
        if ua88>1 and un88=0 then uamt88=uamt88+uiamt(L);
        if um88=-3 then uamt88=-3;
    end;
end;
```

```
if 109 le L le 120 then do;
    if um89 ge 0 and ui(L)>0 then do;
        um89=um89+1;
    end;
    if ui(L)>1 then um89=-3;
    if -3 le uiamt(L) le -1 then do;
        un89=un89+1;
    end;
end;
```

```
    uamt89=uiamt(L);
end;
if uiamt(L) ge 0 then do;
    ua89=ua89+1;
    if ua89=1 and un89=0 then uamt89=uiamt(L);
    if ua89>1 and un89=0 then uamt89=uamt89+uiamt(L);
    if um89=-3 then uamt89=-3;
end;
end;
```

```
if 121 le L le 132 then do;
    if um90 ge 0 and ui(L)>0 then do;
        um90=um90+1;
    end;
    if ui(L)>1 then um90=-3;
    if -3 le uiamt(L) le -1 then do;
        un90=un90+1;
        uamt90=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua90=ua90+1;
        if ua90=1 and un90=0 then uamt90=uiamt(L);
        if ua90>1 and un90=0 then uamt90=uamt90+uiamt(L);
        if um90=-3 then uamt90=-3;
    end;
end;
```

```
if 133 le L le 144 then do;
    if um91 ge 0 and ui(L)>0 then do;
        um91=um91+1;
    end;
    if ui(L)>1 then um91=-3;
    if -3 le uiamt(L) le -1 then do;
        un91=un91+1;
        uamt91=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua91=ua91+1;
        if ua91=1 and un91=0 then uamt91=uiamt(L);
        if ua91>1 and un91=0 then uamt91=uamt91+uiamt(L);
        if um91=-3 then uamt91=-3;
    end;
end;
```

```
if 145 le L le 156 then do;
    if um92 ge 0 and ui(L)>0 then do;
        um92=um92+1;
    end;
    if ui(L)>1 then um92=-3;
    if -3 le uiamt(L) le -1 then do;
        un92=un92+1;
```

```
    uamt92=uiamt(L);
end;
if uiamt(L) ge 0 then do;
    ua92=ua92+1;
    if ua92=1 and un92=0 then uamt92=uiamt(L);
    if ua92>1 and un92=0 then uamt92=uamt92+uiamt(L);
    if um92=-3 then uamt92=-3;
end;
end;
```

```
if 157 le L le 168 then do;
    if um93 ge 0 and ui(L)>0 then do;
        um93=um93+1;
    end;
    if ui(L)>1 then um93=-3;
    if -3 le uiamt(L) le -1 then do;
        un93=un93+1;
        uamt93=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua93=ua93+1;
        if ua93=1 and un93=0 then uamt93=uiamt(L);
        if ua93>1 and un93=0 then uamt93=uamt93+uiamt(L);
        if um93=-3 then uamt93=-3;
    end;
end;
```

```
if 169 le L le 180 then do;
    if um94 ge 0 and ui(L)>0 then do;
        um94=um94+1;
    end;
    if ui(L)>1 then um94=-3;
    if -3 le uiamt(L) le -1 then do;
        un94=un94+1;
        uamt94=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua94=ua94+1;
        if ua94=1 and un94=0 then uamt94=uiamt(L);
        if ua94>1 and un94=0 then uamt94=uamt94+uiamt(L);
        if um94=-3 then uamt94=-3;
    end;
end;
```

```
if 181 le L le 192 then do;
    if um95 ge 0 and ui(L)>0 then do;
        um95=um95+1;
    end;
    if ui(L)>1 then um95=-3;
    if -3 le uiamt(L) le -1 then do;
        un95=un95+1;
    end;
end;
```

```
    uamt95=uiamt(L);
end;
if uiamt(L) ge 0 then do;
    ua95=ua99+1;
    if ua95=1 and un95=0 then uamt95=uiamt(L);
    if ua95>1 and un95=0 then uamt95=uamt95+uiamt(L);
    if um95=-3 then uamt95=-3;
end;
end;
```

```
if 193 le L le 204 then do;
    if um96 ge 0 and ui(L)>0 then do;
        um96=um96+1;
    end;
    if ui(L)>1 then um96=-3;
    if -3 le uiamt(L) le -1 then do;
        un96=un96+1;
        uamt96=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua96=ua96+1;
        if ua96=1 and un96=0 then uamt96=uiamt(L);
        if ua96>1 and un96=0 then uamt96=uamt96+uiamt(L);
        if um96=-3 then uamt96=-3;
    end;
end;
```

```
if 205 le L le 216 then do;
    if um97 ge 0 and ui(L)>0 then do;
        um97=um97+1;
    end;
    if ui(L)>1 then um97=-3;
    if -3 le uiamt(L) le -1 then do;
        un97=un97+1;
        uamt97=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua97=ua97+1;
        if ua97=1 and un97=0 then uamt97=uiamt(L);
        if ua97>1 and un97=0 then uamt97=uamt97+uiamt(L);
        if um97=-3 then uamt97=-3;
    end;
end;
```

```
if 217 le L le 228 then do;
    if um98 ge 0 and ui(L)>0 then do;
        um98=um98+1;
    end;
    if ui(L)>1 then um98=-3;
    if -3 le uiamt(L) le -1 then do;
        un98=un98+1;
    end;
end;
```

```
    uamt98=uiamt(L);
end;
if uiamt(L) ge 0 then do;
    ua98=ua98+1;
    if ua98=1 and un98=0 then uamt98=uiamt(L);
    if ua98>1 and un98=0 then uamt98=uamt98+uiamt(L);
    if um98=-3 then uamt98=-3;
end;
end;
```

```
if 229 le L le 240 then do;
    if um99 ge 0 and ui(L)>0 then do;
        um99=um99+1;
    end;
    if ui(L)>1 then um99=-3;
    if -3 le uiamt(L) le -1 then do;
        un99=un99+1;
        uamt99=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua99=ua99+1;
        if ua99=1 and un99=0 then uamt99=uiamt(L);
        if ua99>1 and un99=0 then uamt99=uamt99+uiamt(L);
        if um99=-3 then uamt99=-3;
    end;
end;
```

```
if 241 le L le 252 then do;
    if um00 ge 0 and ui(L)>0 then do;
        um00=um00+1;
    end;
    if ui(L)>1 then um00=-3;
    if -3 le uiamt(L) le -1 then do;
        un00=un00+1;
        uamt00=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua00=ua00+1;
        if ua00=1 and un00=0 then uamt00=uiamt(L);
        if ua00>1 and un00=0 then uamt00=uamt00+uiamt(L);
        if um00=-3 then uamt00=-3;
    end;
end;
```

```
if 253 le L le 264 then do;
    if um01 ge 0 and ui(L)>0 then do;
        um01=um01+1;
    end;
    if ui(L)>1 then um01=-3;
    if -3 le uiamt(L) le -1 then do;
        un01=un01+1;
    end;
end;
```

```

    uamt01=uiamt(L);
end;
if uiamt(L) ge 0 then do;
    ua01=ua01+1;
    if ua01=1 and un01=0 then uamt01=uiamt(L);
    if ua01>1 and un01=0 then uamt01=uamt01+uiamt(L);
    if um01=-3 then uamt01=-3;
end;
end;

```

```

if 265 le L le 276 then do;
    if um02 ge 0 and ui(L)>0 then do;
        um02=um02+1;
    end;
    if ui(L)>1 then um02=-3;
    if -3 le uiamt(L) le -1 then do;
        un02=un02+1;
        uamt02=uiamt(L);
    end;
    if uiamt(L) ge 0 then do;
        ua02=ua02+1;
        if ua02=1 and un02=0 then uamt02=uiamt(L);
        if ua02>1 and un02=0 then uamt02=uamt02+uiamt(L);
        if um02=-3 then uamt02=-3;
    end;
end;
end;

```

```

end;
end;

```

```

/** [E5] */
/** [E4] */

```

```

/** Part 3. Combining information from earlier rounds
/** Calculating total months ever received.

```

```

**/

```

```

**/

```

```

do L=1 to DIM(ui);
    if ttlu ge 0 and ui(L)>0 then do;
        ttlu=ttl+1;
    end;
    if ui(L)>1 then ttlu=-3;
    if -3 le ui(L) lt 0 then ttlu=-3;
end;

```

```

if ttlu ge 0 and ui_ever ge 0 then do;
    ttlu=ttl+ui_ever;
end;
if ttlu<0 OR ui_ever<0 then do;
    ttlu=-3;
end;

```

```

/** Calculating the unemployment spells.
do I=1 to DIM(nu);
    if ui_spells(I)=-4 then ui_spells(I)=0;

```

```

**/

```

```

if nu(I) ge 0 and ui_spells(I) ge 0 then do;
  nu(I)=nu(I)+ui_spells(I);
end;
if (ui_spells(I) ge -3 and ui_spells(I) le -1) then do;
  nu(I)=-3;
end;
if nu(I)=0 and oldui_spells(I)=-4 then do;
  nu(I)=-4;
end;
end;

```

```

/** Calculating the total number of months and total amount receiving in each year.          **/

```

```

do I=1 to DIM(uitotal);
  if uitotal(I)=-4 then uitotal(I)=0;
  if uiatotal(I)=-4 then uiatotal(I)=0;

  if um(I) ge 0 and uitotal(I) ge 0 then do;
    uitotal(I)=um(I)+uitotal(I);
  end;
  if um(I)<0 OR uitotal(I)<0 then do;
    uitotal(I)=-3;
  end;
  if uamt(I) ge 0 and uiatotal(I) ge 0 then do;
    uiatotal(I)=uamt(I)+uiatotal(I);
  end;
  if uamt(I)<0 OR uiatotal(I)<0 OR uitotal(I)<0 then do;
    uiatotal(I)=-3;
  end;
  if uamt(I)=0 and olduiatotal(I)=-4 and uitotal(I)=0 then do;
    uiatotal(I)=-4;
  end;
end;

```

```

/** Giving Valid skips (-4) to those not eligible for program participation at DLI and -5 to
noninterviews.**/

```

```

do I=1 to DIM(uitotal);
  if p1210=-4 then do;
    uitotal(I)=-4;
    nu(I)=-4;
    uiatotal(I)=-4;
    ttlu=-4;
  end;
  if p1210=-5 then do;
    uitotal(I)=-5;
    nu(I)=-5;
    uiatotal(I)=-5;
    ttlu=-5;
  end;
end;

```

```

/** To change all invalid skips to -3.          **/

```



```

array vars uitotal80-uitotal99 uitotal00 uitotal01 uitotal02
      uiatotal80-uiatotal99 uiatotal00 uiatotal01 uiatotal02
      nu80-nu99 nu00 nu01 nu02 tflu;
do over vars;
  if vars=-2 then vars=-3;
  if vars=-1 then vars=-3;
end;

```

---

**\*\*\*\*\* SECTION 2: WORKERS COMPENSATION \*\*\*\*\***

```

/** All the arrays used in this program.                               **/
array intd      (4)  int00_d int99_d int98_d int97_d;
array intm      (4)  int00_m int99_m int98_m int97_m;
array inty      (4)  int00_y int99_y int98_y int97_y;

array wcatotalpr (4,23) wca4total80-wca4total99 wca4total00 wca4total01 wca4total02
      wca3total80-wca3total99 wca3total00 wca3total01 wca3total02
      wca2total80-wca2total99 wca2total00 wca2total01 wca2total02
      wca1total80-wca1total99 wca1total00 wca1total01 wca1total02;

array wcatotal  (23) wcatotal80-wcatotal99 wcatotal00 wcatotal01 wcatotal02;
array oldwcatotal (23) oldwcatotal80-oldwcatotal99 oldwcatotal00 oldwcatotal01 oldwcatotal02;
array wcm        (23) wcm80-wcm99 wcm00 wcm01 wcm02;
array camt       (23) camt80-camt99 camt00 camt01 camt02;
array wca        (23) wca80-wca99 wca00 wca01 wca02;
array wcn        (23) wcn80-wcn99 wcn00 wcn01 wcn02;
array wc         (269) wc001-wc269;
array wcamt      (269) wcamt001-wcamt269;
array yms        (2)  ymsc11 ymsc21;
array csm        (2)  csmc11 csmc21;
array yme        (2)  ymec11 ymec21;
array cem        (2)  cemc11 cemc21;
array s_day      (2)  sdayc11 sdayc21;
array e_day      (2)  edayc11 edayc21;
array eflag      (2)  eflagc11 eflagc21;
array aflag      (2)  aflagc11 aflagc21;
array mos        (2)  mosc11 mosc21;
array agefl      (2)  ageflc11 ageflc21;
array ymsmo      (2)  ymsmo11 ymsmo21;
array ymemo      (2)  ymemo11 ymemo21;
array newms      (2)  newms11 newms21;
array newme      (2)  newme11 newme21;

/** start YEAR, MONTH, ACTUAL DAY and ESTIMATED DAY
information.                                                         **/
array ys        (2)  ps102001Y p102001Y;
array ms        (2)  ps102001M p102001M;
array ds        (2)  ps104001 p104001;
array es        (2)  ps105001 p105001;

```

```

/** end YEAR, MONTH, ACTUAL DAY and ESTIMATED DAY
information.                               **/
array ye      (2)  ps109001Y p109001Y;
array me      (2)  ps109001M p109001M;
array de      (2)  ps111001 p111001;
array ee      (2)  ps112001 p112001;

/** ESTIMATED WEEKS - if unable to answer start or stop date questions.           **/
array wks     (2)  ps114001 p114001;

/** CURRENTLY receiving information:                                             **/
array cur     (2)  ps106001 p106001;
array cwks    (2)  ps9900 p9700;

/** Receiving at DLI:                                                            **/
array dli     (2)  ps9800 p9800;

/** AMOUNT RECEIVED - actual & estimated.                                       **/
array inc     (2)  ps121001 p121001;
array einc    (2)  ps122001 p122001;

/** Part 1. Creating certain basic variables                                     **/
/** change weekly wc figures to daily amount.                                   **/
do J=1 to 2;
  if inc(J) gt 0 then inc(J)=inc(J)/7;
end;

/** change previous round variables to account for noninterviews -
  If r4 interview, use r4 created var. If noninterview in r4, then use r3 variable.
  If noninterview in r3, then use r2 variable. If noninterview in r2, then use r1 variable
  If variable not in r1, r2, r3 or r4 then set to 0.                             **/

do Q=1 to 4;
  if inty(Q) ne -5 then do;
    do I=1 to 23;
      wcatotal(I)=wcatotalpr(Q,I);
    end;
    Q=4;
  end;
end;

do I=1 to 23;
  oldwcatotal(I)=wcatotal(I);
  if wcatotal(I)=. then do;
    oldwcatotal(I)=-4;
    wcatotal(I)=0;
  end;
end;
end;

```

```

/** Initialize the following counters to zero.                                **/
/** Counter for WCAMT ge 0 (receiving positive amounts) but WC=0 (not receiving).
**/
/** Counter for WC gt 0 (receiving) but WCAMT is missing (amount receiving info not
available)                                **/
/** Total number of months ever received.                                    **/
/** Month counter for correction of problematic amounts reported.          **/

wck=0;
wckm=0;
wckmm=0;
wcmnths=0;
SAMEMONTH=0;

/** Creating variables relating to date of birth, age, age 14 etc.          **/
cm14=((bdate_y+14)-1980)*12+bdate_m;
ym14=(bdate_y*100+bdate_m)+1400;
cmb=((bdate_y)-1980)*12+bdate_m;
doicm=(yint_y-1980)*12+yint_m;
iym=yint_y*100+yint_m;
aaicm=(yint_y-bdate_y)*12+yint_m-bdate_m;
array mon (12) mon1-mon12;
array wcdays (12) wcdays1-wcdays12;

do Z=1 to 12;
  mon(Z)=Z;
end;

do Z=1 to 12;
  if mon(Z)=1 OR mon(Z)=3 OR mon(Z)=5 OR mon(Z)=7 OR mon(Z)=8 OR mon(Z)=10 OR mon(Z)
=12
  then do;
    wcdays(Z)=31;
  end;
  if mon(Z)=2 then do;
    wcdays(Z)=28;
  end;
  if mon(Z)=4 OR mon(Z)=6 OR mon(Z)=9 OR mon(Z)=11 then do;
    wcdays(Z)=30;
  end;
end;

/** Creating variables pertaining to the date of last interview.          **/
if p1210=1 then do;
  do Q=1 to 4;
    if inty[Q] ne -5 then do;
      if intm[Q] lt 12 then dliym=(inty[Q]*100)+(intm[Q]+1);
      if intm[Q]=12 then dliym=((inty[Q]+1)*100)+1;
      dlicm=((inty[Q]-1980)*12)+(intm[Q]+1);
      dli_d=intd[Q];
      dli_m=dliym-round(dliym,100);
    end;
  end;
end;

```

```
    dli_y=round(dliym,100)/100;
    Q=4;
end;
end;
end;
```

```
if p1210 ne 1 then do;
    dliym=ym14;
    dli_d=bdate_d;
    dli_m=bdate_m;
    dli_y=bdate_y;
    dlicm=cm14;
end;
```

```
do Z=1 to 12;
    if yint_m=mon(Z) then intday=wcdays(Z)+1-yint_d;
    if dli_m=mon(Z) then ddliday=wcdays(Z)+1-dli_d;
end;
```

```
if dli_m=1 OR dli_m=8 then do;
    dliday=(31-dli_d+31);
    dliday1=31-dli_d;
    dliday2=31;
end;
```

```
if dli_m=3 then do;
    dliday=(28-dli_d+31);
    dliday1=28-dli_d;
    dliday2=31;
end;
```

```
if dli_m=5 OR dli_m=7 OR dli_m=10 OR dli_m=12 then do;
    dliday=(30-dli_d+31);
    dliday1=30-dli_d;
    dliday2=31;
end;
```

```
if dli_m=4 OR dli_m=6 OR dli_m=9 OR dli_m=11 then do;
    dliday=(31-dli_d+30);
    dliday1=31-dli_d;
    dliday2=30;
end;
```

```
if dli_m=2 then do;
    dliday=(31-dli_d+28);
    dliday1=31-dli_d;
    dliday2=28;
end;
```

```
/** Part 2. Creating variables using Round 5 information
if p1210>-4 then do;
```

```
/**
/** [B1] **/
```

```
/** Combining currently receiving information for the NEVER PRIOR sections.
if ps9867>-3 then p9700=ps9867;
```

```
**/
```

```

/** Initializing all wc from the month of birth to the date of current interview to zero.      **/
do L=1 to 269;
  if cmb le L le doicm then do;
    wc(L)=0;
  end;
end;

```

/\*\* This portion of the SAS program defines the start date and end dates. If the respondent reports still receiving, the interview date is used as the temporary end date for the last loop reported. In the next survey round, the respondent will be asked if he or she is still receiving and, if not, a permanent end date equivalent to the interview date of the previous round will be assigned. Users will be able to tell which method was used by looking at the following participation flag variable created during the program.

The categories are the following:

- 1=respondent reported participation dates
- 2=start month imputed
- 3=start month and year imputed
- 4=stop month imputed
- 5=stop month and year imputed
- 6=start and stop dates imputed
- 7=error in data due to round 2 programming error \*\*/

```

do J=1 to 2;                                     /** [B2] **/

```

```

/** 1. If start and stop dates are all valid                                             **/

```

```

  if ys(J)>0 and ms(J)>0 then do;
    yms(J)=(ys(J)*100)+ms(J);

```

```

  /** set start day, if given.                                                         **/

```

```

  if ds(J)>0 OR es(J)>0 then do;
    if ms(J)=1 OR ms(J)=3 OR ms(J)=5 OR ms(J)=7 OR ms(J)=8 OR ms(J)=10 OR ms(J)=12
    then do;

```

```

      if ds(J)>0 and ds(J) ne . then s_day(J)=(32-ds(J));
      if es(J)=1 then s_day(J)=31;
      if es(J)=2 then s_day(J)=21;
      if es(J)=3 then s_day(J)=11;

```

```

    end;
    if ms(J)=4 OR ms(J)=6 OR ms(J)=9 OR ms(J)=11 then do;

```

```

      if ds(J)>0 and ds(J) ne . then s_day(J)=(31-ds(J));
      if es(J)=1 then s_day(J)=30;
      if es(J)=2 then s_day(J)=20;
      if es(J)=3 then s_day(J)=10;

```

```

    end;
    if ms(J)=2 then do;
      if ds(J)>0 and ds(J) ne . then s_day(J)=(29-ds(J));
      if es(J)=1 then s_day(J)=28;
      if es(J)=2 then s_day(J)=18;
      if es(J)=3 then s_day(J)=8;

```

```

    end;
  end;
end;

```

```

/** set start date to DLI if receiving at DLI.                                **/
if dli(J)=1 then do;
  yms(J)=dliym;
  s_day(J)=dliday;
  if ps9900=1 then do;
    yme(J)=iym;
    e_day(J)=yint_d;
    NOGAPS=1;
  end;
end;

/** set start date to DLI if start date is before DLI.                       **/
if yms(J) ne . and yms(J)<dliym then do;
  yms(J)=dliym;
  s_day(J)=ddliday;
end;

/** set end date to current int date if receiving now.                       **/
if cur(J)=1 then do;
  yme(J)=iym;
  e_day(J)=yint_d;
end;

/** set end date if both end month and year are valid.                      **/
if ye(J)>0 and me(J)>0 then do;
  yme(J)=(ye(J)*100)+me(J);

  /** set end day, if given.                                                **/
  if de(J)>0 and de(J) ne . then do;
    e_day(J)=de(J);
  end;

  /** end day unknown, estimate given.                                     **/
  if de(J)<0 and ee(J)>0 and ee(J) ne . then do;
    if ee(J)=1 then e_day(J)=9;
    if ee(J)=2 then e_day(J)=19;
    if ee(J)=3 then e_day(J)=28;
  end;
end;

/** if end month greater than int month, constrain by int month.          **/
if yme(J)>iym then do;
  yme(J)=iym;
  e_day(J)=yint_d;
end;
if yms(J)>0 and yme(J) ge dliym then eflag(J)=1;

/** determine months from estimated weeks variable.                        **/
if wks(J) ge 0 and (ys(J)<0 OR ms(J)<0 OR ye(J)<0 OR me(J)<0) then do;
  mos(J)=floor(wks(J)/4.3);

```

end;

agefl(J)=0;

if ys(J)>0 and ys(J)<1990 then agefl(J)=1;

/\*\* 2. If start year is known and month is unknown

Note: if wks ge -3 then it means that stop date is not known.

\*/

if ys(J)>0 and -3 le ms(J) le -1 then do;

/\*\* 2.a If weeks are known and currently receiving, then count backwards by the number of weeks from the

current interview date. If the number of weeks falls short of the start year, the start month is

December of that year. If the number of weeks is past the start year, then the start month is January

of that year. \*/

if (wks(J) ge 0 and cwks(J)=1) then do;

yms(J)=iym;

e\_day(J)=yint\_d;

if mos(J) le yint\_m then yms(J)=yms(J)- mos(J) ;

if (yint\_m+ 0) le mos(J) le (yint\_m+ 12) then yms(J)=yms(J)- 100-(mos(J)- 12);

if (yint\_m+12) le mos(J) le (yint\_m+ 24) then yms(J)=yms(J)- 200-(mos(J)- 24);

if (yint\_m+24) le mos(J) le (yint\_m+ 36) then yms(J)=yms(J)- 300-(mos(J)- 36);

if (yint\_m+36) le mos(J) le (yint\_m+ 48) then yms(J)=yms(J)- 400-(mos(J)- 48);

if (yint\_m+48) le mos(J) le (yint\_m+ 60) then yms(J)=yms(J)- 500-(mos(J)- 60);

if (yint\_m+60) le mos(J) le (yint\_m+ 72) then yms(J)=yms(J)- 600-(mos(J)- 72);

if (yint\_m+72) le mos(J) le (yint\_m+ 84) then yms(J)=yms(J)- 700-(mos(J)- 84);

if (yint\_m+84) le mos(J) le (yint\_m+ 96) then yms(J)=yms(J)- 800-(mos(J)- 96);

if (yint\_m+96) le mos(J) le (yint\_m+108) then yms(J)=yms(J)- 900-(mos(J)-108);

if (yint\_m+108) le mos(J) le (yint\_m+120) then yms(J)=yms(J)-1000-(mos(J)-120);

do Z=1 to 12;

if (yms(J)-(round(yms(J), 100)))=mon(Z) then s\_day(J)=wcdays(Z);

end;

if yms(J)<((ys(J)\*100)+01) then do;

yms(J)=((ys(J)\*100)+01);

s\_day(J)=31;

end;

if yms(J)<dliym then do;

yms(J)=dliym;

s\_day(J)=ddliday;

end;

if yms(J)>((ys(J)\*100)+12) then do;

yms(J)=((ys(J)\*100)+12);

s\_day(J)=31;

end;

end;

/\*\* 2.b weeks missing and currently receiving - set end date to current interview date and start date

to December of start year and constrain start date by Current Interview Date and DLI. \*\*/  
else if (-3 le wks(J) lt 0 and cwks(J)=1) then do;

```
yms(J)=iym;  
e_day(J)=yint_d;  
yms(J)=((ys(J)*100)+12);  
s_day(J)=31;
```

```
if yms(J)>iym then do;  
  yms(J)=iym;  
  s_day(J)=intday;  
end;
```

```
if yms(J)<dliym then do;  
  yms(J)=dliym;  
  s_day(J)=ddliday;  
end;  
end;
```

/\*\* 2.c weeks missing and not currently receiving - set start date to January of start year and end date to December of start year. Constrain start date by DLI and end date by Current Interview Date.  
\*\*/

```
else if (-3 le wks(J) lt 0 and cwks(J)=0) then do;  
  yms(J)=((ys(J)*100)+01);  
  s_day(J)=31;  
  yme(J)=((ys(J)*100)+12);  
  e_day(J)=31;
```

```
if yms(J)<dliym then do;  
  yms(J)=dliym;  
  s_day(J)=ddliday;  
end;
```

```
if yme(J)>iym then do;  
  yme(J)=iym;  
  e_day(J)=yint_d;  
end;  
end;
```

/\*\* 2.d If weeks are known and not currently receiving, then count forward by the number of weeks from January of the start year. Constrain start date by DLI. If the count exceeds the interview date then stop counting at the interview date. \*\*/

```
else if (wks(J) ge 0 and cwks(J)=0) then do;  
  yms(J)=((ys(J)*100)+01);  
  s_day(J)=31;  
  if yms(J)<dliym then do;  
    yms(J)=dliym;  
    s_day(J)=ddliday;  
  end;  
end;
```



```

ymsmo(J)=(yms(J)-round(yms(J),100))+mos(J);
if 0 le ymsmo(J) le 12 then yme(J)=yms(J)+ 0+mos(J)- 0;
if 13 le ymsmo(J) le 24 then yme(J)=yms(J)+100+mos(J)- 12;
if 25 le ymsmo(J) le 36 then yme(J)=yms(J)+200+mos(J)- 24;
if 37 le ymsmo(J) le 48 then yme(J)=yms(J)+300+mos(J)- 36;
if 49 le ymsmo(J) le 60 then yme(J)=yms(J)+400+mos(J)- 48;
if 61 le ymsmo(J) le 72 then yme(J)=yms(J)+500+mos(J)- 60;
if 73 le ymsmo(J) le 84 then yme(J)=yms(J)+600+mos(J)- 72;
if 85 le ymsmo(J) le 96 then yme(J)=yms(J)+700+mos(J)- 84;
if 97 le ymsmo(J) le 108 then yme(J)=yms(J)+800+mos(J)- 96;
if 109 le ymsmo(J) le 120 then yme(J)=yms(J)+900+mos(J)-108;

```

```

do Z=1 to 12;
  if (yme(J)-(round(yme(J), 100)))=mon(Z) then e_day(J)=wcdays(Z);
end;

```

```

if yme(J)>iym then do;
  yme(J)=iym;
  e_day(J)=yint_d;
end;
end;

```

```

if yme(J)>0 and CWKS(J)=0 then do;
  yms(J)=((ys(J)*100)+01);
  s_day(J)=31;
end;

```

```

if yms(J)<dliym then do;
  yms(J)=dliym;
  s_day(J)=ddliday;
end;

```

```

eflag(J)=2;
end;

```

```

/** 3. If start year is unknown. **/
if -3 le ys(J) le -1 then do;

```

```

/** 3.a If currently receiving and weeks are known then count back from interview date receiving.
**/

```

```

if (wks(J) ge 0 and cwks(J)=1) then do;
  if 0 le mos(J) le (yint_m+ 0) then yms(J)=iym- 0-(mos(J)- 0);
  if (yint_m+ 0) le mos(J) le (yint_m+ 12) then yms(J)=iym- 100-(mos(J)- 12);
  if (yint_m+ 12) le mos(J) le (yint_m+ 24) then yms(J)=iym- 200-(mos(J)- 24);
  if (yint_m+ 24) le mos(J) le (yint_m+ 36) then yms(J)=iym- 300-(mos(J)- 36);
  if (yint_m+ 36) le mos(J) le (yint_m+ 48) then yms(J)=iym- 400-(mos(J)- 48);
  if (yint_m+ 48) le mos(J) le (yint_m+ 60) then yms(J)=iym- 500-(mos(J)- 60);
  if (yint_m+ 60) le mos(J) le (yint_m+ 72) then yms(J)=iym- 600-(mos(J)- 72);
  if (yint_m+ 72) le mos(J) le (yint_m+ 84) then yms(J)=iym- 700-(mos(J)- 84);
  if (yint_m+ 84) le mos(J) le (yint_m+ 96) then yms(J)=iym- 800-(mos(J)- 96);

```

```
if (yint_m+ 96) le mos(J) le (yint_m+108) then yms(J)=iym- 900-(mos(J)-108);
if (yint_m+108) le mos(J) le (yint_m+120) then yms(J)=iym-1000-(mos(J)-120);
```

```
do Z=1 to 12;
  if yms(J)-(round(yms(J),100))=mon(Z) then s_day(J)=wcdays(Z);
end;
```

```
yms(J)=iym;
e_day(J)=yint_d;
```

```
if yms(J)<dliym then do;
  yms(J)=dliym;
  s_day(J)=ddliday;
end;
end;
```

/\*\* 3.a If currently receiving and weeks are unknown then take DLI as start date and Current Interview Date

```
as stop date. **/
if (-3 le wks(J) lt 0 and cwks(J)=1) then do;
  yms(J)=dliym;
  yme(J)=iym;
  s_day(J)=ddliday;
  e_day(J)=yint_d;
end;
```

/\*\* 3.c If weeks are known, then count back from interview date to find the most recent year the respondent

could have begun receiving and receive for that number of months. Set start date as January of that year. If currently not receiving then count forward the number of months from January of that year and set the stop date. If currently receiving then just set the Current Interview date as the stop date. Constrain start and stop dates by DLI and Current Interview Date, respectively. \*\*/

```
if (wks(J) ge 0 and cwks(J)=0) then do;
  if 0 le mos(J) le yint_m then do;
    yms(J)=(yint_y*100)+01;
    yme(J)=(yint_y*100)+01+mos(J);
  end;
  if (yint_m+01) le mos(J) le (yint_m+12) then do;
    yms(J)=((yint_y-1)*100)+01;
    if mos(J) le 12 then yme(J)=((yint_y-1)*100)+01+mos(J);
    if mos(J) ge 13 then yme(J)=(yint_y*100)+01+mos(J)-12;
  end;
  if (yint_m+13) le mos(J) le (yint_m+24) then do;
    yms(J)=((yint_y-2)*100)+01;
    if mos(J) le 24 then yme(J)=((yint_y-1)*100)+01+mos(J)-12;
    if mos(J) ge 25 then yme(J)=(yint_y*100)+01+mos(J)-24;
  end;
  if (yint_m+25) le mos(J) le (yint_m+36) then do;
    yms(J)=((yint_y-3)*100)+01;
    if mos(J) le 36 then yme(J)=((yint_y-1)*100)+01+mos(J)-24;
    if mos(J) ge 37 then yme(J)=(yint_y*100)+01+mos(J)-36;
```

```

end;
if (yint_m+37) le mos(J) le (yint_m+48) then do;
  yms(J)=((yint_y-4)*100)+01;
  if mos(J) le 48 then yme(J)=((yint_y-1)*100)+01+mos(J)-36;
  if mos(J) ge 49 then yme(J)=(yint_y*100)+01+mos(J)-48;
end;

s_day(J)=31;
do Z=1 to 12;
  if (yme(J)-(round(yms(J),100)))=mon(Z) then e_day(J)=wcdays(Z);
end;

if yme(J)>iym then do;
  yme(J)=iym;
  e_day(J)=yint_d;
end;
end;

eflag(J)=3;
end;

```

```

if yme(J)>0 and yms(J)=. then do;
  yms(J)=dliym;
  s_day(J)=ddliday;
  if j=1 then eflag(J)=3;
  else if j>1 then eflag(J)=7;
end;

```

/\*\* Note: If either stop month or year is ge -3 then it implies that this unemp spell has ended. Even if the respondent reports currently receiving that would imply another loop. \*\*/

/\*\* 4. If stop year is known and stop month is unknown. \*\*/  
 if yms(J)>0 and ye(J)>0 and -3 le me(J) le -1 then do;

/\*\* 4.a If weeks are known, then count forward from start year. If the number of months falls short of the

stop year, then use January of the end year as the stop date. If the number of months exceeds the stop year, then end the array in the December of the stop year. Constrain stop date by Current Interview Date. If currently receiving then use interview date as the stop date. \*\*/

```

if wks(J) ge 0 then do;
  ymsmo(J)=yms(J)- round (yms(J),100);
  if 01 le (ymsmo(J)+mos(J)) le 12 then yme(J)=yms(J)+mos(J);
  if 13 le (ymsmo(J)+mos(J)) le 24 then yme(J)=yms(J)+100+mos(J)-12;
  if 25 le (ymsmo(J)+mos(J)) le 36 then yme(J)=yms(J)+200+mos(J)-24;
  if 37 le (ymsmo(J)+mos(J)) le 48 then yme(J)=yms(J)+300+mos(J)-36;
  if 49 le (ymsmo(J)+mos(J)) le 60 then yme(J)=yms(J)+400+mos(J)-48;
  if 61 le (ymsmo(J)+mos(J)) le 72 then yme(J)=yms(J)+500+mos(J)-60;
  if 73 le (ymsmo(J)+mos(J)) le 84 then yme(J)=yms(J)+600+mos(J)-72;

do Z=1 to 12;
  if yme(J)-round(yme(J),100)=mon(Z) then e_day(J)=wcdays(Z);

```

```

end;

if yme(J)>((ye(J)*100)+12) then do;
  yme(J)=((ye(J)*100)+12);
  e_day(J)=31;
end;

if yme(J)<((ye(J)*100)+01) then do;
  yme(J)=((ye(J)*100)+01);
  e_day(J)=31;
end;

if yme(J)>iym then do;
  yme(J)=iym;
  e_day(J)=yint_d;
end;
end;

```

/\*\* 4.b If weeks are unknown, then set December of stop year as the stop month. \*\*/

```

if -3 le wks(J) lt 0 then do;
  yme(J)=((ye(J)*100)+12);
  e_day(J)=31;
end;

```

```

if yme(J)>iym then do;
  yme(J)=iym;
  e_day(J)=yint_d;
end;

```

```

eflag(J)=4;
end;

```

/\*\* 5. If stop year is unknown. \*\*/  
 if -3 le ye(J) le -1 then do;

/\*\* 5.a If weeks are known then count forward by the number of weeks from the start date and constrain stop \*\*/

```

date by Current Interview date.
if yms(J)>0 and wks(J) ge 0 then do;
  ymsmo(J)=yms(J)- round (yms(J),100);
  if 01 le (ymsmo(J)+mos(J)) le 12 then yme(J)=yms(J)+mos(J);
  if 13 lt (ymsmo(J)+mos(J)) le 24 then yme(J)=yms(J)+100+mos(J)-12;
  if 25 lt (ymsmo(J)+mos(J)) le 36 then yme(J)=yms(J)+200+mos(J)-24;
  if 37 le (ymsmo(J)+mos(J)) le 48 then yme(J)=yms(J)+300+mos(J)-36;
  if 49 le (ymsmo(J)+mos(J)) le 60 then yme(J)=yms(J)+400+mos(J)-48;
  if 61 le (ymsmo(J)+mos(J)) le 72 then yme(J)=yms(J)+500+mos(J)-60;
  if 73 le (ymsmo(J)+mos(J)) le 84 then yme(J)=yms(J)+600+mos(J)-72;

  do Z=1 to 12;
    if yme(J)-round(yme(J),100)=mon(Z) then e_day(J)=wcdays(Z);
  end;

```

```

    if yme(J)>iym then do;
        yme(J)=iym;
        e_day(J)=yint_d;
    end;
end;

```

/\*\* 5.b If weeks are unknown then set stop date as December of start year. \*\*/

```

if yms(J)>0 and -3 le wks(J) lt 0 then do;
    yme(J)=(round (yms(J),100))+12;
    e_day(J)=31;
end;

```

```

if yme(J)>iym then do;
    yme(J)=iym;
    e_day(J)=yint_d;
end;

```

```

    eflag(J)=5;
end;

```

/\*\* 6. If the start and stop years are unknown and the weeks are unknown, use current and last interview date. \*\*/

```

if -3 le ys(J) le -1 and cwks(J)=0 and -3 le wks(J) lt 0 then do;
    yme(J)=iym;
    e_day(J)=yint_d;
    yms(J)=dliym;
    s_day(J)=ddliday;
    if j=1 then eflag(J)=6;
    else if j>1 then eflag(J)=7;
end;

```

/\*\* This portion calculates the number of days receiving if start and end year and months are the same \*\*/

```

newms(J)=(yms(J)- round (yms(J),100));
newme(J)=(yme(J)- round (yme(J),100));
if (yms(J)=yme(J) and yms(J) gt 0 and yms(J) ne . and newms(J) gt 0 and newms(J) ne .
and e_day(J) ne . and e_day(J) ge 0 and s_day(J) ne . and s_day(J) ge 0) then do;
    SAMEMONTH=1;
    if (newms(J)=1 OR newms(J)=3 OR newms(J)=5 OR newms(J)=7
OR newms(J)=8 OR newms(J)=10 OR newms(J)=12) then do;
        if e_day(J) ge (32-s_day(J)) then do;
            s_day(J)=e_day(J)-(32-s_day(J))+1;
            e_day(J)=0;
        end;
        else if e_day(J) lt (32-s_day(J)) then do;
            s_day(J)=0;
            e_day(J)=0;
            HANDEDIT=1;
        end;
    end;
end;

```

```

if (newms(J)=4 OR newms(J)=6 OR newms(J)=9 OR newms(J)=11) then do;
  if e_day(J) ge (31-s_day(J)) then do;
    s_day(J)=e_day(J)-(31-s_day(J))+1;
    e_day(J)=0;
  end;
  else if e_day(J) lt (31-s_day(J)) then do;
    s_day(J)=0;
    e_day(J)=0;
    HANDEDIT=1;
  end;
end;

if newms(J)=2 then do;
  if e_day(J) ge (29-s_day(J)) then do;
    s_day(J)=e_day(J)-(29-s_day(J))+1;
    e_day(J)=0;
  end;
  else if e_day(J) lt (29-s_day(J)) then do;
    s_day(J)=0;
    e_day(J)=0;
    HANDEDIT=1;
  end;
end;
end;

```

/\*\* This portion of the program uses the category reported by the respondent to create an estimated amount. The estimated amount is the midpoint rounded down. Note that the 12th category lists \$1251 as the amount. This amount was chosen since the category is unbounded - the number represents one dollar above the lower bound. \*/

```

if -3 le inc(J) le -1 and einc(J)=1 then inc(J)=(50/7);
if -3 le inc(J) le -1 and einc(J)=2 then inc(J)=(125/7);
if -3 le inc(J) le -1 and einc(J)=3 then inc(J)=(175/7);
if -3 le inc(J) le -1 and einc(J)=4 then inc(J)=(225/7);
if -3 le inc(J) le -1 and einc(J)=5 then inc(J)=(275/7);
if -3 le inc(J) le -1 and einc(J)=6 then inc(J)=(351/7);
if inc(J)>-4 then aflag(J)=0;

```

```

csm(J)=(round(yms(J),100)-198000)*.12+(yms(J)-round(yms(J),100));
cem(J)=(round(yme(J),100)-198000)*.12+(yme(J)-round(yme(J),100));
cdli=(round(dliym,100)-198000)*.12+(dliym-round(dliym,100))-1;

```

```

/** added to zero out people who back report.                                **/
end;                                                                                   /** [E2] **/

```

```

/** HANDEDITS - 1                                                                **/
/** This person reports receiving WC from Nov 9, 2001 to Nov 7, 2001. So we switched the start and
end dates
which makes the number of days this person received WC as 3.                                                                **/
if pubid=3225 then do; sdayc21=3; edayc21=0; DAYS_PROB=1; end;

```

```

/** This portion creates the month receiving indicator (WC) and the amount receiving each month
(WCAMT) */
do J=1 to 2;
  C=0;
  do L=1 to 269;
    C=C+1;
    if 0 le L le doicm then do;
      if csm(J) le C le cem(J) then do;
        if (C=csm(J) and inc(J) ge 0) then do;
          wc(L)=eflag(J);
          if s_day(J) ne . then wcamt(L)=round((inc(J)*s_day(J)), 1);
        end;

      else if (C=cem(J) and inc(J) ge 0) then do;
        wc(L)=eflag(J);
        if e_day(J) ne . then wcamt(L)=round((inc(J)*e_day(J)), 1);
      end;

      else if csm(J)<C and C<cem(J) and inc(J) ge 0 then do;
        wc(L)=eflag(J);
        if C in (169,171,173,175,176,178,180,181,183,185,187,188,
          190,192,193,195,197,199,200,202,204,205,207,209,
          211,212,214,216,217,219,221,223,224,226,228,229,
          231,233,235,236,238,240,241,243,245,247,248,250,
          252,253,255,257,259,260,262,264,265,267,269) then do;
          wcamt(L)=round((inc(J)*31),1);
        end;
      else if C in (172,174,177,179,184,186,189,191,196,198,201,203,
          208,210,213,215,220,222,225,227,232,234,237,239,
          244,246,249,251,256,258,261,263,268) then do;
          wcamt(L)=round((inc(J)*30),1);
        end;
      else if C in (194,242) then do;
          wcamt(L)=round((inc(J)*29),1);
        end;
      else if C in (170,182,206,218,230,254,266) then do;
          wcamt(L)=round((inc(J)*28),1);
        end;
      end;

      else if inc(J)<0 then do;
        wc(L)=eflag(J);
        wcamt(L)=inc(J);
      end;

      if wcamt(L) le 1000 and (wcodd=0 OR wcodd=.) then do; wcodd=0; end;
      if wcamt(L)>1000 then do; wcodd=wcamt(L); end;
      if wc(L)>0 and wcamt(L)=. then do; wcck=wcck+1; end;
      if (wcamt(L)>0 OR (-3 le wcamt(L) lt 0)) and wc(L)=0 then do; wcckm=wcckm+1; end;
      if wc(L)=-3 and wcamt(L) ne -3 then do; wcckmm=wcckmm+1; end;
      if wcamt(L)=0 then dummywc=1;
    end;
  end;
end;

```

```

    end;

    end;
end;
end;

/** This portion deletes spells that started before DLI                                **/
do J=1 to 2;
  C=0;
  do L=1 to 269;
    C=C+1;
    if csm(J) le cdli and p010 eq 1 and csm(J) ne . then do;
      if csm(J) le C le cdli then do;
        wc(L)=0;
        wcamt(L)=.;
      end;
    end;
  end;
end;
end;
end;
end;
/** [E1] **/

/** This portion creates the months receiving in each year variables (WCMXX) and the amount
receiving **
** each year variables (CAMTXX). **/
do I=1 to 23;
  if p1210>-4 OR p1210=-5 then do;
    wcm(I)=0;
    camt(I)=0;
    wca(I)=0;
    wcn(I)=0;
  end;
end;

if p1210>-4 OR p1210=-5 then do;
/** [B3] **/

/** calculate total no. months to divide the problematic amts over. **/
do L=1 to 269;
  if wc(L)>0 and wcamt(L)>1000 and wcodd>0 then do;
    wcmnths=wcmnths+1;
  end;
end;

do L=1 to 269;
/** [B4] **/
  if 1 le L le 12 then do;
    if wcm80 ge 0 and wc(L)>0 then do;
      wcm80=wcm80+1;
    end;
    if wc(L)>1 then wcm80=-3;
    if -3 le wcamt(L) le -1 then do;
      wcn80=wcn80+1;
      camt80=wcamt(L);
    end;
  end;
end;

```



```
end;  
if wcamt(L) ge 0 then do;  
  wca80=wca80+1;  
  if wca80=1 and wcn80=0 then camt80=wcamt(L);  
  if wca80>1 and wcn80=0 then camt80=camt80+wcamt(L);  
  if wcm80=-3 then camt80=-3;  
end;  
end;
```

```
if 13 le L le 24 then do;  
  if wcm81 ge 0 and wc(L)>0 then do;  
    wcm81=wcm81+1;  
  end;  
  if wc(L)>1 then wcm81=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn81=wc(L)+1;  
    camt81=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca81=wca81+1;  
    if wca81=1 and wcn81=0 then camt81=wcamt(L);  
    if wca81>1 and wcn81=0 then camt81=camt81+wcamt(L);  
    if wcm81=-3 then camt81=-3;  
  end;  
end;
```

```
if 25 le L le 36 then do;  
  if wcm82 ge 0 and wc(L)>0 then do;  
    wcm82=wcm82+1;  
  end;  
  if wc(L)>1 then wcm82=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn82=wc(L)+1;  
    camt82=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca82=wca82+1;  
    if wca82=1 and wcn82=0 then camt82=wcamt(L);  
    if wca82>1 and wcn82=0 then camt82=camt82+wcamt(L);  
    if wcm82=-3 then camt82=-3;  
  end;  
end;
```

```
if 37 le L le 48 then do;  
  if wcm83 ge 0 and wc(L)>0 then do;  
    wcm83=wcm83+1;  
  end;  
  if wc(L)>1 then wcm83=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn83=wc(L)+1;  
    camt83=wcamt(L);  
end;
```

```
end;  
if wcamt(L) ge 0 then do;  
  wca83=wca83+1;  
  if wca83=1 and wcn83=0 then camt83=wcamt(L);  
  if wca83>1 and wcn83=0 then camt83=camt83+wcamt(L);  
  if wcm83=-3 then camt83=-3;  
end;  
end;
```

```
if 49 le L le 60 then do;  
  if wcm84 ge 0 and wc(L)>0 then do;  
    wcm84=wcm84+1;  
  end;  
  if wc(L)>1 then wcm84=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn84=wc(L)+1;  
    camt84=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca84=wca84+1;  
    if wca84=1 and wcn84=0 then camt84=wcamt(L);  
    if wca84>1 and wcn84=0 then camt84=camt84+wcamt(L);  
    if wcm84=-3 then camt84=-3;  
  end;  
end;
```

```
if 61 le L le 72 then do;  
  if wcm85 ge 0 and wc(L)>0 then do;  
    wcm85=wcm85+1;  
  end;  
  if wc(L)>1 then wcm85=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn85=wc(L)+1;  
    camt85=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca85=wca85+1;  
    if wca85=1 and wcn85=0 then camt85=wcamt(L);  
    if wca85>1 and wcn85=0 then camt85=camt85+wcamt(L);  
    if wcm85=-3 then camt85=-3;  
  end;  
end;
```

```
if 73 le L le 84 then do;  
  if wcm86 ge 0 and wc(L)>0 then do;  
    wcm86=wcm86+1;  
  end;  
  if wc(L)>1 then wcm86=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn86=wc(L)+1;  
    camt86=wcamt(L);  
end;
```

```
end;  
if wcamt(L) ge 0 then do;  
  wca86=wca86+1;  
  if wca86=1 and wcn86=0 then camt86=wcamt(L);  
  if wca86>1 and wcn86=0 then camt86=camt86+wcamt(L);  
  if wcm86=-3 then camt86=-3;  
end;  
end;
```

```
if 85 le L le 96 then do;  
  if wcm87 ge 0 and wc(L)>0 then do;  
    wcm87=wcm87+1;  
  end;  
  if wc(L)>1 then wcm87=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn87=wc(L)+1;  
    camt87=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca87=wca87+1;  
    if wca87=1 and wcn87=0 then camt87=wcamt(L);  
    if wca87>1 and wcn87=0 then camt87=camt87+wcamt(L);  
    if wcm87=-3 then camt87=-3;  
  end;  
end;
```

```
if 97 le L le 108 then do;  
  if wcm88 ge 0 and wc(L)>0 then do;  
    wcm88=wcm88+1;  
  end;  
  if wc(L)>1 then wcm88=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn88=wc(L)+1;  
    camt88=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca88=wca88+1;  
    if wca88=1 and wcn88=0 then camt88=wcamt(L);  
    if wca88>1 and wcn88=0 then camt88=camt88+wcamt(L);  
    if wcm88=-3 then camt88=-3;  
  end;  
end;
```

```
if 109 le L le 120 then do;  
  if wcm89 ge 0 and wc(L)>0 then do;  
    wcm89=wcm89+1;  
  end;  
  if wc(L)>1 then wcm89=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn89=wc(L)+1;  
    camt89=wcamt(L);  
end;
```

```
end;  
if wcamt(L) ge 0 then do;  
  wca89=wca89+1;  
  if wca89=1 and wcn89=0 then camt89=wcamt(L);  
  if wca89>1 and wcn89=0 then camt89=camt89+wcamt(L);  
  if wcm89=-3 then camt89=-3;  
end;  
end;
```

```
if 121 le L le 132 then do;  
  if wcm90 ge 0 and wc(L)>0 then do;  
    wcm90=wcm90+1;  
  end;  
  if wc(L)>1 then wcm90=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn90=wc(L)+1;  
    camt90=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca90=wca90+1;  
    if wca90=1 and wcn90=0 then camt90=wcamt(L);  
    if wca90>1 and wcn90=0 then camt90=camt90+wcamt(L);  
    if wcm90=-3 then camt90=-3;  
  end;  
end;
```

```
if 133 le L le 144 then do;  
  if wcm91 ge 0 and wc(L)>0 then do;  
    wcm91=wcm91+1;  
  end;  
  if wc(L)>1 then wcm91=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn91=wc(L)+1;  
    camt91=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca91=wca91+1;  
    if wca91=1 and wcn91=0 then camt91=wcamt(L);  
    if wca91>1 and wcn91=0 then camt91=camt91+wcamt(L);  
    if wcm91=-3 then camt91=-3;  
  end;  
end;
```

```
if 145 le L le 156 then do;  
  if wcm92 ge 0 and wc(L)>0 then do;  
    wcm92=wcm92+1;  
  end;  
  if wc(L)>1 then wcm92=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn92=wc(L)+1;  
    camt92=wcamt(L);  
  end;  
end;
```

```
end;  
if wcamt(L) ge 0 then do;  
  wca92=wca92+1;  
  if wca92=1 and wcn92=0 then camt92=wcamt(L);  
  if wca92>1 and wcn92=0 then camt92=camt92+wcamt(L);  
  if wcm92=-3 then camt92=-3;  
end;  
end;
```

```
if 157 le L le 168 then do;  
  if wcm93 ge 0 and wc(L)>0 then do;  
    wcm93=wcm93+1;  
  end;  
  if wc(L)>1 then wcm93=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn93=wc93+1;  
    camt93=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca93=wca93+1;  
    if wca93=1 and wcn93=0 then camt93=wcamt(L);  
    if wca93>1 and wcn93=0 then camt93=camt93+wcamt(L);  
    if wcm93=-3 then camt93=-3;  
  end;  
end;
```

```
if 169 le L le 180 then do;  
  if wcm94 ge 0 and wc(L)>0 then do;  
    wcm94=wcm94+1;  
  end;  
  if wc(L)>1 then wcm94=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn94=wc94+1;  
    camt94=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca94=wca94+1;  
    if wca94=1 and wcn94=0 then camt94=wcamt(L);  
    if wca94>1 and wcn94=0 then camt94=camt94+wcamt(L);  
    if wcm94=-3 then camt94=-3;  
  end;  
end;
```

```
if 181 le L le 192 then do;  
  if wcm95 ge 0 and wc(L)>0 then do;  
    wcm95=wcm95+1;  
  end;  
  if wc(L)>1 then wcm95=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn95=wc95+1;  
    camt95=wcamt(L);  
end;
```

```
end;  
if wcamt(L) ge 0 then do;  
  wca95=wca99+1;  
  if wca95=1 and wcn95=0 then camt95=wcamt(L);  
  if wca95>1 and wcn95=0 then camt95=camt95+wcamt(L);  
  if wcm95=-3 then camt95=-3;  
end;  
end;
```

```
if 193 le L le 204 then do;  
  if wcm96 ge 0 and wc(L)>0 then do;  
    wcm96=wcm96+1;  
  end;  
  if wc(L)>1 then wcm96=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn96=wc96+1;  
    camt96=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca96=wca96+1;  
    if wca96=1 and wcn96=0 then camt96=wcamt(L);  
    if wca96>1 and wcn96=0 then camt96=camt96+wcamt(L);  
    if wcm96=-3 then camt96=-3;  
  end;  
end;
```

```
if 205 le L le 216 then do;  
  if wcm97 ge 0 and wc(L)>0 then do;  
    wcm97=wcm97+1;  
  end;  
  if wc(L)>1 then wcm97=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn97=wc97+1;  
    camt97=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca97=wca97+1;  
    if wca97=1 and wcn97=0 then camt97=wcamt(L);  
    if wca97>1 and wcn97=0 then camt97=camt97+wcamt(L);  
    if wcm97=-3 then camt97=-3;  
  end;  
end;
```

```
if 217 le L le 228 then do;  
  if wcm98 ge 0 and wc(L)>0 then do;  
    wcm98=wcm98+1;  
  end;  
  if wc(L)>1 then wcm98=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn98=wc98+1;  
    camt98=wcamt(L);  
  end;  
end;
```

```
end;  
if wcamt(L) ge 0 then do;  
  wca98=wca98+1;  
  if wca98=1 and wcn98=0 then camt98=wcamt(L);  
  if wca98>1 and wcn98=0 then camt98=camt98+wcamt(L);  
  if wcm98=-3 then camt98=-3;  
end;  
end;
```

```
if 229 le L le 240 then do;  
  if wcm99 ge 0 and wc(L)>0 then do;  
    wcm99=wcm99+1;  
  end;  
  if wc(L)>1 then wcm99=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn99=wc(L)+1;  
    camt99=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca99=wca99+1;  
    if wca99=1 and wcn99=0 then camt99=wcamt(L);  
    if wca99>1 and wcn99=0 then camt99=camt99+wcamt(L);  
    if wcm99=-3 then camt99=-3;  
  end;  
end;
```

```
if 241 le L le 252 then do;  
  if wcm00 ge 0 and wc(L)>0 then do;  
    wcm00=wcm00+1;  
  end;  
  if wc(L)>1 then wcm00=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn00=wc(L)+1;  
    camt00=wcamt(L);  
  end;  
  if wcamt(L) ge 0 then do;  
    wca00=wca00+1;  
    if wca00=1 and wcn00=0 then camt00=wcamt(L);  
    if wca00>1 and wcn00=0 then camt00=camt00+wcamt(L);  
    if wcm00=-3 then camt00=-3;  
  end;  
end;
```

```
if 253 le L le 264 then do;  
  if wcm01 ge 0 and wc(L)>0 then do;  
    wcm01=wcm01+1;  
  end;  
  if wc(L)>1 then wcm01=-3;  
  if -3 le wcamt(L) le -1 then do;  
    wcn01=wc(L)+1;  
    camt01=wcamt(L);  
  end;  
end;
```





```
wcatotal(I)=-5;
end;
end;
```

```
/** To change all invalid skips to -3. **/
```

```
array vars wcatotal80-wcatotal99 wcatotal00 wcatotal01 wcatotal02;
```

```
do over vars;
  if vars=-2 then vars=-3;
  if vars=-1 then vars=-3;
end;
```

---

**/\*\* \*\*\*\*\* SECTION 3: OTHER GOVERNMENT ASSISTANCE \*\*\*\*\* \*/**

```
*** change previous round variables to account for noninterviews;
```

```
array gtotal (I) gtotal80-gtotal99 gtotal00 gtotal01 gtotal02;
array oldgtotal (I) oldgtotal80-oldgtotal99 oldgtotal00 oldgtotal01 oldgtotal02;
array g4total (I) g4total80-g4total99 g4total00 g4total01 g4total02;
array g3total (I) g3total80-g3total99 g3total00 g3total01 g3total02;
array g2total (I) g2total80-g2total99 g2total00 g2total01 g2total02;
array g1total (I) g1total80-g1total99 g1total00 g1total01 g1total02;
```

```
array gatotal (I) gatotal80-gatotal99 gatotal00 gatotal01 gatotal02;
array oldgatotal (I) oldgatotal80-oldgatotal99 oldgatotal00 oldgatotal01 oldgatotal02;
array ga4total (I) ga4total80-ga4total99 ga4total00 ga4total01 ga4total02;
array ga3total (I) ga3total80-ga3total99 ga3total00 ga3total01 ga3total02;
array ga2total (I) ga2total80-ga2total99 ga2total00 ga2total01 ga2total02;
array ga1total (I) ga1total80-ga1total99 ga1total00 ga1total01 ga1total02;
```

```
*** if r4 interview, use r4 created var;
*** if noninterview in r4, then use r3 created var;
*** if noninterview in r3, then use r2 variable;
*** if noninterview in r2, then use r1 variable;
*** if variable not in r1, r2 or r3 then set to 0;
```

```
do I=1 to 23;
  gtotal=g4total;
  if g4total=-5 then do;
    gtotal=g3total;
    if g3total=-5 then do;
      gtotal=g2total;
      if g2total=-5 then do;
        gtotal=g1total;
      end;
    end;
  end;
end;
```

```
gatotal=ga4total;
if ga4total=-5 then do;
  gatotal=ga3total;
  if ga3total=-5 then do;
    gatotal=ga2total;
    if ga2total=-5 then do;
      gatotal=ga1total;
    end;
  end;
end;
end;
```

```
oldgtotal=gtotal;
oldgatotal=gatotal;
```

/\* To take care of cases in earlier rounds where although the R did not receive any govt. programs, the amount

received variable is 0 instead of -4. \*/

```
if gttotal=0 and gatotal=0 then do;
  oldgatotal=-4;
end;
if gttotal=. then do;
  oldgtotal=-4;
  gttotal=0;
end;
if gatotal=. then do;
  oldgatotal=-4;
  gatotal=0;
end;
end;
```

```
gprg_ever=g4prg_ever;
if g4prg_ever=-5 then do;
  gprg_ever=g3prg_ever;
  if g3prg_ever=-5 then do;
    gprg_ever=g2prg_ever;
    if g2prg_ever=-5 then do;
      gprg_ever=g1prg_ever;
    end;
  end;
end;
end;
```

/\*\* HANDEDITS (0) In these cases the total months ever received was wrong in the earlier rounds. \*\*/

```
if pubid=1198 then do; gprg_ever=-3; end;
if pubid=1428 then do; gprg_ever=41; end;
if pubid=3806 then do; gprg_ever=5; end;
if pubid=3824 then do; gprg_ever=-3; end;
if pubid=7104 then do; gprg_ever=-3; end;
if pubid=7131 then do; gprg_ever=21; end;
if pubid=9018 then do; gprg_ever=30; end;
if pubid=4303 then do; gprg_ever=45; end;
if pubid=4329 then do; gprg_ever=46; end;
```

```

if pubid=6977 then do; gprg_ever=219; end;
if pubid=8092 then do; gprg_ever=81; end;
if pubid=8748 then do; gprg_ever=47; end;

if gprg_ever=-4 then do;
  gprg_ever=0;
end;

ack=0; ackm=0; ackh=0; fck=0; fckm=0; fckh=0; wck=0; wckm=0; wckh=0; ock=0; ockm=0; ockh=0;

cm14=((BDATE_Y+14)-1980)*12+BDATE_M;
cmb=((BDATE_Y)-1980)*12+BDATE_M;
ym14=(BDATE_Y*100+BDATE_M)+1400;
iym=YINT_Y*100+YINT_M;

if p1210=1 then do;
  if int00_y=-5 then do;          /*not interviewed in round4*/
    if int99_y=-5 then do;      /*not interviewed in round 4 & 3*/
      if int98_y=-5 then do;    /*not interviewed round 4,3 & 2*/
        if int97_m lt 12 then dliym=int97_y*100+(int97_m+1);
        if int97_m=12 then dliym=((int97_y+1)*100)+1;
        dllicm=(int97_y-1980)*12+(int97_m+1);
      end;
      if int98_y ne -5 then do;  /*not interviewed in round 4 & 3 BUT in 2*/
        if int98_m lt 12 then dliym=int98_y*100+(int98_m+1);
        if int98_m=12 then dliym=((int98_y+1)*100)+1;
        dllicm=(int98_y-1980)*12+(int98_m+1);
      end;
    end;
    if int99_y ne -5 then do;    /*not interviewed in round 4 BUT in 3*/
      if int99_m lt 12 then dliym=int99_y*100+(int99_m+1);
      if int99_m=12 then dliym=((int99_y+1)*100)+1;
      dllicm=(int99_y-1980)*12+(int99_m+1);
    end;
  end;
  if int00_y ne -5 then do;     /*interviewed in round4*/
    if int00_m lt 12 then dliym=int00_y*100+(int00_m+1);
    if int00_m=12 then dliym=((int00_y+1)*100)+1;
    dllicm=(int00_y-1980)*12+(int00_m+1);
  end;
end;

end;

if p1210 ne 1 then dliym=ym14;
if p1210 ne 1 then dllicm=cm14;

doicm=(YINT_Y-1980)*12+YINT_M;
aaicm=(YINT_Y-BDATE_Y)*12+YINT_M-BDATE_M;

array F (L) F001-F269;
array famt (L) famt001-famt269;

```

```
array fhhm (L) fhhm001-fhhm269;
array W (L) W001-W269;
array wamt (L) wamt001-wamt269;
array whhm (L) whhm001-whhm269;
array A (L) A001-A269;
array aamt (L) aamt001-aamt269;
array ahhm (L) ahhm001-ahhm269;
array O (L) O001-O269;
array oamt (L) oamt001-oamt269;
array ohhm (L) ohhm001-ohhm269;
```

```
/*2. [b1] begin - Rs eligible for a program in round 5*/
if P1210>-4 then do;
```

```
  if PS35707>-3 then P16200=PS35707;           /**AFDC**/
  if PS16377>-3 then P18900=PS16377;           /**Food Stamps**/
  if PS36087>-3 then P30500=PS36087;           /**Other**/
  if PS21397>-3 then P21500=PS21397;           /**WIC**/
```

```
do L=1 to 269;
  if cmb le L le doicm then do;
    A=0;
    W=0;
    F=0;
    O=0;
  end;
end;
```

/\*\* HANDEDIT (1a): These repondents reported invalid start and stop date, BUT were never asked if they received again. If they reported currently receiving in this section we impute the stop date without using that information i.e. coding them as NOT currently receiving. Then in HANDEDIT (1b) we code them again as currently receiving. Continued in HANDEDIT (1b). \*\*/

```
if pubid=227 then do; ps16410=0; ps21100A1=0; end;
if pubid=8409 then do; ps16410=0; ps21100A1=0; end;
if pubid=8627 then do; ps16410=0; ps21100A1=0; end;
if pubid=7575 then do; p18900=0; end;
if pubid=838 then do; ps36120=0; ps32700A1=0; end;
if pubid=2749 then do; ps36120=0; ps32700A1=0; end;
if pubid=7596 then do; ps36120=0; ps32700A1=0; end;
if pubid=2930 then do; p21500=0; end;
if pubid=8219 then do; p222001=0; p21500=0; end;
```

```
/** START YEAR information **/
```

```
array ysf1 (J) PS191001Y PS191002Y PS191003Y PS191004Y;           /**food stamps, sdli**/
array ysf2 (J) P194001Y P194002Y P194003Y P194004Y;           /**food stamps, int**/
array ysw1 (J) PS220001Y PS220002Y PS220003Y PS220004Y;           /**wic, sdli**/
array ysw2 (J) P220001Y P220002Y P220003Y P220004Y;           /**wic, int**/
array ysa1 (J) PS358001Y PS358002Y PS358003Y PS358004Y;           /**afdc, sdli**/
array ysa2 (J) P167001Y P167002Y P167003Y P167004Y;           /**afdc, int**/
array yso1 (J) PS310001Y PS310002Y PS310003Y PS310004Y;           /**other, sdli**/
array yso2 (J) P310001Y P310002Y P310003Y P310004Y;           /**other, int**/
```

```
/** START MONTH information **/  
array msf1 (J) PS191001M PS191002M PS191003M PS191004M; /**food stamps, sdli**/  
array msf2 (J) P194001M P194002M P194003M P194004M; /**food stamps, int**/  
array msw1 (J) PS220001M PS220002M PS220003M PS220004M; /**wic, sdli**/  
array msw2 (J) P220001M P220002M P220003M P220004M; /**wic, int**/  
array msa1 (J) PS358001M PS358002M PS358003M PS358004M; /**afdc, sdli**/  
array msa2 (J) P167001M P167002M P167003M P167004M; /**afdc, int**/  
array mso1 (J) PS310001M PS310002M PS310003M PS310004M; /**other, sdli**/  
array mso2 (J) P310001M P310002M P310003M P310004M; /**other, int**/
```

```
/** end YEAR information **/  
array yef1 (J) PS200001Y PS200002Y PS200003Y PS200004Y; /**food stamps, sdli**/  
array yef2 (J) P200001Y P200002Y P200003Y P200004Y; /**food stamps, int**/  
array yew1 (J) PS226001Y PS226002Y PS226003Y PS226004Y; /**wic, sdli**/  
array yew2 (J) P226001Y P226002Y P226003Y P226004Y; /**wic, int**/  
array yea1 (J) PS358401Y PS358402Y PS358403Y PS358404Y; /**afdc, sdli**/  
array yea2 (J) P172001Y P172002Y P172003Y P172004Y; /**afdc, int**/  
array yeo1 (J) PS316001Y PS316002Y PS316003Y PS316004Y; /**other, sdli**/  
array yeo2 (J) P316001Y P316002Y P316003Y P316004Y; /**other, int**/
```

```
/** end MONTH information **/  
array mef1 (J) PS200001M PS200002M PS200003M PS200004M; /**food stamps, sdli**/  
array mef2 (J) P200001M P200002M P200003M P200004M; /**food stamps, int**/  
array mew1 (J) PS226001M PS226002M PS226003M PS226004M; /**wic, sdli**/  
array mew2 (J) P226001M P226002M P226003M P226004M; /**wic, int**/  
array mea1 (J) PS358401M PS358402M PS358403M PS358404M; /**afdc, sdli**/  
array mea2 (J) P172001M P172002M P172003M P172004M; /**afdc, int**/  
array meo1 (J) PS316001M PS316002M PS316003M PS316004M; /**other, sdli**/  
array meo2 (J) P316001M P316002M P316003M P316004M; /**other, int**/
```

```
array ymsf1 (J) ymsf11-ymsf14;  
array ymsf2 (J) ymsf21-ymsf24;  
array ymsw1 (J) ymsw11-ymsw14;  
array ymsw2 (J) ymsw21-ymsw24;  
array ymsa1 (J) ymsa11-ymsa14;  
array ymsa2 (J) ymsa21-ymsa24;  
array ymso1 (J) ymso11-ymso14;  
array ymso2 (J) ymso21-ymso24;
```

```
array csmf1 (J) csmf11-csmf14;  
array csmf2 (J) csmf21-csmf24;  
array csmw1 (J) csmw11-csmw14;  
array csmw2 (J) csmw21-csmw24;  
array csma1 (J) csma11-csma14;  
array csma2 (J) csma21-csma24;  
array csmo1 (J) csmo11-csmo14;  
array csmo2 (J) csmo21-csmo24;
```

```
array ymef1 (J) ymef11-ymef14;  
array ymef2 (J) ymef21-ymef24;
```

array ymew1 (J) ymew11-ymew14;  
array ymew2 (J) ymew21-ymew24;  
array ymeal (J) ymeal1-ymeal4;  
array ymeal2 (J) ymeal21-ymeal24;  
array ymeo1 (J) ymeo11-ymeo14;  
array ymeo2 (J) ymeo21-ymeo24;

array cemf1 (J) cemf11-cemf14;  
array cemf2 (J) cemf21-cemf24;  
array cemw1 (J) cemw11-cemw14;  
array cemw2 (J) cemw21-cemw24;  
array cema1 (J) cema11-cema14;  
array cema2 (J) cema21-cema24;  
array cemo1 (J) cemo11-cemo14;  
array cemo2 (J) cemo21-cemo24;

/\*\* CURRENTLY receiving information \*\*/

array curf1 (J) PS21100A1 PS21100A2 PS21100A3 PS21100A4; /\*\*food stamps, no gap  
sdli\*\*/  
array curf2 (J) P196001 P196002 P196003 P196004; /\*\*food stamps, int\*\*/  
array curw1 (J) PS23700A1 PS23700A2 PS23700A3 PS23700A4; /\*\*wic, no gap sdli\*\*/  
array curw2 (J) P222001 P222002 P222003 P222004; /\*\*wic, int\*\*/  
array cura1 (J) PS359501 PS359502 PS359503 PS359504; /\*\*afdc, no gap sdli\*\*/  
array cura2 (J) P169001 P169002 P169003 P169004; /\*\*afdc, int\*\*/  
array curo1 (J) PS32700A1 PS32700A2 PS32700A3 PS32700A4; /\*\*other, no gap sdli\*\*/  
array curo2 (J) P312001 P312002 P312003 P312004; /\*\*other, int\*\*/

/\*\* EDIT FLAGS for dates \*\*/

array eflagf1 (J) eflagf11-eflagf14; /\*\*food stamps, sdli\*\*/  
array eflagf2 (J) eflagf21-eflagf24; /\*\*food stamps, int\*\*/  
array eflagw1 (J) eflagw11-eflagw14; /\*\*wic, sdli\*\*/  
array eflagw2 (J) eflagw21-eflagw24; /\*\*wic, int\*\*/  
array eflaga1 (J) eflaga11-efлага14; /\*\*afdc, sdli\*\*/  
array eflaga2 (J) eflaga21-efлага24; /\*\*afdc, int\*\*/  
array eflago1 (J) eflago11-efлаго14; /\*\*other, sdli\*\*/  
array eflago2 (J) eflago21-efлаго24; /\*\*other, int\*\*/

/\*\* EDIT FLAGS for amounts \*\*/

array aflagf1 (J) aflagf11-aflagf14; /\*\*food stamps, sdli\*\*/  
array aflagf2 (J) aflagf21-aflagf24; /\*\*food stamps, int\*\*/  
array aflagw1 (J) aflagw11-aflagw14; /\*\*wic, sdli\*\*/  
array aflagw2 (J) aflagw21-aflagw24; /\*\*wic, int\*\*/  
array aflaga1 (J) aflaga11-aflaga14; /\*\*afdc, sdli\*\*/  
array aflaga2 (J) aflaga21-aflaga24; /\*\*afdc, int\*\*/  
array aflago1 (J) aflago11-aflago14; /\*\*other, sdli\*\*/  
array aflago2 (J) aflago21-aflago24; /\*\*other, int\*\*/

/\*\* AMOUNT RECEIVED - actual & estimated \*\*/

array incf1 (J) PS207001 PS207002 PS207003 PS207004; /\*\*food stamps, amount  
sdli\*\*/  
array incf2 (J) P207001 P207002 P207003 P207004; /\*\*food stamps, amount int\*\*/

```

array eincf1 (J) PS208001 PS208002 PS208003 PS208004;          /**food stamps, est amount
sdli**/
array eincf2 (J) P208001 P208002 P208003 P208004;             /**food stamps, est amount
int**/

array incw1 (J) PS233001 PS233002 PS233003 PS233004;          /**wic, amount sdli**/
array incw2 (J) P233001 P233002 P233003 P233004;             /**wic, amount int**/
array eincw1 (J) PS234001 PS234002 PS234003 PS234004;          /**wic, est amount sdli**/
array eincw2 (J) P234001 P234002 P234003 P234004;             /**wic, est amount int**/

array inca1 (J) PS359001 PS359002 PS359003 PS359004;          /**afdc, amount sdli**/
array inca2 (J) P181001 P181002 P181003 P181004;             /**afdc, amount int**/
array einca1 (J) PS359101 PS359102 PS359103 PS359104;          /**afdc, est amount sdli**/
array einca2 (J) P182001 P182002 P182003 P182004;             /**afdc, est amount int**/

array inco1 (J) PS323001 PS323002 PS323003 PS323004;          /**other, amount sdli**/
array inco2 (J) P323001 P323002 P323003 P323004;             /**other, amount int**/
array einco1 (J) PS324001 PS324002 PS324003 PS324004;          /**other, est amount sdli**/
array einco2 (J) P324001 P324002 P324003 P324004;             /**other, est amount int**/

/** ESTIMATED WEEKS - unable to answer start or stop quex**/
array wksf1 (J) PS202001 PS202002 PS202003 PS202004;          /**food stamps, est weeks
sdli**/
array wksf2 (J) P202001 P202002 P202003 P202004;             /**food stamps, est weeks int**/

array wksw1 (J) PS228001 PS228002 PS228003 PS228004;          /**wic, est weeks sdli**/
array wksw2 (J) P228001 P228002 P228003 P228004;             /**wic, est weeks int**/

array wksw1 (J) PS358601 PS358602 PS358603 PS358604;          /**afdc, est weeks sdli**/
array wksw2 (J) P175001 P175002 P175003 P175004;             /**afdc, est weeks int**/

array wkso1 (J) PS318001 PS318002 PS318003 PS318004;          /**other, est weeks sdli**/
array wkso2 (J) P318001 P318002 P318003 P318004;             /**other, est weeks int**/

array mosf1 (J) mosf11-mosf14;
array mosf2 (J) mosf21-mosf24;
array mosw1 (J) mosw11-mosw14;
array mosw2 (J) mosw21-mosw24;
array mosa1 (J) mosa11-mosa14;
array mosa2 (J) mosa21-mosa24;
array moso1 (J) moso11-moso14;
array moso2 (J) moso21-moso24;

/** PERSON receiving amount, respondent **/
array rcvrf1 (J) PS2090011 PS2090021 PS2090031 PS2090041;/**food stamps, sdli**/
array rcvrf2 (J) P2090011 P2090021 P2090031 P2090041;          /**food stamps, int**/
array rcvrw1 (J) PS2350011 PS2350021 PS2350031 PS2350041;/**wic, sdli**/
array rcvrw2 (J) P2350011 P2350021 P2350031 P2350041;          /**wic, int**/
array rcvra1 (J) PS3592011 PS3592021 PS3592031 PS3592041;/**afdc, sdli**/
array rcvra2 (J) P1830011 P1830021 P1830031 P1830041;          /**afdc, int**/
array rcvro1 (J) PS3250011 PS3250021 PS3250031 PS3250041;/**other, sdli**/

```

array rcvro2 (J) P3250011 P3250021 P3250031 P3250041; /\*\*other, int\*\*/

/\*\* PERSON receiving amount, spouse \*\*/

array rcvtf1 (J) PS2090012 PS2090022 PS2090032 PS2090042;/\*\*food stamps, sdli\*\*/

array rcvtf2 (J) P2090012 P2090022 P2090032 P2090042; /\*\*food stamps, int\*\*/

array rcvsw1 (J) PS2350012 PS2350022 PS2350032 PS2350042;/\*\*wic, sdli\*\*/

array rcvsw2 (J) P2350012 P2350022 P2350032 P2350042; /\*\*wic, int\*\*/

array rcvsa1 (J) PS3592012 PS3592022 PS3592032 PS3592042;/\*\*afdc, sdli\*\*/

array rcvsa2 (J) P1830012 P1830022 P1830032 P1830042; /\*\*afdc, int\*\*/

array rcvso1 (J) PS3250012 PS3250022 PS3250032 PS3250042;/\*\*other, sdli\*\*/

array rcvso2 (J) P3250012 P3250022 P3250032 P3250042; /\*\*other, int\*\*/

/\*\* PERSON receiving amount, child \*\*/

array rcvtf1 (J) PS2090013 PS2090023 PS2090033 PS2090043;/\*\*food stamps, sdli\*\*/

array rcvtf2 (J) P2090013 P2090023 P2090033 P2090043; /\*\*food stamps, int\*\*/

array rcvsw1 (J) PS2350013 PS2350023 PS2350033 PS2350043;/\*\*wic, sdli\*\*/

array rcvsw2 (J) P2350013 P2350023 P2350033 P2350043; /\*\*wic, int\*\*/

array rcvca1 (J) PS3592013 PS3592023 PS3592033 PS3592043;/\*\*afdc, sdli\*\*/

array rcvca2 (J) P1830013 P1830023 P1830033 P1830043; /\*\*afdc, int\*\*/

array rcvco1 (J) PS3250013 PS3250023 PS3250033 PS3250043;/\*\*other, sdli\*\*/

array rcvco2 (J) P3250013 P3250023 P3250033 P3250043; /\*\*other, int\*\*/

/\*\* PERSON receiving amount family member \*\*/

array rcvff1 (J) PS2090014 PS2090024 PS2090034 PS2090044;/\*\*food stamps, sdli\*\*/

array rcvff2 (J) P2090014 P2090024 P2090034 P2090044; /\*\*food stamps, int\*\*/

array rcvfw1 (J) PS2350014 PS2350024 PS2350034 PS2350044;/\*\*wic, sdli\*\*/

array rcvfw2 (J) P2350014 P2350024 P2350034 P2350044; /\*\*wic, int\*\*/

array rcvfa1 (J) PS3592014 PS3592024 PS3592034 PS3592044;/\*\*afdc, sdli\*\*/

array rcvfa2 (J) P1830014 P1830024 P1830034 P1830044; /\*\*afdc, int\*\*/

array rcvfo1 (J) PS3250014 PS3250024 PS3250034 PS3250044;/\*\*other, sdli\*\*/

array rcvfo2 (J) P3250014 P3250024 P3250034 P3250044; /\*\*other, int\*\*/

/\*\* PERSON receiving amount, other person \*\*/

array rcvof1 (J) PS2090015 PS2090025 PS2090035 PS2090045;/\*\*food stamps, sdli\*\*/

array rcvof2 (J) P2090015 P2090025 P2090035 P2090045; /\*\*food stamps, int\*\*/

array rcvow1 (J) PS2350015 PS2350025 PS2350035 PS2350045;/\*\*wic, sdli\*\*/

array rcvow2 (J) P2350015 P2350025 P2350035 P2350045; /\*\*wic, int\*\*/

array rcvoa1 (J) PS3592015 PS3592025 PS3592035 PS3592045;/\*\*afdc, sdli\*\*/

array rcvoa2 (J) P1830015 P1830025 P1830035 P1830045; /\*\*afdc, int\*\*/

array rcvoo1 (J) PS3250015 PS3250025 PS3250035 PS3250045;/\*\*other, sdli\*\*/

array rcvoo2 (J) P3250015 P3250025 P3250035 P3250045; /\*\*other, int\*\*/

array whof1 (J) whof11-whof14;

array whof2 (J) whof21-whof24;

array whow1 (J) whow11-whow14;

array whow2 (J) whow21-whow24;

array whoa1 (J) whoa11-whoa14;

array whoa2 (J) whoa21-whoa24;

array whoo1 (J) whoo11-who14;

array whoo2 (J) whoo21-who24;



```
/** PERSON was receiving at DLI, chk **/  
array dlif1 (J) PS19063A1 PS19063A2 PS19063A3 PS19063A4; /** food stamps **/  
array dliw1 (J) PS21900A1 PS21900A2 PS21900A3 PS21900A4; /** wic **/  
array dlial (J) PS357901 PS357902 PS357903 PS357904; /** afdc **/  
array dlio1 (J) PS30900A1 PS30900A2 PS30900A3 PS30900A4; /** other **/
```

```
** these ARRAYs encompass all programs for the ARRAYs;  
array ys (K) ysa1 ysa2 ysw1 ysw2 ysf1 ysf2 yso1 yso2;  
array ms (K) msa1 msa2 msw1 msw2 msf1 msf2 msol1 msol2;  
array ye (K) yea1 yea2 yew1 yew2 yef1 yef2 yeol1 yeol2;  
array me (K) meal1 meal2 mew1 mew2 mef1 mef2 meol1 meol2;  
array cur (K) cura1 cura2 curw1 curw2 curf1 curf2 curol1 curol2;  
array eflag (K) eflaga1 eflaga2 eflagw1 eflagw2 eflagf1 eflagf2 eflago1 eflago2;  
array aflag (K) aflaga1 aflaga2 aflagw1 aflagw2 aflagf1 aflagf2 aflago1 aflago2;  
array yms (K) ymsa1 ymsa2 ymsw1 ymsw2 ymsf1 ymsf2 ymsol1 ymsol2;  
array wks (K) wksa1 wksa2 wksw1 wksw2 wkswf1 wkswf2 wkswol1 wkswol2;  
array cwks (K) PS35740 P16200 PS21430 P21500 PS16410 P18900 PS36120 P30500;  
array yme (K) ymeal1 ymeal2 ymew1 ymew2 ymef1 ymef2 ymeol1 ymeol2;  
array mos (K) mosal1 mosal2 mosw1 mosw2 mosf1 mosf2 mosol1 mosol2;  
array csm (K) csmal1 csmal2 csmw1 csmw2 csmf1 csmf2 csmol1 csmol2;  
array cem (K) cemal1 cemal2 cemw1 cemw2 cemf1 cemf2 cemol1 cemol2;  
array rcvr (K) rcvra1 rcvra2 rcvrw1 rcvrw2 rcvrf1 rcvrf2 rcvro1 rcvro2;  
array rcvs (K) rcvsa1 rcvsa2 rcvsw1 rcvsw2 rcvsf1 rcvsf2 rcvsol1 rcvsol2;  
array revc (K) revca1 revca2 revcw1 revcw2 revcf1 revcf2 revcol1 revcol2;  
array revf (K) revfa1 revfa2 revfw1 revfw2 revff1 revff2 revfo1 revfo2;  
array rcvo (K) rcvoa1 rcvoa2 rcvow1 rcvow2 rcvof1 rcvof2 rcvool1 rcvool2;  
array who (K) whoa1 whoa2 whow1 whow2 whof1 whof2 whool1 whool2;  
array dli (K) dlial1 dlial2 dliw1 dliw2 dlif1 dlif2 dlio1 dlio2;  
array inc (K) incal1 incal2 incw1 incw2 incf1 incf2 incol1 incol2;
```

```
array agefla1 (J) agefla11-agefla14;  
array agefla2 (J) agefla21-agefla24;  
array ageflw1 (J) ageflw11-ageflw14;  
array ageflw2 (J) ageflw21-ageflw24;  
array ageflf1 (J) ageflf11-ageflf14;  
array ageflf2 (J) ageflf21-ageflf24;  
array ageflo1 (J) ageflo11-ageflo14;  
array ageflo2 (J) ageflo21-ageflo24;
```

```
array agefl (K) agefla1 agefla2 ageflw1 ageflw2 ageflf1 ageflf2 ageflo1 ageflo2;
```

```
/** HANDEDIT (2): These respondents reported valid start dates, currently receiving and NO stop  
periods for more than a month. HOWEVER they were asked if they received again. To this they gave a  
fresh set of start and stop dates which sometimes overlapped the previous loop, sometimes not. So we are  
ignoring ALL other loops other than loop 1. **/
```

```
/* These respondents reported 2 loops */
```

```
if pubid in (354, 1054, 1067, 1203, 1212, 1846, 2094, 2438, 2447, 2760, 2795, 3057, 3167, 3992, 4124,  
4184, 4343, 5039, 5349, 5381, 5846, 5942, 6071, 6469, 6611, 6622, 7035, 8125, 8301, 8392, 8581,  
8965) then do;
```

```
do J=2 to 4;
```

```
    ysw2=-4; msw2=-4; yew2=-4; mew2=-4; incw2=-4; incw2=-4; wksw2=-4;
```

```

    rcvrw2=-4; rcvsw2=-4; rcvcw2=-4; rcvfw2=-4; rcvow2=-4; curw2=-4;
end;
PROB_QUEX_w=2;
end;
/* These respondents reported more than 2 loops */
if pubid in (274, 470, 594, 1191, 2827, 3777, 4504, 6749, 6754, 6821, 8497) then do;
  do J=2 to 4;
    ysw2=-4; msw2=-4; yew2=-4; mew2=-4; incw2=-4; eincw2=-4; wksw2=-4;
    rcvrw2=-4; rcvsw2=-4; rcvcw2=-4; rcvfw2=-4; rcvow2=-4; curw2=-4;
  end;
  PROB_QUEX_w=3;
end;

```

/\*\* This portion of the SAS program defines the start date and end dates. If the respondent reports still receiving, the interview date is used as the temporary end date for the last loop reported. In the next survey round, the respondent will be asked if he or she is still receiving and, if not, a permanent end date equivalent to the interview date of the previous round will be assigned. Users will be able to tell which method was used by looking at the following participation flag variable created during the program.

The categories are the following:

1=respondent reported participation dates

2=start month imputed

3=start month and year imputed

4=stop month imputed

5=stop month and year imputed

6=start and stop dates imputed

7=error in data due to round 5 questionnaire problem or inconsistent response. Both results in our not sure whether the respondents received in these months or not.

Questionnaire Error: These respondents reported valid start and stop dates but were never asked the follow-up question - if they received again.

Conflicting Response: These respondents reported currently receiving, a valid start date and a valid stop date that is different from the current interview date but in the follow-up question as to whether they received again, they answered NO. /\*\*/

```

do K=1 to 8;
  do J=1 to 4;
    if ys>0 and ms>0 then yms=(ys*100)+ms; /* set start date if both start month and year
are valid */
    if dli=1 then do;
      yms=dliym; /* set start date to date last int if receiving then */
      if cwks=1 then do;
        yme=iym;
        NOGAPS=1;
      end;
    end;
    if cur=1 then yme=iym; /* set end date to current int date if receiving now */
    if ye>0 and me>0 then yme=(ye*100)+me; /* set end date if both end month and year are valid */
    if yme>iym then yme=iym; /* if end month greater than int month, constrain by int
month */
    if yms>0 and yme ge dliym then eflag=1;
    if wks ge 0 and (ys<0 OR ms<0 OR ye<0 OR me<0) then do;

```

```

        mos=floor(wks/4.3);                                /* determine months from estimated weeks
variable */
        end;

        agefl=0;
        if ys>0 and ys<1990 then agefl=1;

/** 1. If start year is known and month is unknown */

/** 1.a' If weeks are known and currently receiving, then count backwards by the number of weeks from
the interview date. If the number of weeks falls short of the start year, the start month is December of
that year. If the number of weeks is past the start year, then the start month is January of that year. */

        if ys>0 and -3 le ms le -1 then do;
            if (wks ge 0 and cwks=1) then do;
                yme=iy;
                if mos le YINT_M then yms=iy-mos;
                if (YINT_M ) le mos le (YINT_M+ 12) then yms=iy- 100-(mos- 12);
                if (YINT_M+ 12) le mos le (YINT_M+ 24) then yms=iy- 200-(mos- 24);
                if (YINT_M+ 24) le mos le (YINT_M+ 36) then yms=iy- 300-(mos- 36);
                if (YINT_M+ 36) le mos le (YINT_M+ 48) then yms=iy- 400-(mos- 48);
                if (YINT_M+ 48) le mos le (YINT_M+ 60) then yms=iy- 500-(mos- 60);
                if (YINT_M+ 60) le mos le (YINT_M+ 72) then yms=iy- 600-(mos- 72);
                if (YINT_M+ 72) le mos le (YINT_M+ 84) then yms=iy- 700-(mos- 84);
                if (YINT_M+ 84) le mos le (YINT_M+ 96) then yms=iy- 800-(mos- 96);
                if (YINT_M+ 96) le mos le (YINT_M+108) then yms=iy- 900-(mos-108);
                if (YINT_M+108) le mos le (YINT_M+120) then yms=iy-1000-(mos-120);
                if (YINT_M+120) le mos le (YINT_M+132) then yms=iy-1100-(mos-132);
                if yms<((ys*100)+01) then yms=((ys*100)+01);
                if yms<dliym then yms=dliym;
                if yms>((ys*100)+12) then yms=((ys*100)+12);
            end;

/** Weeks missing and continuously receiving - set end date to current interview date and start
date to December of start year */
            else if (-3 le wks lt 0 and cwks=1) then do;
                yme=iy;
                yms=((ys*100)+12);
            end;

/** Weeks missing and not continuously receiving - set end date to December and start date to
January of start year */
            else if (-3 le wks lt 0 and cwks=0) then do;
                yms=((ys*100)+01);
                if yme=. then yme=((ys*100)+12);
                if yme>iy then yme=iy;
            end;

/** 1.a" If weeks are known and not currently receiving, then count forward by the number of weeks
from January of the start year. If the count exceeds the interview date then stop counting
at the interview date. */

```

```

else if (wks ge 0 and cwks=0) then do;
  yms=((ys*100)+01);
  if yms<dliym then yms=dliym;
  if 0 le mos le 12 then yme=(yms+mos);
  if wks>0 and cwks=0 and 13 le mos le 24 then yme=((ys+ 1)*100)+mos- 12;
  if wks>0 and cwks=0 and 25 le mos le 36 then yme=((ys+ 2)*100)+mos- 24;
  if wks>0 and cwks=0 and 37 le mos le 48 then yme=((ys+ 3)*100)+mos- 36;
  if wks>0 and cwks=0 and 49 le mos le 60 then yme=((ys+ 4)*100)+mos- 48;
  if wks>0 and cwks=0 and 61 le mos le 72 then yme=((ys+ 5)*100)+mos- 60;
  if wks>0 and cwks=0 and 73 le mos le 84 then yme=((ys+ 6)*100)+mos- 72;
  if wks>0 and cwks=0 and 85 le mos le 96 then yme=((ys+ 7)*100)+mos- 84;
  if wks>0 and cwks=0 and 97 le mos le 108 then yme=((ys+ 8)*100)+mos- 96;
  if wks>0 and cwks=0 and 109 le mos le 120 then yme=((ys+ 9)*100)+mos-108;
  if wks>0 and cwks=0 and 121 le mos le 132 then yme=((ys+10)*100)+mos-120;
  if wks>0 and cwks=0 and 133 le mos le 144 then yme=((ys+11)*100)+mos-132;
  if yme>iym then yme=iym;
end;

```

```

if yme>0 and cwks=0 then yms=((ys*100)+01);
if yms<dliym then yms=dliym;
eflag=2;
end;

```

/\*\* 2. If start year is unknown but weeks are known then count back from interview date if currently receiving. If not currently receiving, then count back from interview date to find the most recent year the respondent could have begun receiving and receive for that number of months: then count forward the number of months from January of that year. \*\*/

```

if -3 le ys le -1 then do;
  if (wks ge 0 and cwks=1) then do;
    if 0 le mos le YINT_M then yms=iym-mos;
    if (YINT_M ) le mos le (YINT_M+ 12) then yms=iym- 100-(mos- 12);
    if (YINT_M+ 12) le mos le (YINT_M+ 24) then yms=iym- 200-(mos- 24);
    if (YINT_M+ 24) le mos le (YINT_M+ 36) then yms=iym- 300-(mos- 36);
    if (YINT_M+ 36) le mos le (YINT_M+ 48) then yms=iym- 400-(mos- 48);
    if (YINT_M+ 48) le mos le (YINT_M+ 60) then yms=iym- 500-(mos- 60);
    if (YINT_M+ 60) le mos le (YINT_M+ 72) then yms=iym- 600-(mos- 72);
    if (YINT_M+ 72) le mos le (YINT_M+ 84) then yms=iym- 700-(mos- 84);
    if (YINT_M+ 84) le mos le (YINT_M+ 96) then yms=iym- 800-(mos- 96);
    if (YINT_M+ 96) le mos le (YINT_M+108) then yms=iym- 900-(mos-108);
    if (YINT_M+108) le mos le (YINT_M+120) then yms=iym-1000-(mos-120);
    if (YINT_M+120) le mos le (YINT_M+132) then yms=iym-1100-(mos-132);
    yme=iym;
    if yms<dliym then yms=dliym;
  end;
end;

```

```

if (-3 le wks lt 0 and cwks=1) then do;
  yms=dliym;
  yme=iym;
end;

```

```

if (wks ge 0 and cwks=0) then do;
  if 0 le mos le YINT_M then do;
    yms=(YINT_Y*100)+01;
    yme=(YINT_Y*100)+01+mos;
  end;
  if (YINT_M+01) le mos le (YINT_M+12) then do;
    yms=((YINT_Y-1)*100)+01;
    if (YINT_M+01) le mos le 12 then yme=((YINT_Y-1)*100)+01+mos;
    if 13 le mos le (YINT_M+12) then yme=(YINT_Y*100)+01+mos;
  end;
  if (YINT_M+13) le mos le (YINT_M+24) then do;
    yms=((YINT_Y-2)*100)+01;
    if (YINT_M+13) le mos le 24 then yme=((YINT_Y-1)*100)+01+mos;
    if 25 le mos le (YINT_M+24) then yme=(YINT_Y*100)+01+mos;
  end;
  if yme>iym then yme=iym;
end;
eflag=3;
end;

```

```

if yme>0 and yms=. then do;
  yms=dliym;
  if j=1 then eflag=3;
  else if j>1 then eflag=7;
end;

```

/\*\* 3. If stop year is known and weeks are known and not currently receiving, but stop month is not known, then count forward from start year. If the number of months falls short of the stop year, then use January of the end year as the stop date; if the number of months exceeds the stop year, then end the array in the December of the stop year. If the stop year is equal to the interview year and the stop month exceeds the interview month, then stop at the interview date. If currently receiving, use interview date as the stop date. \*\*/

```

if yms>0 and ye>0 and -2 le me le -1 then do;
  if wks ge 0 then do;
    if 01 le ((yms- ROUND (yms,100))+mos) le 12 then yme=yms+mos;
    if 13 le ((yms- ROUND (yms,100))+mos) le 24 then yme=yms+ 100+mos- 12;
    if 25 le ((yms- ROUND (yms,100))+mos) le 36 then yme=yms+ 200+mos- 24;
    if 37 le ((yms- ROUND (yms,100))+mos) le 48 then yme=yms+ 300+mos- 36;
    if 49 le ((yms- ROUND (yms,100))+mos) le 60 then yme=yms+ 400+mos- 48;
    if 61 le ((yms- ROUND (yms,100))+mos) le 72 then yme=yms+ 500+mos- 60;
    if 73 le ((yms- ROUND (yms,100))+mos) le 84 then yme=yms+ 600+mos- 72;
    if 85 le ((yms- ROUND (yms,100))+mos) le 96 then yme=yms+ 700+mos- 84;
    if 97 le ((yms- ROUND (yms,100))+mos) le 108 then yme=yms+ 800+mos- 96;
    if 109 le ((yms- ROUND (yms,100))+mos) le 120 then yme=yms+ 900+mos-108;
    if 121 le ((yms- ROUND (yms,100))+mos) le 132 then yme=yms+1000+mos-120;
    if 133 le ((yms- ROUND (yms,100))+mos) le 144 then yme=yms+1100+mos-132;
    if yme>((ye*100)+12) then yme=((ye*100)+12);
    if yme<((ye*100)+01) then yme=((ye*100)+01);
    if yme>iym then yme=iym;
  end;
end;

```

```
end;
```

```
if -3 le wks lt 0 then yme=((ye*100)+12);  
if yme>iym then yme=iym;
```

```
eflag=4;  
end;
```

```
/** 4. If stop year is unknown, and weeks are known **/
```

```
if -3 le ye le -1 then do;  
  if yms>0 and wks ge 0 then do;  
    if 01 le ((yms- ROUND (yms,100))+mos) le 12 then yme=yms+mos;  
    if 12 lt ((yms- ROUND (yms,100))+mos) le 24 then yme=yms+100+mos-12;  
    if 24 lt ((yms- ROUND (yms,100))+mos) le 36 then yme=yms+200+mos-24;  
    if yme>iym then yme=iym;  
  end;
```

```
/** 5. If stop year is unknown, and weeks are unknown **/
```

```
if yms>0 and -3 le wks lt 0 then yme=(ROUND (yms,100))+12;  
eflag=5;  
end;
```

```
/** 6. If the start and stop years are unknown and the weeks are unknown, use current and last  
interview date **/
```

```
if -3 le ys le -1 and -3 le ye le -1 and cwks=0 and -3 le wks lt 0 then do;  
  yme=iym;  
  yms=dliym;  
  if j=1 then eflag=6;  
  else if j>1 then eflag=7;  
end;
```

/\*\* This portion of the program creates a variable that determines who receives afdc in the household. It collapses the answers to 8 categories. The coding is the following -

```
1=respondent only  
2=spouse/partner only  
3=child only  
4=respondent and spouse/partner  
5=respondent and child  
6=spouse/partner and child  
7=respondent and spouse/partner and child  
8=other
```

The first 7 categories may include an 'other' person as captured by response categories 4 (other relatives in household) and 5 (other (specIFy)) in the original question. If only 'other' person is listed as receiving, then the 8th answer category is used in the created variable. \*\*\*/

```
if rcvr=1 and rcvs=0 and rcvc=0 then who=01;  
if rcvr=0 and rcvs=1 and rcvc=0 then who=02;  
if rcvr=0 and rcvs=0 and rcvc=1 then who=03;
```

```

if rcvr=1 and rcvs=1 and rcvc=0 then who=04;
if rcvr=1 and rcvs=0 and rcvc=1 then who=05;
if rcvr=0 and rcvs=1 and rcvc=1 then who=06;
if rcvr=1 and rcvs=1 and rcvc=1 then who=07;
if -3 le rcvr lt 0 then who=rcvr;
if rcvr=0 and rcvs=0 and rcvc=0 and (rcvf=1 OR rcvo=1) then who=08;

```

/\*\* This portion of the program uses the category reported by the respondent to create an estimated amount. The estimated amount is the midpoint rounded down. Note that the 12th category lists \$1251 as the amount. This amount was chosen since the category is unbounded - the number represents one dollar above the lower bound.\*\*/

```
/** food stamps **/
```

```

if -3 le incf1 le -1 and 1 le eincf1 le 10 then incf1=(eincf1*100)-50;
if -3 le incf1 le -1 and eincf1=11 then incf1=1125;
if -3 le incf1 le -1 and eincf1=12 then incf1=1251;
if -3 le incf2 le -1 and 1 le eincf2 le 10 then incf2=(eincf2*100)-50;
if -3 le incf2 le -1 and eincf2=11 then incf2=1125;
if -3 le incf2 le -1 and eincf2=12 then incf2=1251;

```

```
/** afdc **/
```

```

if -3 le inca1 le -1 and 1 le einca1 le 10 then inca1=(einca1*100)-50;
if -3 le inca1 le -1 and einca1=11 then inca1=1125;
if -3 le inca1 le -1 and einca1=12 then inca1=1251;
if -3 le inca2 le -1 and 1 le einca2 le 10 then inca2=(einca2*100)-50;
if -3 le inca2 le -1 and einca2=11 then inca2=1125;
if -3 le inca2 le -1 and einca2=12 then inca2=1251;

```

```
/** other **/
```

```

if -3 le inco1 le -1 and 1 le einco1 le 10 then inco1=(einco1*100)-50;
if -3 le inco1 le -1 and einco1=11 then inco1=1125;
if -3 le inco1 le -1 and einco1=12 then inco1=1251;
if -3 le inco2 le -1 and 1 le einco2 le 10 then inco2=(einco2*100)-50;
if -3 le inco2 le -1 and einco2=11 then inco2=1125;
if -3 le inco2 le -1 and einco2=12 then inco2=1251;

```

```
/** wic **/
```

```

if -3 le incw1 le -1 and 1 le eincw1 le 5 then incw1=(eincw1*20)-10;
if -3 le incw1 le -1 and eincw1=6 then incw1=101;
if -3 le incw2 le -1 and 1 le eincw2 le 5 then incw2=(eincw2*20)-10;
if -3 le incw2 le -1 and eincw2=6 then incw2=101;

```

```
if inc>-4 then aflag=0;
```

```
csm=(ROUND(yms,100)-198000)*.12+(yms-ROUND(yms,100));
```

```
cem=(ROUND(yme,100)-198000)*.12+(yme-ROUND(yme,100));
```

```
cdli=(ROUND(dliym,100)-198000)*.12+(dliym-ROUND(dliym,100))-1; /* added to zero out
```

```
people who back report */
```

```
end;
```

```
/* [e3] */
```

```
end;
```

```
/* [e2] */
```

/\*\* HANDEDIT (3): These respondents reported receiving at dli, not currently receiving and never stopping for more than 1 month. We conjectured that they may have stopped receiving less than a month before the date of current interview and so we should give them a stop date of 1 month prior to the current interview date. We also give them a FLAG... FLAG\_MAYBE.

This program uses the variable ps32700A1 to indicate whether the respondent was currently receiving or not. This variable is however not a response variable but a check item that we compute - it takes on the value 1 if EITHER the respondent says currently receiving and never stopped receiving OR was receiving and never stopped receiving. The respondent above satisfied the second criterion and so had a value of 1 for ps32700A1. So in these cases the program interpreted the respondents as currently receiving and gave them an end date equal to the current interview date. However as said earlier these respondents reported not currently receiving. So giving them an end date a month before the current interview date may be a better idea. (Feb 14, 2002) \*\*/

```
if pubd in (1935, 3930, 4974, 5286, 5548, 5847, 6011, 6589, 6918, 7104, 7541, 8006, 8689) then do;
  FLAG_MAYBE=1;
  oldcemo1=cemo1;
  cemo1=doicm-1;
  if iym=200201 then do;
    ymeo1=200112;
  end;
  if iym ne 200201 then do;
    ymeo1=iym-1;
  end;
end;
```

/\*\* This part of the program creates the variables that indicate whether the respondent received a particular government program for each month and the monthly amount received. \*\*/

```
do K=1 to 8;                                     /** [b4] **/
  do J=1 to 4;                                     /** [b5] **/
    C=0;
    do L=1 to 269;
      C=C+1;
      if 0 le L le doicm then do;
        if csm le C le cem then do;
          if 1 le K le 2 then do;
            a=eflag;
            ahhm=who;
            aamt=inc;
            aodd=0;
            if aamt>1000 then do;
              aodd=aamt;
            end;
            if a>0 and (aamt=. OR ahhm=.) then do;
              ack=ack+1;
            end;
            if aamt>0 and (a=0 OR ahhm=.) then do;
              ackm=ackm+1;
            end;
          end;
        end;
      end;
    end;
  end;
```



```

    if ahhm>0 and (a=0 OR aamt=.) then do;
        ackh=ackh+1;
    end;
    if aamt=0 then dummya=1;
end;
if 3 le K le 4 then do;
    w=eflag;
    whhm=who;
    wamt=inc;
    woddd=0;
    if wamt>1000 then do;
        woddd=wamt;
    end;
    if w>0 and (wamt=. OR whhm=.) then do;
        wck=wck+1;
    end;
    if wamt>0 and (w=0 OR whhm=.) then do;
        wckm=wckm+1;
    end;
    if whhm>0 and (w=0 OR wamt=.) then do;
        wckh=wckh+1;
    end;
    if wamt=0 then dummyw=1;
end;
if 5 le K le 6 then do;
    f=eflag;
    fhhm=who;
    famt=inc;
    fodd=0;
    if famt>1000 then do;
        fodd=famt;
    end;
    if f>0 and (famt=. OR fhhm=.) then do;
        fck=fck+1;
    end;
    if famt>0 and (f=0 OR fhhm=.) then do;
        fckm=fckm+1;
    end;
    if fhhm>0 and (f=0 OR famt=.) then do;
        fckh=fckh+1;
    end;
    if fhhm=. and f>0 and famt ne . then do;
        fhhm=-3;
        fckneg=1;
    end;
    if famt=0 then dummyf=1;
end;
if 7 le K le 8 then do;
    o=eflag;
    ohhm=who;

```

round 3\*/

/\*this fixes a problem in



```
if pubid=5556 then do; PROB_QUEX_w=1; w267=7; end;
if pubid=7106 then do; PROB_QUEX_w=1; do L=252 to 264; w=7; end; end;
if pubid=7131 then do; PROB_QUEX_w=1; do L=263 to 267; w=7; end; end;
if pubid=7715 then do; PROB_QUEX_f=1; do L=228 to 268; f=7; end; end;
if pubid=8283 then do; PROB_QUEX_f=1; do L=261 to 264; f=7; end; end;
if pubid=8561 then do; PROB_QUEX_w=1; do L=263 to 265; w=7; end; end;
if pubid=7715 then do; PROB_QUEX_w=1; do L=238 to 268; w=7; end; end;
```

/\*\* HANDEDIT (4b): These respondents reported valid start and stop dates, not currently receiving BUT were never asked if they received again. So we code the months receiving as 7 and the monthly amounts receiving as -3 from the month after the valid stop date to the current interview date. We leave out cases where the imputed stop date is the current interview date. \*\*/

```
if pubid= 16 then do; PROB_QUEX_f=1; f267=7; end;
if pubid= 451 then do; PROB_QUEX_f=1; do L=224 to 264; f=7; end; end;
if pubid=2502 then do; PROB_QUEX_f=1; do L=265 to 266; f=7; end; end;
if pubid=4111 then do; PROB_QUEX_f=1; do L=265 to 266; f=7; end; end;
if pubid=5586 then do; PROB_QUEX_f=1; do L=264 to 265; f=7; end; end;
if pubid=5633 then do; PROB_QUEX_f=1; do L=259 to 263; f=7; end; end;
if pubid=5893 then do; PROB_QUEX_f=1; do L=261 to 264; f=7; end; end;
if pubid=6465 then do; PROB_QUEX_f=1; do L=258 to 264; f=7; end; end;
if pubid=6895 then do; PROB_QUEX_f=1; do L=223 to 267; f=7; end; end;
if pubid=7909 then do; PROB_QUEX_f=1; do L=255 to 265; f=7; end; end;
```

```
if pubid= 16 then do; PROB_QUEX_o=1; do L=232 to 267; o=7; end; end;
if pubid= 872 then do; PROB_QUEX_o=1; do L=254 to 265; o=7; end; end;
if pubid= 982 then do; PROB_QUEX_o=1; do L=247 to 268; o=7; end; end;
if pubid=1559 then do; PROB_QUEX_o=1; do L=258 to 263; o=7; end; end;
if pubid=4674 then do; PROB_QUEX_o=1; do L=259 to 264; o=7; end; end;
if pubid=5606 then do; PROB_QUEX_o=1; do L=258 to 269; o=7; end; end;
if pubid=6463 then do; PROB_QUEX_o=1; do L=259 to 265; o=7; end; end;
if pubid=6856 then do; PROB_QUEX_o=1; do L=255 to 265; o=7; end; end;
if pubid=6895 then do; PROB_QUEX_o=1; do L=224 to 268; o=7; end; end;
if pubid=7739 then do; PROB_QUEX_o=1; do L=253 to 264; o=7; end; end;
if pubid=8059 then do; PROB_QUEX_o=1; do L=257 to 264; o=7; end; end;
if pubid=8808 then do; PROB_QUEX_o=1; do L=260 to 266; o=7; end; end;
```

```
if pubid=1368 then do; PROB_QUEX_w=1; do L=256 to 265; w=7; end; end;
if pubid=1415 then do; PROB_QUEX_w=1; do L=252 to 265; w=7; end; end;
if pubid=1583 then do; PROB_QUEX_w=1; do L=263 to 264; w=7; end; end;
if pubid=1939 then do; PROB_QUEX_w=1; do L=235 to 266; w=7; end; end;
if pubid=1950 then do; PROB_QUEX_w=1; do L=262 to 266; w=7; end; end;
if pubid=3211 then do; PROB_QUEX_w=1; do L=263 to 265; w=7; end; end;
if pubid=3477 then do; PROB_QUEX_w=1; do L=267 to 268; w=7; end; end;
if pubid=3742 then do; PROB_QUEX_w=1; do L=259 to 267; w=7; end; end;
if pubid=4111 then do; PROB_QUEX_w=1; do L=254 to 266; w=7; end; end;
if pubid=4171 then do; PROB_QUEX_w=1; do L=264 to 266; w=7; end; end;
if pubid=4211 then do; PROB_QUEX_w=1; do L=259 to 267; w=7; end; end;
if pubid=4865 then do; PROB_QUEX_w=1; do L=265 to 266; w=7; end; end;
```

```

if pubid=5049 then do; PROB_QUEX_w=1; do L=255 to 265; w=7; end; end;
if pubid=5314 then do; PROB_QUEX_w=1; do L=253 to 265; w=7; end; end;
if pubid=6466 then do; PROB_QUEX_w=1; do L=241 to 265; w=7; end; end;
if pubid=6492 then do; PROB_QUEX_w=1; do L=259 to 266; w=7; end; end;
if pubid=6748 then do; PROB_QUEX_w=1; do L=235 to 265; w=7; end; end;
if pubid=6895 then do; PROB_QUEX_w=1; do L=224 to 267; w=7; end; end;
if pubid=7233 then do; PROB_QUEX_w=1; w265=7; end;
if pubid=7719 then do; PROB_QUEX_w=1; do L=261 to 264; w=7; end; end;
if pubid=7918 then do; PROB_QUEX_w=1; do L=257 to 265; w=7; end; end;
if pubid=8121 then do; PROB_QUEX_w=1; do L=258 to 265; w=7; end; end;
if pubid=8256 then do; PROB_QUEX_w=1; do L=255 to 265; w=7; end; end;
if pubid=8397 then do; PROB_QUEX_w=1; w265=7; end;
if pubid=8493 then do; PROB_QUEX_w=1; w265=7; end;
if pubid=8968 then do; PROB_QUEX_w=1; do L=237 to 267; w=7; end; end;
if pubid=9018 then do; PROB_QUEX_w=1; do L=263 to 264; w=7; end; end;

```

/\*\* HANDEDIT (5): These respondents reported valid start and stop dates, currently receiving which are different from

code the current interview dates BUT when asked if they received again, they denied it. So we the months receiving as 7 from the month after the valid stop date to the current interview date.

We leave out cases where the imputed stop date is the current interview date. \*\*/

```

if pubid= 143 then do; do L=261 to 263; w=7; PROB_DATA_w=1; end; end;
if pubid=1511 then do; do L=259 to 264; f=7; PROB_DATA_f=1; end; end;
if pubid=1875 then do; do L=265 to 266; f=7; PROB_DATA_f=1; end; end;
if pubid=5467 then do; do L=258 to 263; f=7; PROB_DATA_f=1; end; end;
if pubid=8037 then do; do L=262 to 264; f=7; PROB_DATA_f=1; end; end;
if pubid=8325 then do; f265=7; PROB_DATA_f=1; end;

```

/\*\* HANDEDIT (6a): These respondents reported overlapping loops with different amounts for each loop. So we

are coding those monthly amounts as -3 and months receiving as 9 and code these later to -3. \*\*/

```

if pubid=6460 then do;
do L=259 to 265; famt=-3; f=9; end; OVERLAP_f=1;
end;

```

```

if pubid=519 then do;
do L=253 to 264; wamt=-3; w=9; end; OVERLAP_w=1;
end;

```

/\*\* (new r5) If start date<current int date and R went through the section previously, zero months post dli. \*\*/

```

do K=1 to 8;          /** [b6] **/
do J=1 to 4;         /** [b7] **/
C=0;
do L=1 to 269;
C=C+1;

```



```

omnths=0;
wmnths=0;

if gprg_ever>-4 OR P1210>-4 OR p1210=-5 then do;
  do L=1 to 269;
    if a>0 and aamt>1000 and aodd>0 then do; /* calculate total no. months to divide the problematic
amts over */
      amnths=amnths+1;
    end;
    if f>0 and famt>1000 and fodd>0 then do;
      fmnths=fmnths+1;
    end;
    if o>0 and oamt>1000 and oodd>0 then do;
      omnths=omnths+1;
    end;
    if w>0 and wamt>1000 and wodd>0 then do;
      wmnths=wmnths+1;
    end;
  end;
end;
end;

```

```

if gprg_ever>-4 OR P1210>-4 OR p1210=-5 then do; /** [b8] **/

```

```

  do L=1 to 269; /** [b9] **/
    if aamt>1000 and amnths>0 and a>0 then do; /*divide the amount by the number of
months*/
      aamt=ROUND(aamt/amnths, 1);
    end;
    if famt>1000 and fmnths>0 and f>0 then do;
      famt=ROUND(famt/fmnths, 1);
    end;
    if oamt>1000 and omnths>0 and o>0 then do;
      oamt=ROUND(oamt/omnths, 1);
    end;
    if wamt>1000 and wmnths>0 and w>0 then do;
      wamt=ROUND(wamt/wmnths, 1);
    end;

```

```

if 1 le L le 12 then do;
  if M80 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M80=M80+1;
  end;
  if a>1 then M80=-3;
  if w>1 then M80=-3;
  if f>1 then M80=-3;
  if o>1 then M80=-3;
  array allamt80 aamt wamt famt oamt;
  do over allamt80;
    if -3 le allamt80 le -1 then do;
      nn80=nn80+1;
      amt80=allamt80;
    end;
  end;
end;

```

```
end;  
if allamt80 ge 0 then do;  
  na80=na80+1;  
  if na80=1 and nn80=0 then amt80=allamt80;  
  if na80>1 and nn80=0 then amt80=amt80+allamt80;  
  if M80=-3 then amt80=-3;  
end;  
end;  
end;
```

```
if 13 le L le 24 then do;  
  if M81 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;  
    M81=M81+1;  
  end;  
  if a>1 then M81=-3;  
  if w>1 then M81=-3;  
  if f>1 then M81=-3;  
  if o>1 then M81=-3;  
  array allamt81 aamt wamt famt oamt;  
  do over allamt81;  
    if -3 le allamt81 le -1 then do;  
      nn81=nn81+1;  
      amt81=allamt81;  
    end;  
    if allamt81 ge 0 then do;  
      na81=na81+1;  
      if na81=1 and nn81=0 then amt81=allamt81;  
      if na81>1 and nn81=0 then amt81=amt81+allamt81;  
      if M81=-3 then amt81=-3;  
    end;  
  end;  
end;  
end;
```

```
if 25 le L le 36 then do;  
  if M82 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;  
    M82=M82+1;  
  end;  
  if a>1 then M82=-3;  
  if w>1 then M82=-3;  
  if f>1 then M82=-3;  
  if o>1 then M82=-3;  
  array allamt82 aamt wamt famt oamt;  
  do over allamt82;  
    if -3 le allamt82 le -1 then do;  
      nn82=nn82+1;  
      amt82=allamt82;  
    end;  
    if allamt82 ge 0 then do;  
      na82=na82+1;  
      if na82=1 and nn82=0 then amt82=allamt82;  
      if na82>1 and nn82=0 then amt82=amt82+allamt82;  
    end;  
  end;  
end;
```

```

    if M82=-3 then amt82=-3;
    end;
end;
end;
if 37 le L le 48 then do;
    if M83 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
        M83=M83+1;
    end;
    if a>1 then M83=-3;
    if w>1 then M83=-3;
    if f>1 then M83=-3;
    if o>1 then M83=-3;
    array allamt83 aamt wamt famt oamt;
    do over allamt83;
        if -3 le allamt83 le -1 then do;
            nn83=nn83+1;
            amt83=allamt83;
        end;
        if allamt83 ge 0 then do;
            na83=na83+1;
            if na83=1 and nn83=0 then amt83=allamt83;
            if na83>1 and nn83=0 then amt83=amt83+allamt83;
            if M83=-3 then amt83=-3;
        end;
    end;
end;
end;

```

```

if 49 le L le 60 then do;
    if M84 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
        M84=M84+1;
    end;
    if a>1 then M84=-3;
    if w>1 then M84=-3;
    if f>1 then M84=-3;
    if o>1 then M84=-3;
    array allamt84 aamt wamt famt oamt;
    do over allamt84;
        if -3 le allamt84 le -1 then do;
            nn84=nn84+1;
            amt84=allamt84;
        end;
        if allamt84 ge 0 then do;
            na84=na84+1;
            if na84=1 and nn84=0 then amt84=allamt84;
            if na84>1 and nn84=0 then amt84=amt84+allamt84;
            if M84=-3 then amt84=-3;
        end;
    end;
end;
end;

```



```

if 61 le L le 72 then do;
  if M85 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M85=M85+1;
  end;
  if a>1 then M85=-3;
  if w>1 then M85=-3;
  if f>1 then M85=-3;
  if o>1 then M85=-3;
  array allamt85 aamt wamt famt oamt;
  do over allamt85;
    if -3 le allamt85 le -1 then do;
      nn85=nn85+1;
      amt85=allamt85;
    end;
    if allamt85 ge 0 then do;
      na85=na85+1;
      if na85=1 and nn85=0 then amt85=allamt85;
      if na85>1 and nn85=0 then amt85=amt85+allamt85;
      if M85=-3 then amt85=-3;
    end;
  end;
end;

```

```

if 73 le L le 84 then do;
  if M86 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M86=M86+1;
  end;
  if a>1 then M86=-3;
  if w>1 then M86=-3;
  if f>1 then M86=-3;
  if o>1 then M86=-3;
  array allamt86 aamt wamt famt oamt;
  do over allamt86;
    if -3 le allamt86 le -1 then do;
      nn86=nn86+1;
      amt86=allamt86;
    end;
    if allamt86 ge 0 then do;
      na86=na86+1;
      if na86=1 and nn86=0 then amt86=allamt86;
      if na86>1 and nn86=0 then amt86=amt86+allamt86;
      if M86=-3 then amt86=-3;
    end;
  end;
end;

```

```

if 85 le L le 96 then do;
  if M87 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M87=M87+1;
  end;
  if a>1 then M87=-3;

```

```

if w>1 then M87=-3;
if f>1 then M87=-3;
if o>1 then M87=-3;
array allamt87 aamt wamt famt oamt;
do over allamt87;
  if -3 le allamt87 le -1 then do;
    nn87=nn87+1;
    amt87=allamt87;
  end;
  if allamt87 ge 0 then do;
    na87=na87+1;
    if na87=1 and nn87=0 then amt87=allamt87;
    if na87>1 and nn87=0 then amt87=amt87+allamt87;
    if M87=-3 then amt87=-3;
  end;
end;
end;
end;

```

```

if 97 le L le 108 then do;
  if M88 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M88=M88+1;
  end;
  if a>1 then M88=-3;
  if w>1 then M88=-3;
  if f>1 then M88=-3;
  if o>1 then M88=-3;
  array allamt88 aamt wamt famt oamt;
  do over allamt88;
    if -3 le allamt88 le -1 then do;
      nn88=nn88+1;
      amt88=allamt88;
    end;
    if allamt88 ge 0 then do;
      na88=na88+1;
      if na88=1 and nn88=0 then amt88=allamt88;
      if na88>1 and nn88=0 then amt88=amt88+allamt88;
      if M88=-3 then amt88=-3;
    end;
  end;
end;
end;
end;

```

```

if 109 le L le 120 then do;
  if M89 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M89=M89+1;
  end;
  if a>1 then M89=-3;
  if w>1 then M89=-3;
  if f>1 then M89=-3;
  if o>1 then M89=-3;
  array allamt89 aamt wamt famt oamt;
  do over allamt89;

```

```

if -3 le allamt89 le -1 then do;
  nn89=nn89+1;
  amt89=allamt89;
end;
if allamt89 ge 0 then do;
  na89=na89+1;
  if na89=1 and nn89=0 then amt89=allamt89;
  if na89>1 and nn89=0 then amt89=amt89+allamt89;
  if M89=-3 then amt89=-3;
end;
end;
end;

```

```

if 121 le L le 132 then do;
  if M90 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M90=M90+1;
  end;
  if a>1 then M90=-3;
  if w>1 then M90=-3;
  if f>1 then M90=-3;
  if o>1 then M90=-3;
  array allamt90 aamt wamt famt oamt;
  do over allamt90;
    if -3 le allamt90 le -1 then do;
      nn90=nn90+1;
      amt90=allamt90;
    end;
    if allamt90 ge 0 then do;
      na90=na90+1;
      if na90=1 and nn90=0 then amt90=allamt90;
      if na90>1 and nn90=0 then amt90=amt90+allamt90;
      if M90=-3 then amt90=-3;
    end;
  end;
end;
end;

```

```

if 133 le L le 144 then do;
  if M91 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M91=M91+1;
  end;
  if a>1 then M91=-3;
  if w>1 then M91=-3;
  if f>1 then M91=-3;
  if o>1 then M91=-3;
  array allamt91 aamt wamt famt oamt;
  do over allamt91;
    if -3 le allamt91 le -1 then do;
      nn91=nn91+1;
      amt91=allamt91;
    end;
    if allamt91 ge 0 then do;

```

```
    na91=na91+1;
    if na91=1 and nn91=0 then amt91=allamt91;
    if na91>1 and nn91=0 then amt91=amt91+allamt91;
    if M91=-3 then amt91=-3;
end;
end;
end;
```

```
if 145 le L le 156 then do;
  if M92 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M92=M92+1;
  end;
  if a>1 then M92=-3;
  if w>1 then M92=-3;
  if f>1 then M92=-3;
  if o>1 then M92=-3;
  array allamt92 aamt wamt famt oamt;
  do over allamt92;
    if -3 le allamt92 le -1 then do;
      nn92=nn92+1;
      amt92=allamt92;
    end;
    if allamt92 ge 0 then do;
      na92=na92+1;
      if na92=1 and nn92=0 then amt92=allamt92;
      if na92>1 and nn92=0 then amt92=amt92+allamt92;
      if M92=-3 then amt92=-3;
    end;
  end;
end;
end;
```

```
if 157 le L le 168 then do;
  if M93 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M93=M93+1;
  end;
  if a>1 then M93=-3;
  if w>1 then M93=-3;
  if f>1 then M93=-3;
  if o>1 then M93=-3;
  array allamt93 aamt wamt famt oamt;
  do over allamt93;
    if -3 le allamt93 le -1 then do;
      nn93=nn93+1;
      amt93=allamt93;
    end;
    if allamt93 ge 0 then do;
      na93=na93+1;
      if na93=1 and nn93=0 then amt93=allamt93;
      if na93>1 and nn93=0 then amt93=amt93+allamt93;
      if M93=-3 then amt93=-3;
    end;
  end;
end;
```

```

end;
end;

if 169 le L le 180 then do;
  if M94 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M94=M94+1;
  end;
  if a>1 then M94=-3;
  if w>1 then M94=-3;
  if f>1 then M94=-3;
  if o>1 then M94=-3;
  array allamt94 aamt wamt famt oamt;
  do over allamt94;
    if -3 le allamt94 le -1 then do;
      nn94=nn94+1;
      amt94=allamt94;
    end;
    if allamt94 ge 0 then do;
      na94=na94+1;
      if na94=1 and nn94=0 then amt94=allamt94;
      if na94>1 and nn94=0 then amt94=amt94+allamt94;
      if M94=-3 then amt94=-3;
    end;
  end;
end;
end;

```

```

if 181 le L le 192 then do;
  if M95 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M95=M95+1;
  end;
  if a>1 then M95=-3;
  if w>1 then M95=-3;
  if f>1 then M95=-3;
  if o>1 then M95=-3;
  array allamt95 aamt wamt famt oamt;
  do over allamt95;
    if -3 le allamt95 le -1 then do;
      nn95=nn95+1;
      amt95=allamt95;
    end;
    if allamt95 ge 0 then do;
      na95=na95+1;
      if na95=1 and nn95=0 then amt95=allamt95;
      if na95>1 and nn95=0 then amt95=amt95+allamt95;
      if M95=-3 then amt95=-3;
    end;
  end;
end;
end;

```

```

if 193 le L le 204 then do;
  if M96 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;

```

```

M96=M96+1;
end;
if a>1 then M96=-3;
if w>1 then M96=-3;
if f>1 then M96=-3;
if o>1 then M96=-3;
array allamt96 aamt wamt famt oamt;
do over allamt96;
  if -3 le allamt96 le -1 then do;
    nn96=nn96+1;
    amt96=allamt96;
  end;
  if allamt96 ge 0 then do;
    na96=na96+1;
    if na96=1 and nn96=0 then amt96=allamt96;
    if na96>1 and nn96=0 then amt96=amt96+allamt96;
    if M96=-3 then amt96=-3;
  end;
end;
end;

```

```

if 205 le L le 216 then do;
  if M97 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M97=M97+1;
  end;
  if a>1 then M97=-3;
  if w>1 then M97=-3;
  if f>1 then M97=-3;
  if o>1 then M97=-3;
  array allamt97 aamt wamt famt oamt;
  do over allamt97;
    if -3 le allamt97 le -1 then do;
      nn97=nn97+1;
      amt97=allamt97;
    end;
    if allamt97 ge 0 then do;
      na97=na97+1;
      if na97=1 and nn97=0 then amt97=allamt97;
      if na97>1 and nn97=0 then amt97=amt97+allamt97;
      if M97=-3 then amt97=-3;
    end;
  end;
end;

```

```

if 217 le L le 228 then do;
  if M98 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M98=M98+1;
  end;
  if a>1 then M98=-3;
  if w>1 then M98=-3;
  if f>1 then M98=-3;

```

```

if o>1 then M98=-3;
array allamt98 aamt wamt famt oamt;
do over allamt98;
  if -3 le allamt98 le -1 then do;
    nn98=nn98+1;
    amt98=allamt98;
  end;
  if allamt98 ge 0 then do;
    na98=na98+1;
    if na98=1 and nn98=0 then amt98=allamt98;
    if na98>1 and nn98=0 then amt98=amt98+allamt98;
    if M98=-3 then amt98=-3;
  end;
end;
end;

```

```

if 229 le L le 240 then do;
  if M99 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M99=M99+1;
  end;
  if a>1 then M99=-3;
  if w>1 then M99=-3;
  if f>1 then M99=-3;
  if o>1 then M99=-3;
array allamt99 aamt wamt famt oamt;
do over allamt99;
  if -3 le allamt99 le -1 then do;
    nn99=nn99+1;
    amt99=allamt99;
  end;
  if allamt99 ge 0 then do;
    na99=na99+1;
    if na99=1 and nn99=0 then amt99=allamt99;
    if na99>1 and nn99=0 then amt99=amt99+allamt99;
    if M99=-3 then amt99=-3;
  end;
end;
end;

```

```

if 241 le L le 252 then do;
  if M00 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    M00=M00+1;
  end;
  if a>1 then M00=-3;
  if w>1 then M00=-3;
  if f>1 then M00=-3;
  if o>1 then M00=-3;
array allamt00 aamt wamt famt oamt;
do over allamt00;
  if -3 le allamt00 le -1 then do;
    nn00=nn00+1;
  end;
end;

```

```
    amt00=allamt00;
end;
if allamt00 ge 0 then do;
    na00=na00+1;
    if na00=1 and nn00=0 then amt00=allamt00;
    if na00>1 and nn00=0 then amt00=amt00+allamt00;
    if M00=-3 then amt00=-3;
end;
end;
end;
```

```
if 253 le L le 264 then do;
    if M01 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
        M01=M01+1;
    end;
    if a>1 then M01=-3;
    if w>1 then M01=-3;
    if f>1 then M01=-3;
    if o>1 then M01=-3;
    array allamt01 aamt wamt famt oamt;
    do over allamt01;
        if -3 le allamt01 le -1 then do;
            nn01=nn01+1;
            amt01=allamt01;
        end;
        if allamt01 ge 0 then do;
            na01=na01+1;
            if na01=1 and nn01=0 then amt01=allamt01;
            if na01>1 and nn01=0 then amt01=amt01+allamt01;
            if M01=-3 then amt01=-3;
        end;
    end;
end;
end;
```

```
if 265 le L le 276 then do;
    if M02 ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
        M02=M02+1;
    end;
    if a>1 then M02=-3;
    if w>1 then M02=-3;
    if f>1 then M02=-3;
    if o>1 then M02=-3;
    array allamt02 aamt wamt famt oamt;
    do over allamt02;
        if -3 le allamt02 le -1 then do;
            nn02=nn02+1;
            amt02=allamt02;
        end;
        if allamt02 ge 0 then do;
            na02=na02+1;
            if na02=1 and nn02=0 then amt02=allamt02;
        end;
    end;
end;
```



```
    if na02>1 and nn02=0 then amt02=amt02+allamt02;
    if M02=-3 then amt02=-3;
end;
end;
end;
```

```
***total months - part 2;
if 1 le L le 269 then do;
  if TTLM ge 0 and (a>0 OR w>0 OR f>0 OR o>0) then do;
    TTLM=TTLM+1;
  end;
  if a>1 then TTLM=-3;
  if w>1 then TTLM=-3;
  if f>1 then TTLM=-3;
  if o>1 then TTLM=-3;
end;
```

```
end;          /** [e9] **/
end;          /** [e8] **/
```

```
/** 3. end **/
```

```
do I=1 to 23;
  if gprg_ever=. then gprg_ever=0;
  if gtotal=-4 then gtotal=0;
  if gatotal=-4 then gatotal=0;

  if M ge 0 and gtotal ge 0 then do;
    gtotal=M+gtotal;
  end;
  if M<0 OR gtotal<0 then do;
    gtotal=-3;
  end;
  if AMT ge 0 and gatotal ge 0 then do;
    gatotal=AMT+gatotal;
  end;
  if AMT<0 OR gatotal<0 OR (gtotal<0 and gtotal ne -4) then do;
    gatotal=-3;
  end;
  if AMT=0 and oldgatotal=-4 and gtotal=0 then do;
    gatotal=-4;
  end;
end;
```

```
***total months, part 2;
```

```
if TTLM ge 0 and gprg_ever ge 0 then do;
  TTLM=TTLM+gprg_ever;
end;
if TTLM<0 OR gprg_ever<0 then do;
  TTLM=-3;
```

end;

/\*\* HANDEDIT (6b): Follow-up to (6a) \*\*/

```
if pubid=6460 then do;
  do L=259 to 265;
    f=-3;
  end;
end;
```

```
if pubid=519 then do;
  do L=253 to 264;
    w=-3;
  end;
end;
```

/\*\* Taking care of non-interviews and valid skips \*\*/

```
do I=1 to 23;
  if p1210=-4 then do;
    gtotal=-4;
    gatotal=-4;
    TTLM=-4;
  end;
  if p1210=-5 then do;
    gtotal=-5;
    gatotal=-5;
    TTLM=-5;
  end;
end;
```

/\*\* To change all invalid skips to -3 \*\*/

```
array vars TTLM gtotal80-gtotal99 gtotal00 gtotal01
          gatotal80-gatotal99 gatotal00 gatotal01;
```

```
do over vars;
  if vars=-2 then vars=-3;
  if vars=-1 then vars=-3;
end;
```

endsas;

---