

Emp_begin.sas

A number of the created employment variables use the same program as input. The program in this section is referred to throughout the employment variables. For example, to create the "Weeks Worked at Employee Job #x during 19xx" variables, survey staff first run the program titled "emp_begin.sas" and then run the program included in the weeks worked section of this appendix. The variables and code for "emp_begin.sas" are provided below.

This program calculates total weeks worked at each job for each respondent. It converts start and stop dates for jobs and within-job gaps to continuous week numbers, subtracts within-job gaps, and finally counts the total weeks worked. The variables listed here are those employed in any program that uses emp_begin.sas as input; they may not be used in this initial program.

Name in Program	Question Name on CD	Name in Program	Question Name on CD
r1int_d	YINF_900_D_1997	EGDY2_3	YEMP-102700.02.03~D
r1int_m	YINF_900_M_1997	EGMO2_3	YEMP-102700.02.03~M
r1int_y	YINF_900_Y_1997	EGYR2_3	YEMP-102700.02.03~Y
birthd	KEY_BDATE_D_1997	EGDY2_4	YEMP-102700.02.04~D
birthm	KEY_BDATE_M_1997	EGMO2_4	YEMP-102700.02.04~M
birthy	KEY_BDATE_Y_1997	EGYR2_4	YEMP-102700.02.04~Y
r2int_d	SYMBOL_CURDATE_D_1998	EGDY2_5	YEMP-102700.02.05~D
r2int_m	SYMBOL_CURDATE_M_1998	EGMO2_5	YEMP-102700.02.05~M
r2int_y	SYMBOL_CURDATE_Y_1998	EGYR2_5	YEMP-102700.02.05~Y
r3int_d	SYMBOL_CURDATE_D_1999	EGDY2_6	YEMP-102700.02.06~D
r3int_m	SYMBOL_CURDATE_M_1999	EGMO2_6	YEMP-102700.02.06~M
r3int_y	SYMBOL_CURDATE_Y_1999	EGYR2_6	YEMP-102700.02.06~Y
r4int_d	SYMBOL_CURDATE_D_2000	EGDY2_7	YEMP-102700.02.07~D
r4int_m	SYMBOL_CURDATE_M_2000	EGMO2_7	YEMP-102700.02.07~M
r4int_y	SYMBOL_CURDATE_Y_2000	EGYR2_7	YEMP-102700.02.07~Y
r5int_d	SYMBOL_CURDATE_D_2001	EGDY3_1	YEMP-102700.03.01~D
r5int_m	SYMBOL_CURDATE_M_2001	EGMO3_1	YEMP-102700.03.01~M
r5int_y	SYMBOL_CURDATE_Y_2001	EGYR3_1	YEMP-102700.03.01~Y
r6int_d	SYMBOL_CURDATE_D_2002	EGDY3_2	YEMP-102700.03.02~D
r6int_m	SYMBOL_CURDATE_M_2002	EGMO3_2	YEMP-102700.03.02~M
r6int_y	SYMBOL_CURDATE_Y_2002	EGYR3_2	YEMP-102700.03.02~Y
r6int_d	YINTDATE~D	EGDY3_3	YEMP-102700.03.03~D
r6int_m	YINTDATE~M	EGMO3_3	YEMP-102700.03.03~M
r6int_y	YINTDATE~Y	EGYR3_3	YEMP-102700.03.03~Y
e200	YEMP-200A	EGDY4_1	YEMP-102700.04.01~D
E2390101-	YEMP-23901.01-.11	EGMO4_1	YEMP-102700.04.01~M
E2390111			
E2450101-	YEMP-24501.01-.08	EGYR4_1	YEMP-102700.04.01~Y
E2450108			
E3440201-	YEMP-34402.01-.06	EGDY4_2	YEMP-102700.04.02~D
E3440206			
E3440301	YEMP-34403.01	EGMO4_2	YEMP-102700.04.02~M
E3440302	YEMP-34403.02	EGYR4_2	YEMP-102700.04.02~Y
E3440303	YEMP-34403.03	EGDY5_1	YEMP-102700.05.01~D
E3440304	YEMP-34403.04	EGMO5_1	YEMP-102700.05.01~M
E3442801	YEMP-34428.01	EGYR5_1	YEMP-102700.05.01~Y
E3442802	YEMP-34428.02	EGDY5_2	YEMP-102700.05.02~D
E3442803	YEMP-34428.03	EGMO5_2	YEMP-102700.05.02~M
E3442804	YEMP-34428.04	EGYR5_2	YEMP-102700.05.02~Y
E2260401	YEMP-22604.01~000002	NUMGAP1	YEMP-103200.01

E2260401	YEMP-22604.01~000003	NUMGAP2	YEMP-103200.02
E2260402	YEMP-22604.02~000002	NUMGAP3	YEMP-103200.03
E2260402	YEMP-22604.02~000003	NUMGAP4	YEMP-103200.04
E2260403	YEMP-22604.03~000002	NUMGAP5	YEMP-103200.05
E2260403	YEMP-22604.03~000003	pubid	PUBID
E2260404	YEMP-22604.04~000002	MILFLAG01-	YEMP_MILFLAG.01-.11
		MILFLAG11	
E2260404	YEMP-22604.04~000003	SELF01-	YEMP_SELFEMP.01-.11
		SELF11	
E2261001	YEMP-22610.01.02	STARD01-	YEMP_STARTDATE.01~D-.11~D
		STARD11	
E2261002	YEMP-22610.02.02	STARM01-	YEMP_STARTDATE.01~M-.11~M
		STARM11	
E2261101	YEMP-22611.01.03	STARY01-	YEMP_STARTDATE.01~Y-.11~Y
		STARY11	
E2261102	YEMP-22611.02.03	STOPD01-	YEMP_STOPDATE.01~D-.11~D
		STOPD11	
E3790201-	YEMP-37902.01-.10	STOPM01	YEMP_STOPDATE.01~M-.11~M
E3790210		STOPM11	
E3790401-	YEMP-37904.01-.07	STOPY01	YEMP_STOPDATE.01~Y-.11~Y
E3790407		STOPY11	
E3800001-	YEMP-38000.01-.11	UID01-UID11	YEMP_UID.01-.11
E3800011			
E38000B01-	YEMP-38000B.01-.11	sGAP01	YEMP-105902.01
E38000B11			
E38000F01-	YEMP-38000F.01-.10	sGAP02	YEMP-105902.02
E38000F10			
E3800201	YEMP-38002.01	sGAP03	YEMP-105902.03
E3800202	YEMP-38002.02	sGAP04	YEMP-105902.04
E3800203	YEMP-38002.03	sGAP05	YEMP-105902.05
E3800204	YEMP-38002.04	sBGDY1_1	YEMP-105912.01.01~D
E3800205	YEMP-38002.05	sBGMO1_1	YEMP-105912.01.01~M
E3810201	YEMP-38102.01	sBGYR1_1	YEMP-105912.01.01~Y
E3810202	YEMP-38102.02	sBGDY2_1	YEMP-105912.02.01~D
E3810203	YEMP-38102.03	sBGMO2_1	YEMP-105912.02.01~M
E3810204	YEMP-38102.04	sBGYR2_1	YEMP-105912.02.01~Y
E3810205	YEMP-38102.05	sBGDY3_1	YEMP-105912.03.01~D
E3810301	YEMP-38103.01	sBGMO3_1	YEMP-105912.03.01~M
E3810302	YEMP-38103.02	sBGYR3_1	YEMP-105912.03.01~Y
E3810303	YEMP-38103.03	sBGDY4_1	YEMP-105912.04.01~D
E3810304	YEMP-38103.04	sBGMO4_1	YEMP-105912.04.01~M
E3810305	YEMP-38103.05	sBGYR4_1	YEMP-105912.04.01~Y
E3810501	YEMP-38105.01	sBGDY1_2	YEMP-105914.01.02~D
E3810502	YEMP-38105.02	sBGMO1_2	YEMP-105914.01.02~M
E3810503	YEMP-38105.03	sBGYR1_2	YEMP-105914.01.02~Y
E3810504	YEMP-38105.04	sBGDY2_2	YEMP-105914.02.02~D
E5990101-	YEMP-59901.01-.07	sBGMO2_2	YEMP-105914.02.02~M
E5990107			
E8800001-	YEMP-88000.01-.07	sBGYR2_2	YEMP-105914.02.02~Y
E8800007			
E8850101-	YEMP-88501.01-.07	sBGDY3_2	YEMP-105914.03.02~D
E8850107			
E9840201-	YEMP-98402.01-.06	sBGMO3_2	YEMP-105914.03.02~M
E9840206			
E9840301	YEMP-98403.01	sBGYR3_2	YEMP-105914.03.02~Y
E9840302	YEMP-98403.02	sBGDY4_2	YEMP-105914.04.02~D
E9840303	YEMP-98403.03	sBGMO4_2	YEMP-105914.04.02~M
E9842901	YEMP-98429.01	sBGYR4_2	YEMP-105914.04.02~Y
E9842902	YEMP-98429.02	sEGDY1_1	YEMP-105916.01.01~D
E9842903	YEMP-98429.03	sEGMO1_1	YEMP-105916.01.01~M
E1002250	YEMP-100225.01~000002	sEGYR1_1	YEMP-105916.01.01~Y

E1002250	YEMP-100225.01~000003	sEGDY1_2	YEMP-105916.01.02~D
E1002250	YEMP-100225.02~000002	sEGMO1_2	YEMP-105916.01.02~M
E1002250	YEMP-100225.02~000003	sEGYR1_2	YEMP-105916.01.02~Y
E1002250	YEMP-100225.03~000002	sEGDY2_1	YEMP-105916.02.01~D
E1002250	YEMP-100225.03~000003	sEGMO2_1	YEMP-105916.02.01~M
E1002250	YEMP-100225.04~000002	sEGYR2_1	YEMP-105916.02.01~Y
E1002250	YEMP-100225.04~000003	sEGDY2_2	YEMP-105916.02.02~D
E1002310	YEMP-100231.01.02	sEGMO2_2	YEMP-105916.02.02~M
E1002310	YEMP-100231.02.02	sEGYR2_2	YEMP-105916.02.02~Y
E1002310	YEMP-100231.03.02	sEGDY3_1	YEMP-105916.03.01~D
E1002310	YEMP-100231.04.02	sEGMO3_1	YEMP-105916.03.01~M
E1002320	YEMP-100232.01.03	sEGYR3_1	YEMP-105916.03.01~Y
GAP01-	YEMP-101500.01-.11	sEGDY3_2	YEMP-105916.03.02~D
GAP11			
BGDY1_1	YEMP-102200.01.01~D	sEGMO3_2	YEMP-105916.03.02~M
BGMO1_1	YEMP-102200.01.01~M	sEGYR3_2	YEMP-105916.03.02~Y
BGYR1_1	YEMP-102200.01.01~Y	sEGDY4_1	YEMP-105916.04.01~D
BGDY2_1	YEMP-102200.02.01~D	sEGMO4_1	YEMP-105916.04.01~M
BGMO2_1	YEMP-102200.02.01~M	sEGYR4_1	YEMP-105916.04.01~Y
BGYR2_1	YEMP-102200.02.01~Y	sEGDY4_2	YEMP-105916.04.02~D
BGDY3_1	YEMP-102200.03.01~D	sEGMO4_2	YEMP-105916.04.02~M
BGMO3_1	YEMP-102200.03.01~M	sEGYR4_2	YEMP-105916.04.02~Y
BGYR3_1	YEMP-102200.03.01~Y	sNUMGAP1	YEMP-105920.01
BGDY4_1	YEMP-102200.04.01~D	sNUMGAP2	YEMP-105920.02
BGMO4_1	YEMP-102200.04.01~M	sNUMGAP3	YEMP-105920.03
BGYR4_1	YEMP-102200.04.01~Y	sNUMGAP4	YEMP-105920.04
BGDY5_1	YEMP-102200.05.01~D		
BGMO5_1	YEMP-102200.05.01~M	r1hrs80-	CV_HOURS_WK_YR_80_1997-
		r1hrs98	CV_HOURS_WK_YR_98_1997
BGYR5_1	YEMP-102200.05.01~Y	r2hrs80 r2hrs99	CV_HOURS_WK_YR_80_1998-
			CV_HOURS_WK_YR_99_1998
BGDY1_2	YEMP-102400.01.02~D	r3hrs80-	CV_HOURS_WK_YR_80_1999-
		r3hrs00	CV_HOURS_WK_YR_00_1999
BGMO1_2	YEMP-102400.01.02~M	r4ehrs80-	CV_HOURS_WK_YR_ET_80_2000-
		4ehrs01	CV_HOURS_WK_YR_ET_01_2000
BGYR1_2	YEMP-102400.01.02~Y	r4hrs80-	CV_HOURS_WK_YR_ALL_80_2000-
		r4hrs01	CV_HOURS_WK_YR_ALL_01_2000
BGDY1_3	YEMP-102400.01.03~D	r4shrs98-	CV_HOURS_WK_YR_SE_98_2000-
		r4shrs01	CV_HOURS_WK_YR_SE_01_2000
BGMO1_3	YEMP-102400.01.03~M	r5ehrs80-	CV_HOURS_WK_YR_ET_80_2001-
		r5ehrs02	CV_HOURS_WK_YR_ET_02_2001
BGYR1_3	YEMP-102400.01.03~Y	r5hrs80-	CV_HOURS_WK_YR_ALL_80_2001-
		r5hrs02	CV_HOURS_WK_YR_ALL_02_2001
BGDY1_4	YEMP-102400.01.04~D	r5shrs98-	CV_HOURS_WK_YR_SE_98_2001-
		r5shrs02	CV_HOURS_WK_YR_SE_02_2001
BGMO1_4	YEMP-102400.01.04~M	r1wks14	CV_WKSWK_EVER_1997
BGYR1_4	YEMP-102400.01.04~Y	r2wks14	CV_WKSWK_EVER_1998
BGDY1_5	YEMP-102400.01.05~D	r3wks14	CV_WKSWK_EVER_1999
BGMO1_5	YEMP-102400.01.05~M	r4emp14	CV_WKSWK_TEEN_2000
BGYR1_5	YEMP-102400.01.05~Y	r4emp20	CV_WKSWK_ADULT_ET_2000
BGDY2_2	YEMP-102400.02.02~D	r4wks20	CV_WKSWK_ADULT_ALL_2000
BGMO2_2	YEMP-102400.02.02~M	r5emp14	CV_WKSWK_TEEN_2001
BGYR2_2	YEMP-102400.02.02~Y	r5emp20	CV_WKSWK_ADULT_ET_2001
BGDY2_3	YEMP-102400.02.03~D	r5wks20	CV_WKSWK_ADULT_ALL_2001
BGMO2_3	YEMP-102400.02.03~M	r1UID1-	YEMP_UID_01_1997-
		r1UID7	YEMP_UID_07_1997
BGYR2_3	YEMP-102400.02.03~Y	r1WKS801-	CV_WKSWK_JOB_YR_01_80_1997-
		r1WKS981	CV_WKSWK_JOB_YR_01_98_1997
BGDY2_4	YEMP-102400.02.04~D	r1WKS802-	CV_WKSWK_JOB_YR_02_80_1997-
		r1WKS982	CV_WKSWK_JOB_YR_02_98_1997
BGMO2_4	YEMP-102400.02.04~M	r1WKS803-	CV_WKSWK_JOB_YR_03_80_1997-

BGYR2_4	YEMP-102400.02.04~Y	r1WKS983	CV_WKSWK_JOB_YR_03_98_1997
BGDY2_5	YEMP-102400.02.05~D	r1WKS804-	CV_WKSWK_JOB_YR_04_80_1997-
BGMO2_5	YEMP-102400.02.05~M	r1WKS984	CV_WKSWK_JOB_YR_04_98_1997
BGYR2_5	YEMP-102400.02.05~Y	r1WKS805-	CV_WKSWK_JOB_YR_05_80_1997-
BGDY2_6	YEMP-102400.02.06~D	r1WKS985	CV_WKSWK_JOB_YR_05_98_1997
BGMO2_6	YEMP-102400.02.06~M	r1WKS806-	CV_WKSWK_JOB_YR_06_80_1997-
BGYR2_6	YEMP-102400.02.06~Y	r1WKS986	CV_WKSWK_JOB_YR_06_98_1997
BGDY2_7	YEMP-102400.02.07~D	r1WKS807-	CV_WKSWK_JOB_YR_07_80_1997-
BGMO2_7	YEMP-102400.02.07~M	r1WKS987	CV_WKSWK_JOB_YR_07_98_1997
BGYR2_7	YEMP-102400.02.07~Y	r2UID1-	YEMP_UID_01_1998-
BGDY3_2	YEMP-102400.03.02~D	r2UID9	YEMP_UID_09_1998
BGMO3_2	YEMP-102400.03.02~M	r2WKS801-	CV_WKSWK_JOB_YR_01_80_1998-
BGYR3_2	YEMP-102400.03.02~Y	r2WKS991	CV_WKSWK_JOB_YR_01_99_1998
BGDY3_3	YEMP-102400.03.03~D	r2WKS802-	CV_WKSWK_JOB_YR_02_80_1998-
BGMO3_3	YEMP-102400.03.03~M	r2WKS992	CV_WKSWK_JOB_YR_02_99_1998
BGYR3_3	YEMP-102400.03.03~Y	r2WKS803-	CV_WKSWK_JOB_YR_03_80_1998-
BGDY4_2	YEMP-102400.04.02~D	r2WKS993	CV_WKSWK_JOB_YR_03_99_1998
BGMO4_2	YEMP-102400.04.02~M	r2WKS804-	CV_WKSWK_JOB_YR_04_80_1998-
BGYR4_2	YEMP-102400.04.02~Y	r2WKS994	CV_WKSWK_JOB_YR_04_99_1998
BGDY5_2	YEMP-102400.05.02~D	r2WKS805-	CV_WKSWK_JOB_YR_05_80_1998-
BGMO5_2	YEMP-102400.05.02~M	r2WKS995	CV_WKSWK_JOB_YR_05_99_1998
BGYR5_2	YEMP-102400.05.02~Y	r2WKS806-	CV_WKSWK_JOB_YR_06_80_1998-
EGDY1_1	YEMP-102700.01.01~D	r2WKS996	CV_WKSWK_JOB_YR_06_99_1998
EGMO1_1	YEMP-102700.01.01~M	r2WKS807-	CV_WKSWK_JOB_YR_07_80_1998-
EGYR1_1	YEMP-102700.01.01~Y	r2WKS997	CV_WKSWK_JOB_YR_07_99_1998
EGDY1_2	YEMP-102700.01.02~D	r2WKS808-	CV_WKSWK_JOB_YR_08_80_1998-
EGMO1_2	YEMP-102700.01.02~M	r2WKS998	CV_WKSWK_JOB_YR_08_99_1998
EGYR1_2	YEMP-102700.01.02~Y	r2WKS809-	CV_WKSWK_JOB_YR_09_80_1998-
EGDY1_3	YEMP-102700.01.03~D	r2WKS999	CV_WKSWK_JOB_YR_09_99_1998
EGMO1_3	YEMP-102700.01.03~M	r3UID1-	YEMP_UID_01_1999-
		r3UID9	YEMP_UID_09_1999
		r3WKS801-	CV_WKSWK_JOB_YR_01_80_1999-
		r3WKS001	CV_WKSWK_JOB_YR_01_00_1999
		r3WKS802-	CV_WKSWK_JOB_YR_02_80_1999-
		r3WKS002	CV_WKSWK_JOB_YR_02_00_1999
		r3WKS803-	CV_WKSWK_JOB_YR_03_80_1999-
		r3WKS003	CV_WKSWK_JOB_YR_03_00_1999
		r3WKS804-	CV_WKSWK_JOB_YR_04_80_1999-
		r3WKS004	CV_WKSWK_JOB_YR_04_00_1999
		r3WKS805-	CV_WKSWK_JOB_YR_05_80_1999-
		r3WKS005	CV_WKSWK_JOB_YR_05_00_1999
		r3WKS806-	CV_WKSWK_JOB_YR_06_80_1999-
		r3WKS006	CV_WKSWK_JOB_YR_06_00_1999
		r3WKS807-	CV_WKSWK_JOB_YR_07_80_1999-
		r3WKS007	CV_WKSWK_JOB_YR_07_00_1999
		r3WKS808-	CV_WKSWK_JOB_YR_08_80_1999-
		r3WKS008	CV_WKSWK_JOB_YR_08_00_1999
		r3WKS809-	CV_WKSWK_JOB_YR_09_80_1999-
		r3WKS009	CV_WKSWK_JOB_YR_09_00_1999
		r4UID1-	YEMP_UID_01_2000-
		r4UID9	YEMP_UID_09_2000
		r4WKS801-	CV_WKSWK_JOB_YR_01_80_2000-
		r4WKS011	CV_WKSWK_JOB_YR_01_01_2000
		r4WKS802-	CV_WKSWK_JOB_YR_02_80_2000-
		r4WKS012	CV_WKSWK_JOB_YR_02_01_2000
		r4WKS803-	CV_WKSWK_JOB_YR_03_80_2000-
		r4WKS013	CV_WKSWK_JOB_YR_03_01_2000
		r4WKS804-	CV_WKSWK_JOB_YR_04_80_2000-
		r4WKS014	CV_WKSWK_JOB_YR_04_01_2000
		r4WKS805-	CV_WKSWK_JOB_YR_05_80_2000-
		r4WKS015	CV_WKSWK_JOB_YR_05_01_2000

EGYR1_3	YEMP-102700.01.03~Y	r4WKS806- r4WKS016	CV_WKSWK_JOB_YR_06_80_2000- CV_WKSWK_JOB_YR_06_01_2000
EGDY1_4	YEMP-102700.01.04~D	r4WKS807- r4WKS017	CV_WKSWK_JOB_YR_07_80_2000- CV_WKSWK_JOB_YR_07_01_2000
EGMO1_4	YEMP-102700.01.04~M	r4WKS808- r4WKS018	CV_WKSWK_JOB_YR_08_80_2000- CV_WKSWK_JOB_YR_08_01_2000
EGYR1_4	YEMP-102700.01.04~Y	r4WKS809- r4WKS019	CV_WKSWK_JOB_YR_09_80_2000- CV_WKSWK_JOB_YR_09_01_2000
EGDY1_5	YEMP-102700.01.05~D	r5UID1- r5UID8	YEMP_UID_01_2001- YEMP_UID_08_2001
EGMO1_5	YEMP-102700.01.05~M	r5WKS801- r5WKS021	CV_WKSWK_JOB_YR_01_80_2001- CV_WKSWK_JOB_YR_01_02_2001
EGYR1_5	YEMP-102700.01.05~Y	r5WKS802- r5WKS022	CV_WKSWK_JOB_YR_02_80_2001- CV_WKSWK_JOB_YR_02_02_2001
EGDY2_1	YEMP-102700.02.01~D	r5WKS803- r5WKS023	CV_WKSWK_JOB_YR_03_80_2001- CV_WKSWK_JOB_YR_03_02_2001
EGMO2_1	YEMP-102700.02.01~M	r5WKS804- r5WKS024	CV_WKSWK_JOB_YR_04_80_2001- CV_WKSWK_JOB_YR_04_02_2001
EGYR2_1	YEMP-102700.02.01~Y	r5WKS805- r5WKS025	CV_WKSWK_JOB_YR_05_80_2001- CV_WKSWK_JOB_YR_05_02_2001
EGDY2_2	YEMP-102700.02.02~D	r5WKS806- r5WKS026	CV_WKSWK_JOB_YR_06_80_2001- CV_WKSWK_JOB_YR_06_02_2001
EGMO2_2	YEMP-102700.02.02~M	r5WKS807- r5WKS027	CV_WKSWK_JOB_YR_07_80_2001- CV_WKSWK_JOB_YR_07_02_2001
EGYR2_2	YEMP-102700.02.02~Y	r5WKS808- r5WKS028	CV_WKSWK_JOB_YR_08_80_2001- CV_WKSWK_JOB_YR_08_02_2001

/* Section 1: Convert the birthday, 14th birthday, 18th birthday, 20th birthday and January 1st of 18th birthday year into a continuous week number. Week 1 is the first week in 1980. */

**** Calulate Age 14 year **;**

AGE14Yr=birthyr+14;

**** Convert Age 14 Birthdate to week number ** ;**

*****Convert age14 month and day to total days (bbdays);**

```

if birthmo>0 and birthdy>0 then do;
if birthmo=1 then bbdays=birthdy;
if birthmo=2 then bbdays=birthdy+31;
if birthmo=3 then bbdays=birthdy+59;
if birthmo=4 then bbdays=birthdy+90;
if birthmo=5 then bbdays=birthdy+120;
if birthmo=6 then bbdays=birthdy+151;
if birthmo=7 then bbdays=birthdy+181;
if birthmo=8 then bbdays=birthdy+212;
if birthmo=9 then bbdays=birthdy+243;
if birthmo=10 then bbdays=birthdy+273;
if birthmo=11 then bbdays=birthdy+304;
if birthmo=12 then bbdays=birthdy+334;
end;

```

***Account for leap years;

if age14yr=1980 or age14yr=1984 or age14yr=1988 or age14yr=1992 or age14yr=1996 or age18yr=2000 then do;

if birthmo>0 and birthdy>0 then do;
if birthmo=1 then bbdays=birthdy;
if birthmo=2 then bbdays=birthdy+31;
if birthmo=3 then bbdays=birthdy+60;
if birthmo=4 then bbdays=birthdy+91;
if birthmo=5 then bbdays=birthdy+121;
if birthmo=6 then bbdays=birthdy+152;
if birthmo=7 then bbdays=birthdy+182;
if birthmo=8 then bbdays=birthdy+213;
if birthmo=9 then bbdays=birthdy+244;
if birthmo=10 then bbdays=birthdy+274;
if birthmo=11 then bbdays=birthdy+305;
if birthmo=12 then bbdays=birthdy+335;
end;
end;

***Convert days into week numbers;

** Basic Formula used throughout program

** weekno=endweek{specific year}+ceil[(totdays+{# of days remaining in DEC})/7]

/* Default age 14 week = 9999 */

age14wk=9999;

if age14yr>0 and bbdays>0 then do;

if age14yr=1993 then do;

age14wk=678+ceil((bbdays+5)/7);

end;

if age14yr=1994 then do;

age14wk=730+ceil((bbdays+6)/7);

end;

if age14yr=1995 then do;

age14wk=783+ceil((bbdays)/7);

end;

if age14yr=1996 then do;

age14wk=835+ceil((bbdays+1)/7);

end;

if age14yr=1997 then do;

age14wk=887+ceil((bbdays+3)/7);

end;

if age14yr=1998 then do;

age14wk=939+ceil((bbdays+4)/7);

```
end;
if age14yr=1999 then do;
age14wk=991+ceil((bbdays+5)/7);
end;
if age14yr=2000 then do;
age14wk=1043+ceil((bbdays+6)/7);
end;
end;
```

```
** Calulate Age 18 year **;
```

```
AGE18YR=birthyr+18;
```

```
** Convert Age 18 Birthdate to week number **;
```

```
***Convert age18 month and day to total days (bbdays);
```

```
if birthmo>0 and birthdy>0 then do;
if birthmo=1 then bbdays=birthdy;
if birthmo=2 then bbdays=birthdy+31;
if birthmo=3 then bbdays=birthdy+59;
if birthmo=4 then bbdays=birthdy+90;
if birthmo=5 then bbdays=birthdy+120;
if birthmo=6 then bbdays=birthdy+151;
if birthmo=7 then bbdays=birthdy+181;
if birthmo=8 then bbdays=birthdy+212;
if birthmo=9 then bbdays=birthdy+243;
if birthmo=10 then bbdays=birthdy+273;
if birthmo=11 then bbdays=birthdy+304;
if birthmo=12 then bbdays=birthdy+334;
end;
```

```
***Account for leap years;
```

```
if age18yr=1980 or age18yr=1984 or age18yr=1988 or age18yr=1992 or age18yr=1996 or
age18yr=2000 then do;
if birthmo>0 and birthdy>0 then do;
if birthmo=1 then bbdays=birthdy;
if birthmo=2 then bbdays=birthdy+31;
if birthmo=3 then bbdays=birthdy+60;
if birthmo=4 then bbdays=birthdy+91;
if birthmo=5 then bbdays=birthdy+121;
if birthmo=6 then bbdays=birthdy+152;
if birthmo=7 then bbdays=birthdy+182;
if birthmo=8 then bbdays=birthdy+213;
if birthmo=9 then bbdays=birthdy+244;
if birthmo=10 then bbdays=birthdy+274;
if birthmo=11 then bbdays=birthdy+305;
if birthmo=12 then bbdays=birthdy+335;
end;
```

end;

***Convert days into week numbers;

/* Default age 18 week = 9999 */

age18wk=9999;

if age18yr>0 and bbdays>0 then do;

if age18yr=1997 then do;

age18wk=887+ceil((bbdays+3)/7);

ag18jan=888;

end;

if age18yr=1998 then do;

age18wk=939+ceil((bbdays+4)/7);

ag18jan=940;

end;

if age18yr=1999 then do;

age18wk=991+ceil((bbdays+5)/7);

ag18jan=992;

end;

if age18yr=2000 then do;

age18wk=1043+ceil((bbdays+6)/7);

ag18jan=1044;

end;

if age18yr=2001 then do;

age18wk=1096+ceil((bbdays+1)/7);

ag18jan=1097;

end;

if age18yr=2002 then do;

age18wk=1148+ceil((bbdays+2)/7);

ag18jan=1149;

end;

end;

** Calulate Age 20 week **;

AGE20YR=birthyr+20;

** Convert Age 20 Birthdate to week number **;

***Convert age20 month and day to total days (bbdays);

if birthmo>0 and birthdy>0 then do;

if birthmo=1 then bbdays=birthdy;

if birthmo=2 then bbdays=birthdy+31;

if birthmo=3 then bbdays=birthdy+59;

if birthmo=4 then bbdays=birthdy+90;

if birthmo=5 then bbdays=birthdy+120;

if birthmo=6 then bbdays=birthdy+151;

if birthmo=7 then bbdays=birthdy+181;

if birthmo=8 then bbdays=birthdy+212;


```
if birthmo=9 then bbdays=birthdy+243;
if birthmo=10 then bbdays=birthdy+273;
if birthmo=11 then bbdays=birthdy+304;
if birthmo=12 then bbdays=birthdy+334;
end;
```

```
***Account for leap years;
```

```
if age20yr=1980 or age20yr=1984 or age20yr=1988 or age20yr=1992 or age20yr=1996 or
age20yr=2000 or age20yr=2004 then do;
if birthmo>0 and birthdy>0 then do;
if birthmo=1 then bbdays=birthdy;
if birthmo=2 then bbdays=birthdy+31;
if birthmo=3 then bbdays=birthdy+60;
if birthmo=4 then bbdays=birthdy+91;
if birthmo=5 then bbdays=birthdy+121;
if birthmo=6 then bbdays=birthdy+152;
if birthmo=7 then bbdays=birthdy+182;
if birthmo=8 then bbdays=birthdy+213;
if birthmo=9 then bbdays=birthdy+244;
if birthmo=10 then bbdays=birthdy+274;
if birthmo=11 then bbdays=birthdy+305;
if birthmo=12 then bbdays=birthdy+335;
end;
end;
```

```
***Convert days into week numbers;
```

```
/* Default age 20 week = 9999 */
age20wk=9999;
```

```
if age20yr>0 and bbdays>0 then do;
if age20yr=1999 then do;
age20wk=991+ceil((bbdays+5)/7);
end;
if age20yr=2000 then do;
age20wk=1043+ceil((bbdays+6)/7);
end;
if age20yr=2001 then do;
age20wk=1096+ceil((bbdays+1)/7);
end;
if age20yr=2002 then do;
age20wk=1148+ceil((bbdays+2)/7);
end;
if age20yr=2003 then do;
age20wk=1200+ceil((bbdays+3)/7);
end;
if age20yr=2004 then do;
age20wk=1252+ceil((bbdays+4)/7);
end;
```

end;

**** Convert Birthdate to week number **;**

/ Default birthdate week=0 if birthdate < 12/30/79 */*

birthwk=0;

if birthyr>0 and bbdays>0 then do;

if birthyr=1980 then do;

birthwk=ceil((bbdays+2)/7);

end;

if birthyr=1981 then do;

birthwk=52+ceil((bbdays+4)/7);

end;

if birthyr=1982 then do;

birthwk=104+ceil((bbdays+5)/7);

end;

if birthyr=1983 then do;

birthwk=156+ceil((bbdays+6)/7);

end;

if birthyr=1984 then do;

birthwk=209+ceil((bbdays)/7);

end;

if birthyr=1985 then do;

birthwk=261+ceil((bbdays+2)/7);

end;

end;

/* Section 2: Convert each interview date into a continuous week number. Week 1 is the first week in 1980. */

**** Convert Round 6 Interview Date to week number **;**

*****Convert interview month and day to total days (intdays);**

if r6INT_M>0 and r6INT_D>0 then do;

if r6INT_M=1 then intdays=r6INT_D;

if r6INT_M=2 then intdays=r6INT_D+31;

if r6INT_M=3 then intdays=r6INT_D+59;

if r6INT_M=4 then intdays=r6INT_D+90;

if r6INT_M=5 then intdays=r6INT_D+120;

if r6INT_M=6 then intdays=r6INT_D+151;

if r6INT_M=7 then intdays=r6INT_D+181;

if r6INT_M=8 then intdays=r6INT_D+212;

if r6INT_M=9 then intdays=r6INT_D+243;

if r6INT_M=10 then intdays=r6INT_D+273;

if r6INT_M=11 then intdays=r6INT_D+304;

```
if r6INT_M=12 then intdays=r6INT_D+334;
end;
```

```
***Convert days into week numbers;
```

```
/* Default interview week = 9999 */
intwk=9999;
```

```
if r6INT_Y>0 and intdays>0 then do;
if r6INT_Y=2002 then do;
INTWK=1148+ceil((intdays+2)/7);
```

```
end;
```

```
if r6INT_y=2003 then do;
intwk=1200+ceil((intdays+3)/7);
```

```
end;
```

```
end;
```

```
** Convert Round 5 Interview Date to week number **;
```

```
***Convert interview month and day to total days (intdays);
```

```
if r5INT_M>0 and r5INT_D>0 then do;
```

```
if r5INT_M=1 then r5days=r5INT_D;
```

```
if r5INT_M=2 then r5days=r5INT_D+31;
```

```
if r5INT_M=3 then r5days=r5INT_D+59;
```

```
if r5INT_M=4 then r5days=r5INT_D+90;
```

```
if r5INT_M=5 then r5days=r5INT_D+120;
```

```
if r5INT_M=6 then r5days=r5INT_D+151;
```

```
if r5INT_M=7 then r5days=r5INT_D+181;
```

```
if r5INT_M=8 then r5days=r5INT_D+212;
```

```
if r5INT_M=9 then r5days=r5INT_D+243;
```

```
if r5INT_M=10 then r5days=r5INT_D+273;
```

```
if r5INT_M=11 then r5days=r5INT_D+304;
```

```
if r5INT_M=12 then r5days=r5INT_D+334;
```

```
end;
```

```
***Convert days into week numbers;
```

```
/* Default interview week = 9999 */
r5int=9999;
```

```
if r5INT_Y>0 and r5days>0 then do;
```

```
if r5INT_Y=2001 then do;
```

```
r5int=1096+ceil((r5days+1)/7);
```

```
end;
```

```
if r5INT_y=2002 then do;
```

```
r5int=1148+ceil((r5days+2)/7);
```

```
end;
```

```
end;
```

```
** Convert Round 4 Interview Date to week number **;
```

***Convert interview month and day to total days (r4days);

```
if R4INT_M>0 and R4INT_D>0 then do;
if R4INT_M=1 then r4days=R4INT_D;
if R4INT_M=2 then r4days=R4INT_D+31;
if R4INT_M=3 then r4days=R4INT_D+59;
if R4INT_M=4 then r4days=R4INT_D+90;
if R4INT_M=5 then r4days=R4INT_D+120;
if R4INT_M=6 then r4days=R4INT_D+151;
if R4INT_M=7 then r4days=R4INT_D+181;
if R4INT_M=8 then r4days=R4INT_D+212;
if R4INT_M=9 then r4days=R4INT_D+243;
if R4INT_M=10 then r4days=R4INT_D+273;
if R4INT_M=11 then r4days=R4INT_D+304;
if R4INT_M=12 then r4days=R4INT_D+334;
end;
```

***Account for leap year 2000;

```
if R4INT_M>0 and R4INT_D>0 and R4INT_Y=2000 then do;
if R4INT_M=1 then r4days=R4INT_D;
if R4INT_M=2 then r4days=R4INT_D+31;
if R4INT_M=3 then r4days=R4INT_D+60;
if R4INT_M=4 then r4days=R4INT_D+91;
if R4INT_M=5 then r4days=R4INT_D+121;
if R4INT_M=6 then r4days=R4INT_D+152;
if R4INT_M=7 then r4days=R4INT_D+182;
if R4INT_M=8 then r4days=R4INT_D+213;
if R4INT_M=9 then r4days=R4INT_D+244;
if R4INT_M=10 then r4days=R4INT_D+274;
if R4INT_M=11 then r4days=R4INT_D+305;
if R4INT_M=12 then r4days=R4INT_D+335;
end;
```

***Convert days into week numbers;

```
/* Default interview week = 9999 */
r4int=9999;
```

```
if R4INT_Y>0 and r4days>0 then do;
if R4INT_Y=2000 then do;
r4int=1043+ceil((r4days+6)/7);
end;
if R4INT_y=2001 then do;
r4int=1096+ceil((r4days+1)/7);
end;
end;
```

** Convert Round 3 Interview Date to week number **;

***Convert interview month and day to total days (r3days);

```
if R3INT_M>0 and R3INT_D>0 then do;
if R3INT_M=1 then r3days=R3INT_D;
if R3INT_M=2 then r3days=R3INT_D+31;
if R3INT_M=3 then r3days=R3INT_D+59;
if R3INT_M=4 then r3days=R3INT_D+90;
if R3INT_M=5 then r3days=R3INT_D+120;
if R3INT_M=6 then r3days=R3INT_D+151;
if R3INT_M=7 then r3days=R3INT_D+181;
if R3INT_M=8 then r3days=R3INT_D+212;
if R3INT_M=9 then r3days=R3INT_D+243;
if R3INT_M=10 then r3days=R3INT_D+273;
if R3INT_M=11 then r3days=R3INT_D+304;
if R3INT_M=12 then r3days=R3INT_D+334;
end;
```

```
***Account for leap year 2000;
if R3INT_M>0 and R3INT_D>0 and R3INT_Y=2000 then do;
if R3INT_M=1 then r3days=R3INT_D;
if R3INT_M=2 then r3days=R3INT_D+31;
if R3INT_M=3 then r3days=R3INT_D+60;
if R3INT_M=4 then r3days=R3INT_D+91;
if R3INT_M=5 then r3days=R3INT_D+121;
if R3INT_M=6 then r3days=R3INT_D+152;
if R3INT_M=7 then r3days=R3INT_D+182;
if R3INT_M=8 then r3days=R3INT_D+213;
if R3INT_M=9 then r3days=R3INT_D+244;
if R3INT_M=10 then r3days=R3INT_D+274;
if R3INT_M=11 then r3days=R3INT_D+305;
if R3INT_M=12 then r3days=R3INT_D+335;
end;
```

```
***Convert days into week numbers;
```

```
/* Default interview week = 9999 */
r3int=9999;
```

```
if R3INT_Y>0 and r3days>0 then do;
if R3INT_Y=1999 then do;
r3int=991+ceil((r3days+5)/7);
end;
if R3INT_y=2000 then do;
r3int=1043+ceil((r3days+6)/7);
end;
end;
```

```
** Convert Date of Last Interview (Round 2) to week number **;
```

```
***Convert interview month and day to total days (rd2days);
```

```
if R2INT_M>0 and R2INT_D>0 then do;
if R2INT_M=1 then rd2days=R2INT_D;
if R2INT_M=2 then rd2days=R2INT_D+31;
if R2INT_M=3 then rd2days=R2INT_D+59;
if R2INT_M=4 then rd2days=R2INT_D+90;
if R2INT_M=5 then rd2days=R2INT_D+120;
if R2INT_M=6 then rd2days=R2INT_D+151;
if R2INT_M=7 then rd2days=R2INT_D+181;
if R2INT_M=8 then rd2days=R2INT_D+212;
if R2INT_M=9 then rd2days=R2INT_D+243;
if R2INT_M=10 then rd2days=R2INT_D+273;
if R2INT_M=11 then rd2days=R2INT_D+304;
if R2INT_M=12 then rd2days=R2INT_D+334;
end;
```

```
***Convert days into week numbers;
/* Default interview week = 9999 */
R2INT=9999;
```

```
if R2INT_Y>0 and rd2days>0 then do;
if R2INT_Y=1998 then do;
R2INT=939+ceil((rd2days+4)/7);
end;
end;
```

```
if R2INT_Y>0 and rd2days>0 then do;
if R2INT_Y=1999 then do;
R2INT=991+ceil((rd2days+5)/7);
end;
end;
```

```
** Convert Date of First Interview (Round 1) to week number **;
```

```
***Convert interview month and day to total days (rd1days);
```

```
if R1INT_M>0 and R1INT_D>0 then do;
if R1INT_M=1 then rd1days=R1INT_D;
if R1INT_M=2 then rd1days=R1INT_D+31;
if R1INT_M=3 then rd1days=R1INT_D+59;
if R1INT_M=4 then rd1days=R1INT_D+90;
if R1INT_M=5 then rd1days=R1INT_D+120;
if R1INT_M=6 then rd1days=R1INT_D+151;
if R1INT_M=7 then rd1days=R1INT_D+181;
if R1INT_M=8 then rd1days=R1INT_D+212;
if R1INT_M=9 then rd1days=R1INT_D+243;
if R1INT_M=10 then rd1days=R1INT_D+273;
if R1INT_M=11 then rd1days=R1INT_D+304;
if R1INT_M=12 then rd1days=R1INT_D+334;
end;
```

```

***Convert days into week numbers;
/* Hand edits carried over from Round 2 */
if (pubid=2 or pubid=5615 or pubid=5902) then do;
r1int_y=1997;
end;

/* Default interview week = 9999 */
R1INT=9999;

if R1INT_Y>0 and rd1days>0 then do;
if R1INT_Y=1997 then do;
R1INT=887+ceil((rd1days+3)/7);
end;
end;

if R1INT_Y>0 and rd1days>0 then do;
if R1INT_Y=1998 then do;
R1INT=939+ceil((rd1days+4)/7);
end;
end;

/* Hand edits carried over from Round 2 interview program */
if pubid in (471,476) then do;
r2int=990; r2int_y=1998;
end;
if pubid=7315 then do;
r1int=919; r1int_y=1997;
end;

/* Done to create a seamless flow between rounds */
intwk=intwk-1;

/* Round 2 non-interview case */
if r2int_d=-5 then r2int=-5;
/* Round 3 non-interview case */
if r3int_d=-5 then r3int=-5;
/* Round 4 non-interview case */
if r4int_d=-5 then r4int=-5;
/* Round 5 non-interview case */
if r5int_d=-5 then r5int=-5;
/* Round 6 non-interview case */
if e200=-5 then intwk=-5;

/* Hand edit for 2 cases from the 4 cases checked in the following. I use the variable "CV_int" to double
check the extreme
4 cases in the following. The first two cases are right. And for the last two, I will hand edit them. */
if pubid=1818 then r2int=992;
if pubid=5294 then r2int=989;

/***** Section 3: Convert Start/Stop dates into NLSY97 week numbers

```

*****/

/* This program reads in raw start and stop dates for each job (max=11) and converts them into NLSY97 week numbers. Some start/stop DAYS and MONTHS have been imputed if missing. */

/* Hand edit on 02/19/2004. */

```
if pubid=4830 then do;
stopd04=-3;
stopm04=-3;
stopy04=-3;
end;
```

/* ostartm represents the "old" start dates, used when start dates are updated to interview dates. */

```
array ostartm (i) ostartm01-ostarm11;
array ostartd (i) ostartd01-ostard11;
array ostarty (i) ostarty01-ostary11;
array ostopm (i) ostopm01-ostopm11;
array ostopd (i) ostopd01-ostopd11;
array ostopy (i) ostopy01-ostopy11;
```

```
array startm (i) starm01-starm11;
array startd (i) stard01-stard11;
array starty (i) stary01-stary11;
array stopm (i) stopm01-stopm11;
array stopd (i) stopd01-stopd11;
array stopy (i) stopy01-stopy11;
array UID (i) UID01-UID11;
```

```
array sttdays (i) sttday01-sttday11; /* total days in that year from startdate (to Jan 1) */
array stpdays (i) stpday01-stpday11; /* total days in that year from stopdate (to Jan 1) */
array startwk (i) starw01-starw11;
array stopwk (i) stopw01-stopw11;
array srflag (i) srflg01-srflg11;
array spflag (i) spflg01-spflg11;
array uflag (i) uflag01-uflag11; /* uflag=1 when job startdate is updated */
array smofl (i) smof01-smof11; /* dummy equals 1 when a start month is imputed */
array emofl (i) emof01-emof11; /* dummy equals 1 when a stop month is imputed */
array self (i) self01-self11;
```

/* Establish dli week*/

```
if r5int_y>0 then do;
dli_y=r5int_y;
dli_m=r5int_m;
dli_d=r5int_d;
dliwk=r5int;
r5dli=1;
end;
```

```
if r5int_y=-5 and r4int_y>0 then do;
dli_y=r4int_y;
```



```

dli_m=r4int_m;
dli_d=r4int_d;
dliwk=r4int;
r4dli=1;
end;
if r5int_y=-5 and r4int_y=-5 and r3int_y>0 then do;
dli_y=r3int_y;
dli_m=r3int_m;
dli_d=r3int_d;
dliwk=r3int;
r3dli=1;
end;
if r5int_y=-5 and r4int_y=-5 and r3int_y=-5 and r2int_y>0 then do;
dli_y=r2int_y;
dli_m=r2int_m;
dli_d=r2int_d;
dliwk=r2int;
r2dli=1;
end;
if r5int_y=-5 and r4int_y=-5 and r3int_y=-5 and r2int_y=-5 and r1int_y>0 then do;
dli_y=r1int_y;
dli_m=r1int_m;
dli_d=r1int_d;
dliwk=r1int;
r1dli=1;
end;

```

```

/* Initialize smofl and emofl */
do i=1 to 11;
smofl=0; emofl=0;
end;

```

```

/* Define old start and stop dates */
do i=1 to 11;
ostartm=startm;
ostartd=startd;
ostarty=starty;
ostopm=stopm;
ostopd=stopd;
ostopy=stopy;
end;

```

***Fill-in start/stop day for those missing;

```

/* flag1 = impute start day (valid month)
flag2 = impute start month (valid day)
flag3 = impute start day and month
flag4 = impute stop day (valid month)
flag5 = impute stop month (valid day)
flag6 = impute stop day and month */

```

```
array flag1 (i) flag101-flag111;  
array flag2 (i) flag201-flag211;  
array flag3 (i) flag301-flag311;  
array flag4 (i) flag401-flag411;  
array flag5 (i) flag501-flag511;  
array flag6 (i) flag601-flag611;
```

```
do i=1 to 11;  
flag1=-4;  
flag2=-4;  
flag3=-4;  
flag4=-4;  
flag5=-4;  
flag6=-4;  
end;
```

```
/* Impute missing start days to 1, missing stop days to 28, missing start month to 1 (Jan.) and missing  
stop months to 12 (Dec.). */
```

```
/* Reset flag(1-3) to zero when the start year is valid */
```

```
do over starty;  
if starty>0 then do;  
flag1=0; flag2=0; flag3=0;  
if startm>0 and startd<=0 then do;  
startd=1;  
flag1=1;  
srflag=1;  
end;  
if startm<=0 and startd>0 then do;  
startm=1;  
flag2=1;  
srflag=1;  
smofl=1;  
end;  
if startm<=0 and startd<=0 then do;  
startm=1;  
startd=1;  
flag3=1;  
srflag=1;  
smofl=1;  
end;  
end;  
end;  
end;
```

```
do over stopy;  
if stopy>0 then do;  
flag4=0; flag5=0; flag6=0;  
if stopm>0 and stopd<=0 then do;  
stopd=28;  
flag4=1;  
spflag=1;  
end;  
end;
```

```
if stopm<=0 and stopd>0 then do;
stopm=12;
flag5=1;
spflag=1;
emofl=1;
end;
if stopm<=0 and stopd<=0 then do;
stopm=12;
stopd=28;
flag6=1;
spflag=1;
emofl=1;
end;
```

/* The following lines prevent imputed values for stop months and days to exceed the Round 6 interview date. */

```
if stopy=r6int_y and stopm=r6int_m and stopd>r6int_d and spflag=1 then do;
stopd=r6int_d;
end;
if stopy=r6int_y and stopm>r6int_m then do;
stopm=r6int_m;
stopd=r6int_d;
end;
end;
end;
```

/* The following lines of code account for the cases where the respondent has a start date for a job reported in an earlier round that comes before the dli. The idea is to only count the weeks employed from dli to today. Then, the activity from previous rounds will be added to the Round 6 activity to get the full history. To achieve this, all jobs reported in earlier rounds (UID's begin with 97,98,1999,2000 or 2001) will have their startdates updated to the dli interview date. The start week and stop week for these jobs will be counted the same. */

```
do i=1 to 11;
uflag=0; /* Initialize uflag */
if -5<uid<200200 and startd>-4 then do;
if starty=dli_y and startm=dli_m and startd<dli_d and starty>0 then do;
startd=dli_d;
uflag=1;
end;
if starty=dli_y and startm<dli_m and starty>0 then do;
startm=dli_m;
startd=dli_d;
uflag=1;
end;
if starty<dli_y and starty>0 then do;
starty=dli_y;
```

```
startm=dli_m;
startd=dli_d;
uflag=1;
end;
end;
end;
```

```
***Convert START month and day to total days;
```

```
do over startm;
if startm>0 and startd>0 then do;
if startm=1 then sttdays=startd;
if startm=2 then sttdays=startd+31;
if startm=3 then sttdays=startd+59;
if startm=4 then sttdays=startd+90;
if startm=5 then sttdays=startd+120;
if startm=6 then sttdays=startd+151;
if startm=7 then sttdays=startd+181;
if startm=8 then sttdays=startd+212;
if startm=9 then sttdays=startd+243;
if startm=10 then sttdays=startd+273;
if startm=11 then sttdays=startd+304;
if startm=12 then sttdays=startd+334;
end;
end;
```

```
***Account for leap years;
```

```
do over starty;
if starty=1980 or starty=1984 or starty=1988 or starty=1992 or starty=1996 or starty=2000 then do;
if startm>0 and startd>0 then do;
if startm=1 then sttdays=startd;
if startm=2 then sttdays=startd+31;
if startm=3 then sttdays=startd+60;
if startm=4 then sttdays=startd+91;
if startm=5 then sttdays=startd+121;
if startm=6 then sttdays=startd+152;
if startm=7 then sttdays=startd+182;
if startm=8 then sttdays=startd+213;
if startm=9 then sttdays=startd+244;
if startm=10 then sttdays=startd+274;
if startm=11 then sttdays=startd+305;
if startm=12 then sttdays=startd+335;
end;
end;
end;
```

```
***Convert STOP month and day to total days;
```

```
do over stopm;
if stopm>0 and stopd>0 then do;
```

```
if stopm=1 then stpdays=stopd;
if stopm=2 then stpdays=stopd+31;
if stopm=3 then stpdays=stopd+59;
if stopm=4 then stpdays=stopd+90;
if stopm=5 then stpdays=stopd+120;
if stopm=6 then stpdays=stopd+151;
if stopm=7 then stpdays=stopd+181;
if stopm=8 then stpdays=stopd+212;
if stopm=9 then stpdays=stopd+243;
if stopm=10 then stpdays=stopd+273;
if stopm=11 then stpdays=stopd+304;
if stopm=12 then stpdays=stopd+334;
end;
end;
```

***Account for leap years;

```
do over stopy;
if stopy=1980 or stopy=1984 or stopy=1988 or stopy=1992 or stopy=1996 or stopy=2000 then do;
if stopm>0 and stopd>0 then do;
if stopm=1 then stpdays=stopd;
if stopm=2 then stpdays=stopd+31;
if stopm=3 then stpdays=stopd+60;
if stopm=4 then stpdays=stopd+91;
if stopm=5 then stpdays=stopd+121;
if stopm=6 then stpdays=stopd+152;
if stopm=7 then stpdays=stopd+182;
if stopm=8 then stpdays=stopd+213;
if stopm=9 then stpdays=stopd+244;
if stopm=10 then stpdays=stopd+274;
if stopm=11 then stpdays=stopd+305;
if stopm=12 then stpdays=stopd+335;
end;
end;
end;
```

***Convert days into week numbers;

```
/* Create year flag variable */
array byear (i) byear01-byear11;
array eyear (i) eyear01-eyear11;
```

```
do over starty;
if starty>0 and sttdays>0 then do;
if starty=1980 then do;
startwk=ceil((sttdays+2)/7);
end;
if starty=1981 then do;
startwk=52+ceil((sttdays+4)/7);
end;
if starty=1982 then do;
startwk=104+ceil((sttdays+5)/7);
```

```
end;
if starty=1983 then do;
startwk=156+ceil((sttdays+6)/7);
end;
if starty=1984 then do;
startwk=209+ceil((sttdays)/7);
end;
if starty=1985 then do;
startwk=261+ceil((sttdays+2)/7);
end;
if starty=1986 then do;
startwk=313+ceil((sttdays+3)/7);
end;
if starty=1987 then do;
startwk=365+ceil((sttdays+4)/7);
end;
if starty=1988 then do;
startwk=417+ceil((sttdays+5)/7);
end;
if starty=1989 then do;
startwk=470+ceil((sttdays)/7);
end;
if starty=1990 then do;
startwk=522+ceil((sttdays+1)/7);
end;
if starty=1991 then do;
startwk=574+ceil((sttdays+2)/7);
end;
if starty=1992 then do;
startwk=626+ceil((sttdays+3)/7);
end;
if starty=1993 then do;
startwk=678+ceil((sttdays+5)/7);
end;
if starty=1994 then do;
startwk=730+ceil((sttdays+6)/7);
end;
if starty=1995 then do;
startwk=783+ceil((sttdays)/7);
end;
if starty=1996 then do;
startwk=835+ceil((sttdays+1)/7);
end;
if starty=1997 then do;
startwk=887+ceil((sttdays+3)/7);
end;
if starty=1998 then do;
startwk=939+ceil((sttdays+4)/7);
end;
if starty=1999 then do;
startwk=991+ceil((sttdays+5)/7);
```

```
end;
if starty=2000 then do;
startwk=1043+ceil((sttdays+6)/7);
end;
if starty=2001 then do;
startwk=1096+ceil((sttdays+1)/7);
end;
if starty=2002 then do;
startwk=1148+ceil((sttdays+2)/7);
end;
if starty=2003 then do;
startwk=1200+ceil((sttdays+3)/7);
end;
if starty=2004 then do;
startwk=1252+ceil((sttdays+4)/7);
end;
end;
if starty<0 and starty>-4 then do;
startwk=-3;
end;
if -4<ostarty<0 then byear=1;
end;
```

```
do over stopy;
if stopy>0 and stpdays>0 then do;
if stopy=1980 then do;
stopwk=ceil((stpdays+2)/7);
end;
if stopy=1981 then do;
stopwk=52+ceil((stpdays+4)/7);
end;
if stopy=1982 then do;
stopwk=104+ceil((stpdays+5)/7);
end;
if stopy=1983 then do;
stopwk=156+ceil((stpdays+6)/7);
end;
if stopy=1984 then do;
stopwk=209+ceil((stpdays)/7);
end;
if stopy=1985 then do;
stopwk=261+ceil((stpdays+2)/7);
end;
if stopy=1986 then do;
stopwk=313+ceil((stpdays+3)/7);
end;
if stopy=1987 then do;
stopwk=365+ceil((stpdays+4)/7);
end;
if stopy=1988 then do;
stopwk=417+ceil((stpdays+5)/7);
```

```
end;
if stopy=1989 then do;
stopwk=470+ceil((stpdays)/7);
end;
if stopy=1990 then do;
stopwk=522+ceil((stpdays+1)/7);
end;
if stopy=1991 then do;
stopwk=574+ceil((stpdays+2)/7);
end;
if stopy=1992 then do;
stopwk=626+ceil((stpdays+3)/7);
end;
if stopy=1993 then do;
stopwk=678+ceil((stpdays+5)/7);
end;
if stopy=1994 then do;
stopwk=730+ceil((stpdays+6)/7);
end;
if stopy=1995 then do;
stopwk=783+ceil((stpdays)/7);
end;
if stopy=1996 then do;
stopwk=835+ceil((stpdays+1)/7);
end;
if stopy=1997 then do;
stopwk=887+ceil((stpdays+3)/7);
end;
if stopy=1998 then do;
stopwk=939+ceil((stpdays+4)/7);
end;
if stopy=1999 then do;
stopwk=991+ceil((stpdays+5)/7);
end;
if stopy=2000 then do;
stopwk=1043+ceil((stpdays+6)/7);
end;
if stopy=2001 then do;
stopwk=1096+ceil((stpdays+1)/7);
end;
if stopy=2002 then do;
stopwk=1148+ceil((stpdays+2)/7);
end;
if stopy=2003 then do;
stopwk=1200+ceil((stpdays+3)/7);
end;
if stopy=2004 then do;
stopwk=1252+ceil((stpdays+4)/7);
end;
end;
if stopy<0 and stopy>-4 then do;
```



```
stopwk=-3;
end;
if -4<ostopy<0 then eyear=1;
end;
```

```
/* To create a seamless list of information from dli Round to Round 6, the following lines of code are included. By decreasing the Round 6 interview date by one (which was done in /emp_prep/emp_interview.sas), jobs that are worked up to the interview date are also decreased by one. This is done so that the Round 6 interview week will not be counted twice for respondents with a job during that time. The same procedure is used in the previous rounds. */
```

```
do over stopwk;
if stopwk>0 and UID>0 then do;
if stopwk>intwk then do;
stopwk=intwk;
end;
end;
end;
```

```
/* The following lines considers jobs that begin the same week as the Round 6 interview date. Since we are updating the Round 6 interview week by -1, we need to account for jobs that start in the same week or tenures of -1 will result. */
```

```
do over startwk;
if startwk>0 and UID>0 then do;
if startwk>intwk then do;
startwk=intwk;
end;
end;
end;
```

```
/* The job start date will be cut off at the week of Jan.1 of the respondent's 18th birthday for self-employed jobs. This is done so every starts the self-employed jobs at the same point. The 18th birthday week was not chosen because too many jobs were completely eliminated using the actual 18th birthday. This method cuts off 2 jobs entirely (both the start and stop week are before Jan. 1 of the 18th birthday), and shortens 26 other jobs. */
```

```
array bself (i) bself01-bself11;
array eself (i) eself01-eself11;
do over startwk;
if self=1 and 0<startwk<=(ag18jan-1) then do;
bself=1;
end;
if self=1 and 0<startwk<=(ag18jan-1) and 0<stopwk<=(ag18jan-1) then do;
eself=1;
end;
end;
```

```
do over startwk;
if bself=1 then do;
```

```

startwk=ag18jan;
end;
if eself=1 then do;
startwk=.;
stopwk=.;
end;
end;

/* To account for non-interview cases */
if starm01=-5 then do;
do over startwk;
startwk=-5;
stopwk=-5;
end;
end;

/* Correcting for imputed values that resulted in the start date being later than the stop date. In these
cases, the imputed date will be updated to the good date. */
do i=1 to 11;
if startwk>stopwk and (flag1=1 or flag2=1 or flag3=1) then do;
startwk=stopwk;
end;
if startwk>stopwk and (flag4=1 or flag5=1 or flag6=1) then do;
stopwk=startwk;
end;
end;

/* Check for back-reporters, i.e. people that report a new job (UID>200200) that begins before the last
interview date. */
do over UID;
if startwk<r5int and startwk>0 and r5int>0 then do;
back5=1;
end;
if startwk<r4int and startwk>0 and r5int=-5 and r4int>0 then do;
back4=1;
end;
if startwk<r3int and startwk>0 and r5int=-5 and r4int=-5 and r3int>0 then do;
back3=1;
end;
if startwk<r2int and startwk>0 and r5int=-5 and r4int=-5 and r3int=-5 and r2int>0 then do;
back2=1;
end;
if startwk<r1int and startwk>0 and r5int=-5 and r4int=-5 and r3int=-5 and r2int=-5 and r1int>0 then do;
back1=1;
end;
end;

array br (i) br01-br11;

do over UID;
if startwk<dliwk and startwk>0 and dliwk>0 then do;

```

```
back=1;
br=1;
end;
end;
```

**** Seven people's start week is later than the stop week.(2 for the 1st job, 4 for the 2nd job and 1 for the 3rd job.)
All are for the same reason. Their corresponding "UFLAG" is 1. They reported an old job with starting date and ending date both earlier than the date of last interview, But we only update the starting date to the date last interview, which turns out to be later than the ending date. So I will update these people's ending date to the date of last interview too--this is done on 12/20/2003;

```
array uuflag (i) uuflag01-uuflag11;
do i=1 to 11;
if startwk>0 and stopwk>0 and startwk>stopwk and uflag=1 then do;
uuflag=1;
stopwk=startwk;
end;
end;
```

****** Section 4: Convert dates of within-job gaps into NLSY97 week numbers. ******

**** JOB 1 GAPS **;**

/* These variables are read as follows:
BGDY1_1 = Begin day of within-job gap 1 on job 1
EGMO1_5 = End month of within-job gap 5 on job 1
BGAP1_3 = Begin week of within-job gap 3 on job 1 [CREATED] */

```
array sbgdy (i) sBGDY1_1-sBGDY1_2;
array sbgmo (i) sBGMO1_1-sBGMO1_2;
array sbgyr (i) sBGYR1_1-sBGYR1_2;
array segdy (i) sEGDY1_1-sEGDY1_2;
array segmo (i) sEGMO1_1-sEGMO1_2;
array segyr (i) sEGYR1_1-sEGYR1_2;
```

```
array bgdy (i) BGDY1_1-BGDY1_5;
array bgmo (i) BGMO1_1-BGMO1_5;
array bgyr (i) BGYR1_1-BGYR1_5;
array egdy (i) EGDY1_1-EGDY1_5;
array egmo (i) EGMO1_1-EGMO1_5;
array egyr (i) EGYR1_1-EGYR1_5;
```

```
array bdays (i) bday1_1-bday1_5;
array edays (i) eday1_1-eday1_5;
array bweek (i) bgap1_1-bgap1_5;
array eweek (i) egap1_1-egap1_5;
array bflag (i) bflg1_1-bflg1_5;
```

```
array eflag (i) eflg1_1-eflg1_5;
```

```
array bgfl (i) bgfl1_1-bgfl1_5;
```

```
array egfl (i) egfl1_1-egfl1_5;
```

```
/* use the self-employed gap information if it is a self-employed job. */
```

```
if self01=1 then do;
```

```
gap01=sgap01;
```

```
numgap1=snumgap1;
```

```
do i=1 to 2;
```

```
bgdy = sbgdy;
```

```
bgmo = sbgmo;
```

```
bgyr = sbgyr;
```

```
egdy = segdy;
```

```
egmo = segmo;
```

```
egyr = segyr;
```

```
end;
```

```
do i=3 to 5;
```

```
bgdy = -4;
```

```
bgmo = -4;
```

```
bgyr = -4;
```

```
egdy = -4;
```

```
egmo = -4;
```

```
egyr = -4;
```

```
end;
```

```
end;
```

```
/* Missing gap days (start and stop) are now imputed to 1.
```

```
Impute start/stop dates if only day is missing.
```

```
*/
```

```
do over bgyr;
```

```
if bgyr>0 then do;
```

```
if bgmo>0 and bgdy<=0 then do;
```

```
bgdy=1;
```

```
bflag=1;
```

```
end;
```

```
if bgyr=stary01 and bgmo=starm01 and bgdy<stard01 and bflag=1 then bgdy=stard01;
```

```
end;
```

```
end;
```

```
do over egyr;
```

```
if egyr>0 then do;
```

```
if egmo>0 and egdy<=0 then do;
```

```
egdy=1;
```

```
eflag=1;
```

```
end;
```

```
if egyr=stopy01 and egmo=stopm01 and egdy>stopd01 and eflag=1 then egdy=stopd01;
```

```
end;
```

```
end;
```

```
/*Set flag for gap that exists but has invalid dates*/
```

```
do over bgyr;  
bgfl=-4;  
if (-4 < bgmo < 0) or (-4 < bgyr < 0) then bgfl=1;  
end;
```

```
do over egyr;  
egfl=-4;  
if (-4 < egyr < 0) or (-4 < egmo < 0) then egfl=1;  
end;
```

```
***Convert START month and day to total days (BDAYS);
```

```
do over bgmo;  
if bgmo>0 and bgdy>0 then do;  
if bgmo=1 then bdays=bgdy;  
if bgmo=2 then bdays=bgdy+31;  
if bgmo=3 then bdays=bgdy+59;  
if bgmo=4 then bdays=bgdy+90;  
if bgmo=5 then bdays=bgdy+120;  
if bgmo=6 then bdays=bgdy+151;  
if bgmo=7 then bdays=bgdy+181;  
if bgmo=8 then bdays=bgdy+212;  
if bgmo=9 then bdays=bgdy+243;  
if bgmo=10 then bdays=bgdy+273;  
if bgmo=11 then bdays=bgdy+304;  
if bgmo=12 then bdays=bgdy+334;  
end;  
end;
```

```
***Account for leap years;
```

```
do over bgyr;  
if bgyr=1980 or bgyr=1984 or bgyr=1988 or bgyr=1992 or bgyr=1996 or bgyr=2000 then do;  
if bgmo>0 and bgdy>0 then do;  
if bgmo=1 then bdays=bgdy;  
if bgmo=2 then bdays=bgdy+31;  
if bgmo=3 then bdays=bgdy+60;  
if bgmo=4 then bdays=bgdy+91;  
if bgmo=5 then bdays=bgdy+121;  
if bgmo=6 then bdays=bgdy+152;  
if bgmo=7 then bdays=bgdy+182;  
if bgmo=8 then bdays=bgdy+213;  
if bgmo=9 then bdays=bgdy+244;  
if bgmo=10 then bdays=bgdy+274;  
if bgmo=11 then bdays=bgdy+305;  
if bgmo=12 then bdays=bgdy+335;  
end;  
end;
```

end;

***Convert STOP month and day to total days (EDAYS);

```
do over egmo;
if egmo>0 and egdy>0 then do;
if egmo=1 then edays=egdy;
if egmo=2 then edays=egdy+31;
if egmo=3 then edays=egdy+59;
if egmo=4 then edays=egdy+90;
if egmo=5 then edays=egdy+120;
if egmo=6 then edays=egdy+151;
if egmo=7 then edays=egdy+181;
if egmo=8 then edays=egdy+212;
if egmo=9 then edays=egdy+243;
if egmo=10 then edays=egdy+273;
if egmo=11 then edays=egdy+304;
if egmo=12 then edays=egdy+334;
end;
end;
```

***Account for leap years;

```
do over egr;
if egr=1980 or egr=1984 or egr=1988 or egr=1992 or egr=1996 or egr=2000 then do;
if egmo>0 and egdy>0 then do;
if egmo=1 then edays=egdy;
if egmo=2 then edays=egdy+31;
if egmo=3 then edays=egdy+60;
if egmo=4 then edays=egdy+91;
if egmo=5 then edays=egdy+121;
if egmo=6 then edays=egdy+152;
if egmo=7 then edays=egdy+182;
if egmo=8 then edays=egdy+213;
if egmo=9 then edays=egdy+244;
if egmo=10 then edays=egdy+274;
if egmo=11 then edays=egdy+305;
if egmo=12 then edays=egdy+335;
end;
end;
end;
```

***Convert days into week numbers;

** Note: This program takes the week following the actual start of the gap **
** as the measure of when the non-working period begins. **

```
do over bgr;
if bgr>0 and bdays>0 then do;
```

```
if bgyr=1980 then do;
bweek=ceil((bdays+2)/7);
end;
if bgyr=1981 then do;
bweek=52+ceil((bdays+4)/7);
end;
if bgyr=1982 then do;
bweek=104+ceil((bdays+5)/7);
end;
if bgyr=1983 then do;
bweek=156+ceil((bdays+6)/7);
end;
if bgyr=1984 then do;
bweek=209+ceil((bdays)/7);
end;
if bgyr=1985 then do;
bweek=261+ceil((bdays+2)/7);
end;
if bgyr=1986 then do;
bweek=313+ceil((bdays+3)/7);
end;
if bgyr=1987 then do;
bweek=365+ceil((bdays+4)/7);
end;
if bgyr=1988 then do;
bweek=417+ceil((bdays+5)/7);
end;
if bgyr=1989 then do;
bweek=470+ceil((bdays)/7);
end;
if bgyr=1990 then do;
bweek=522+ceil((bdays+1)/7);
end;
if bgyr=1991 then do;
bweek=574+ceil((bdays+2)/7);
end;
if bgyr=1992 then do;
bweek=626+ceil((bdays+3)/7);
end;
if bgyr=1993 then do;
bweek=678+ceil((bdays+5)/7);
end;
if bgyr=1994 then do;
bweek=730+ceil((bdays+6)/7);
end;
if bgyr=1995 then do;
bweek=783+ceil((bdays)/7);
end;
if bgyr=1996 then do;
bweek=835+ceil((bdays+1)/7);
end;
```

```
if bgyr=1997 then do;
bweek=887+ceil((bdays+3)/7);
end;
if bgyr=1998 then do;
bweek=939+ceil((bdays+4)/7);
end;
if bgyr=1999 then do;
bweek=991+ceil((bdays+5)/7);
end;
if bgyr=2000 then do;
bweek=1043+ceil((bdays+6)/7);
end;
if bgyr=2001 then do;
bweek=1096+ceil((bdays+1)/7);
end;
if bgyr=2002 then do;
bweek=1148+ceil((bdays+2)/7);
end;
if bgyr=2003 then do;
bweek=1200+ceil((bdays+3)/7);
end;
if bweek>0 then do;
bweek=bweek+1;
end;
end;
end;
```

```
do over egyr;
if egyr>0 and edays>0 then do;
if egyr=1980 then do;
ewweek=ceil((edays+2)/7);
end;
if egyr=1981 then do;
ewweek=52+ceil((edays+4)/7);
end;
if egyr=1982 then do;
ewweek=104+ceil((edays+5)/7);
end;
if egyr=1983 then do;
ewweek=156+ceil((edays+6)/7);
end;
if egyr=1984 then do;
ewweek=209+ceil((edays)/7);
end;
if egyr=1985 then do;
ewweek=261+ceil((edays+2)/7);
end;
if egyr=1986 then do;
ewweek=313+ceil((edays+3)/7);
end;
if egyr=1987 then do;
```



```
eweek=365+ceil((edays+4)/7);
end;
if egr=1988 then do;
eweek=417+ceil((edays+5)/7);
end;
if egr=1989 then do;
eweek=470+ceil((edays)/7);
end;
if egr=1990 then do;
eweek=522+ceil((edays+1)/7);
end;
if egr=1991 then do;
eweek=574+ceil((edays+2)/7);
end;
if egr=1992 then do;
eweek=626+ceil((edays+3)/7);
end;
if egr=1993 then do;
eweek=678+ceil((edays+5)/7);
end;
if egr=1994 then do;
eweek=730+ceil((edays+6)/7);
end;
if egr=1995 then do;
eweek=783+ceil((edays)/7);
end;
if egr=1996 then do;
eweek=835+ceil((edays+1)/7);
end;
if egr=1997 then do;
eweek=887+ceil((edays+3)/7);
end;
if egr=1998 then do;
eweek=939+ceil((edays+4)/7);
end;
if egr=1999 then do;
eweek=991+ceil((edays+5)/7);
end;
if egr=2000 then do;
eweek=1043+ceil((edays+6)/7);
end;
if egr=2001 then do;
eweek=1096+ceil((edays+1)/7);
end;
if egr=2002 then do;
eweek=1148+ceil((edays+2)/7);
end;
if egr=2003 then do;
eweek=1200+ceil((edays+3)/7);
end;
if eweek>0 then do;
```

```
eweek=eweek-1;
end;
end;
end;
```

```
/* The following lines omit gap start and stop dates for gaps less than one work week (5 days) */
do over bdays;
if edays-bdays<5 and bweek>eweek and bdays ne . and edays ne . then do;
bweek=.;
eweek=.;
end;
end;
```

```
/* For self-employed jobs, use the "age18jan" as the left-side-limit for gap dates. */
do over bweek;
if self01=1 and 0<bweek<=(ag18jan-1) then bweek=ag18jan;
if self01=1 and 0<bweek<=(ag18jan-1) and 0<eweek<=(ag18jan-1) then do;
bweek=.;
eweek=.;
end;
end;
```

```
/* The following lines are for end gap dates that exceed the interview date due to our rounding methods.
*/
do over eweek;
if 0<eweek<1400 and eweek>intwk then do;
eweek=intwk;
end;
end;
```

```
/* The following omits cases where bweek>eweek, which are caused when missing values are
substituted in.
This can cause problems when writing programs for the created variables. This situation will be fixed by
making
eweek and bweek the same. This is repeated (without comment) for each job. */
```

```
array update (i) updat1_1-updat1_5;
do over bweek;
if 0<eweek<1252 then do;
if bweek>eweek then do;
bweek=eweek;
update=1;
end;
end;
end;
```

```
/* The gap before dli week will be disregarded. */
do over bweek;
if UID01>0 and dliwk>0 and bweek>0 and eweek>0 then do;
if dliwk>bweek and dliwk>eweek then do;
bweek=.;
```

```
eweek=.;
end;
if dliwk<eweeek and dliwk>bweek then do;
bweek=dliwk;
end;
end;
end;
```

```
/* The following erases gaps that occur after the job has ended, only in cases where there are no imputed
job
start or stop months or years (imputed job start/stop days acceptable) */
do over bweek;
if starw01>0 and stopw01>0 and smof01=0 and emof01=0 then do;
if bweek>stopw01 then do;
bweek=.;
end;
if eweeek>stopw01 then do;
eweeek=.;
end;
end;
end;
```

```
/* To correct for bad gap information.*/
```

```
do over bdays;
if eweeek>stopw01 and stopw01>0 then eweeek=stopw01;
if bweek<starw01 and bweek>0 then do;
bweek=starw01;
if eweeek<starw01 and eweeek ne . then do;
eweeek=bweek;
huh01=1;
end;
end;
if bweek>stopw01 and stopw01>0 then huh01=1;
if eweeek<starw01 and eweeek ne . and starw01>0 then huh01=1;
end;
* checked: there is no case with huh01=1;
```

```
/*At this point the program loops through the same code for gaps on jobs 2-5. There are no gaps
reported by any respondent for jobs 6-11. Due to space considerations, this code is not repeated here.
For more information please contact NLS User Services.*/
```

```
/** Note that there is one special hand edit in the JOB 3 GAPS loop. Information is as follows:**
**** checked: pubid=7192 has "huh03=1". He has a repeated but non-continuous job with
uid03=200002. The start month/day of the gap are missing and ending gap dates are before the newly
reported start dates for this job. So I'm going to treat this as a repeated job and ignore the gap reported;
if pubid=7192 then egap3_1=.;
```

```
***** Section 5: Weeks worked by respondent on each job
*****/
```

```
/* This program counts the weeks worked by the respondent and removes the within job gaps. It will place a "1" into weeks where the respondent was employed. Weeks range from the first week of 1980 to the last week of 2003, for a total of 1253 weeks. */
```

```
if pubid=4664 then do;  
smof01=.;  
emof01=.;  
end;
```

```
array job1wks (i) w1_1-w1_1253;  
array job2wks (i) w2_1-w2_1253;  
array job3wks (i) w3_1-w3_1253;  
array job4wks (i) w4_1-w4_1253;  
array job5wks (i) w5_1-w5_1253;  
array job6wks (i) w6_1-w6_1253;  
array job7wks (i) w7_1-w7_1253;  
array job8wks (i) w8_1-w8_1253;  
array job9wks (i) w9_1-w9_1253;  
array job10wks (i) w10_1-w10_1253;  
array job11wks (i) w11_1-w11_1253;
```

```
* Default Settings;
```

```
do i=1 to 1253;  
job1wks=0;  
job2wks=0;  
job3wks=0;  
job4wks=0;  
job5wks=0;  
job6wks=0;  
job7wks=0;  
job8wks=0;  
job9wks=0;  
job10wks=0;  
job11wks=0;  
end;
```

```
* Change military start and stop dates to missing since only civilian jobs are counted;
```

```
if milflag01=1 then do;  
starw01=.;  
stopw01=.;  
end;  
if milflag02=1 then do;  
starw02=.;  
stopw02=.;  
end;  
if milflag03=1 then do;  
starw03=.;  
stopw03=.;  
end;  
if milflag04=1 then do;
```

```
starw04=.;
stopw04=.;
end;
if milflag05=1 then do;
starw05=.;
stopw05=.;
end;
if milflag06=1 then do;
starw06=.;
stopw06=.;
end;
if milflag07=1 then do;
starw07=.;
stopw07=.;
end;
if milflag08=1 then do;
starw08=.;
stopw08=.;
end;
if milflag09=1 then do;
starw09=.;
stopw09=.;
end;
if milflag10=1 then do;
starw10=.;
stopw10=.;
end;
if milflag11=1 then do;
starw11=.;
stopw11=.;
end;
```

```
/* Define rd5wk as the maximum of dliwk and age14wk. This is used for bad start/stop weeks. */
```

```
if dliwk>0 then do;
if dliwk>age14wk then do;
rd5wk=dliwk;
end;
if age14wk=>dliwk then do;
rd5wk=age14wk;
end;
end;
```

```
*** TOTAL WEEKS WORKED ON JOB 1 ***;
```

```
starf_1=0;
stopf_1=0;
```

```
if starw01=-3 and uid01 ne -5 then do;
starw01=rd5wk;
starf_1=1;
end;
```

* "begwk01" is . for this group;

```
if stopw01=-3 and uid01 ne -5 then do;  
stopw01=intwk;  
stopf_1=1;  
end;  
* "endwk01" is . for this group;
```

```
if smof01=1 and bself01 ne 1 then do;  
starf_1=1;  
if stary01=1980 then begwk01=52;  
if stary01=1981 then begwk01=104;  
if stary01=1982 then begwk01=156;  
if stary01=1983 then begwk01=209;  
if stary01=1984 then begwk01=261;  
if stary01=1985 then begwk01=313;  
if stary01=1986 then begwk01=365;  
if stary01=1987 then begwk01=417;  
if stary01=1988 then begwk01=470;  
if stary01=1989 then begwk01=522;  
if stary01=1990 then begwk01=574;  
if stary01=1991 then begwk01=626;  
if stary01=1992 then begwk01=678;  
if stary01=1993 then begwk01=730;  
if stary01=1994 then begwk01=783;  
if stary01=1995 then begwk01=835;  
if stary01=1996 then begwk01=887;  
if stary01=1997 then begwk01=939;  
if stary01=1998 then begwk01=991;  
if stary01=1999 then begwk01=1043;  
if stary01=2000 then begwk01=1096;  
if stary01=2001 then begwk01=1148;  
if stary01=2002 then begwk01=1200;  
if stary01=2003 then begwk01=1252;  
end;
```

```
if emof01=1 then do;  
stopf_1=1;  
if stopy01=1980 then endwk01=1;  
if stopy01=1981 then endwk01=53;  
if stopy01=1982 then endwk01=105;  
if stopy01=1983 then endwk01=157;  
if stopy01=1984 then endwk01=210;  
if stopy01=1985 then endwk01=262;  
if stopy01=1986 then endwk01=314;  
if stopy01=1987 then endwk01=366;  
if stopy01=1988 then endwk01=418;  
if stopy01=1989 then endwk01=471;  
if stopy01=1990 then endwk01=523;  
if stopy01=1991 then endwk01=575;  
if stopy01=1992 then endwk01=627;
```

```
if stopy01=1993 then endwk01=679;
if stopy01=1994 then endwk01=731;
if stopy01=1995 then endwk01=784;
if stopy01=1996 then endwk01=836;
if stopy01=1997 then endwk01=888;
if stopy01=1998 then endwk01=940;
if stopy01=1999 then endwk01=992;
if stopy01=2000 then endwk01=1044;
if stopy01=2001 then endwk01=1097;
if stopy01=2002 then endwk01=1149;
if stopy01=2003 then endwk01=1201;
end;
```

```
if starw01>0 and stopw01>0 then do; /* [1] */
```

```
do i=(starw01) to (stopw01);
job1wks=1;
end;
```

```
*** Remove gap 1 on job 1;
if bgap1_1>0 & egap1_1>0 then do;
do i=(bgap1_1) to (egap1_1);
job1wks=0;
end;
end;
```

```
*** Remove gap 2 on job 1;
if bgap1_2>0 & egap1_2>0 then do;
do i=(bgap1_2) to (egap1_2);
job1wks=0;
end;
end;
```

```
*** Remove gap 3 on job 1;
if bgap1_3>0 & egap1_3>0 then do;
do i=(bgap1_3) to (egap1_3);
job1wks=0;
end;
end;
```

```
*** Remove gap 4 on job 1;
if bgap1_4>0 & egap1_4>0 then do;
do i=(bgap1_4) to (egap1_4);
job1wks=0;
end;
end;
```

```
*** Remove gap 5 on job 1;
if bgap1_5>0 & egap1_5>0 then do;
do i=(bgap1_5) to (egap1_5);
job1wks=0;
```

```
end;  
end;
```

```
*** Remove gap 1 on job 1 - beginning gap date bad;
```

```
if bgfl1_1=1 & egap1_1>0 then do;
```

```
do i=(starw01) to (egap1_1);
```

```
job1wks=-3;
```

```
gpfl1_1=1;
```

```
end;
```

```
end;
```

```
*** Remove gap 1 on job 1 - end gap date bad;
```

```
if bgap1_1>0 & egfl1_1=1 then do;
```

```
do i=(bgap1_1) to min(stopw01,bgap1_2-1,bgap1_3-1,bgap1_4-1,bgap1_5-1,
```

```
egap1_2+1,egap1_3+1,egap1_4+1,egap1_5+1);
```

```
job1wks=-3;
```

```
gpfl1_1=1;
```

```
end;
```

```
end;
```

```
*** Remove gap 1 on job 1 - both gap dates bad;
```

```
if bgfl1_1=1 & egfl1_1=1 then do;
```

```
do i=(starw01) to min(stopw01,bgap1_2-1,bgap1_3-1,bgap1_4-1,bgap1_5-1,
```

```
egap1_2+1,egap1_3+1,egap1_4+1,egap1_5+1);
```

```
job1wks=-3;
```

```
gpfl1_1=1;
```

```
end;
```

```
end;
```

```
*** Remove gap 2 on job 1 - beginning gap date bad;
```

```
if bgfl1_2=1 & egap1_2>0 then do;
```

```
do i=max(starw01,bgap1_1-1,egap1_1+1) to (egap1_2);
```

```
job1wks=-3;
```

```
gpfl1_2=1;
```

```
end;
```

```
end;
```

```
*** Remove gap 2 on job 1 - end gap date bad;
```

```
if bgap1_2>0 & egfl1_2=1 then do;
```

```
do i=(bgap1_2) to min(stopw01,bgap1_3-1,bgap1_4-1,bgap1_5-1,
```

```
egap1_3+1,egap1_4+1,egap1_5+1);
```

```
job1wks=-3;
```

```
gpfl1_2=1;
```

```
end;
```

```
end;
```

```
*** Remove gap 2 on job 1 - both gap dates bad;
```

```
if bgfl1_2=1 & egfl1_2=1 then do;
```

```
do i=max(starw01,bgap1_1-1,egap1_1+1) to min(stopw01,bgap1_3-1,bgap1_4-1,bgap1_5-1,
```

```
egap1_3+1,egap1_4+1,egap1_5+1);
```

```
job1wks=-3;
```

```
gpfl1_2=1;
```

```
end;
```

```
end;
```



```

*** Remove gap 3 on job 1 - beginning gap date bad;
if bgfl1_3=1 & egap1_3>0 then do;
do i=max(starw01,bgap1_1-1,bgap1_2-1,egap1_1+1,egap1_2+1) to (egap1_3);
job1wks=-3;
gpfl1_3=1;
end;
end;
*** Remove gap 3 on job 1 - end gap date bad;
if bgap1_3>0 & egfl1_3=1 then do;
do i=(bgap1_3) to min(stopw01,bgap1_4-1,bgap1_5-1,
egap1_4+1,egap1_5+1);
job1wks=-3;
gpfl1_3=1;
end;
end;
*** Remove gap 3 on job 1 - both gap dates bad;
if bgfl1_3=1 & egfl1_3=1 then do;
do i=max(starw01,bgap1_1-1,bgap1_2-1,egap1_1+1,egap1_2+1) to min(stopw01,bgap1_4-1,bgap1_5-1,
egap1_4+1,egap1_5+1);
job1wks=-3;
gpfl1_3=1;
end;
end;

*** Remove gap 4 on job 1 - beginning gap date bad;
if bgfl1_4=1 & egap1_4>0 then do;
do i=max(starw01,bgap1_1-1,bgap1_2-1,bgap1_3-1,egap1_1+1,egap1_2+1,egap1_3+1) to (egap1_4);
job1wks=-3;
gpfl1_4=1;
end;
end;
*** Remove gap 4 on job 1 - end gap date bad;
if bgap1_4>0 & egfl1_4=1 then do;
do i=(bgap1_4) to min(stopw01,bgap1_5-1,egap1_5+1);
job1wks=-3;
gpfl1_4=1;
end;
end;
*** Remove gap 4 on job 1 - both gap dates bad;
if bgfl1_4=1 & egfl1_4=1 then do;
do i=max(starw01,bgap1_1-1,bgap1_2-1,bgap1_3-1,egap1_1+1,egap1_2+1,egap1_3+1) to min
(stopw01,bgap1_5-1,egap1_5+1);
job1wks=-3;
gpfl1_4=1;
end;
end;

*** Remove gap 5 on job 1 - beginning gap date bad;
if bgfl1_5=1 & egap1_5>0 then do;
do i=max(starw01,bgap1_1-1,bgap1_2-1,bgap1_3-1,bgap1_4-

```

```

1,egap1_1+1,egap1_2+1,egap1_3+1,egap1_4+1) to (egap1_5);
job1wks=-3;
gpfl1_5=1;
end;
end;
*** Remove gap 5 on job 1 - end gap date bad;
if bgap1_5>0 & egfl1_5=1 then do;
do i=(bgap1_5) to (stopw01);
job1wks=-3;
gpfl1_5=1;
end;
end;
*** Remove gap 5 on job 1 - both gap dates bad;
if bgfl1_5=1 & egfl1_5=1 then do;
do i=max(starw01,bgap1_1-1,bgap1_2-1,bgap1_3-1,bgap1_4-
1,egap1_1+1,egap1_2+1,egap1_3+1,egap1_4+1) to (stopw01);
job1wks=-3;
gpfl1_5=1;
end;
end;

end; /* [1] */

if starf_1=1 then do;
do i=(starw01) to min(stopw01,begwk01,bgap1_1-1,bgap1_2-1,bgap1_3-1,bgap1_4-1,bgap1_5-1,
egap1_1+1,egap1_2+1,egap1_3+1,egap1_4+1,egap1_5+1);
job1wks=-3;
end;
end;

if stopf_1=1 then do;
do i=max(starw01,endwk01,bgap1_1-1,bgap1_2-1,bgap1_3-1,bgap1_4-1,bgap1_5-1,
egap1_1+1,egap1_2+1,egap1_3+1,egap1_4+1,egap1_5+1)
to (stopw01);
job1wks=-3;
end;
end;

/*At this point the program repeats the above code for jobs 2-11. This code is omitted due to space
considerations. For more information, please contact NLS User Services.*/

endsas;

```