/\* TAXSIM 27 PREP CODE FOR CPS ASEC

REVISED MAY 24, 2019

THIS FILE PREPARES CURRENT POPULATION ANNUAL SOCIAL AND ECONOMIC SUPPLEMENT (CPS ASEC) DATA FOR NBER'S **TAXSIM 27** PROGRAM FOR USE IN STATA. THE VARIABLE NAMES ARE THOSE ASSIGNED IN INTERNAL VERSIONS OF THE ASEC AND WILL NEED TO BE MODIFIED FOR VERSIONS DOWLOADED FROM IPUMS OR UNICON'S CPS UTILITIES PROGRAM. THIS CODE IS MODIFIED FROM AN EARLIER VERSION DEVELOPED BY JUDITH SCOTT CLAYTON found at http://users.nber.org/~taxsim/to-taxsim/cps/cps-clayton/

\*/

/\* start of program \*/

#d ;

/\* It is assumed that the user has a ASEC file opened at this point. TAXSIM does not like too much data at once so going to run year-by-year and then reassemble below. The startyear and endyear can be modified by the user.

YEAR refers to tax year and not survey year \*/

local period year;

local startyear 1979;

local endyear 2015;

forvalues j=`startyear'/`endyear' {;

preserve;

keep if `j'==`period';

label var year “[TS2] Tax year”;

/\* destringing can (and should) be done in an earlier stage \*/

destring\_all,replace;

/\* define family relations \*/

/\*Unique family identifier \*/

gen double uniqfam=(ph\_seq\*10000)+(fseq\*1000)+(a\_famnum\*100)+(a\_famtyp);

label var uniqfam "Unique family identifier" ;

/\* define head and spouse of household, let head of related subfamilies be own tax unit \*/

gen head=0;

/\* head of primary family \*/

replace head=1 if a\_famtyp==1 & hhdfmx==1;

/\* unrelated partner or individual \*/

replace head=1 if (a\_famtyp==5 & hhdfmx==50) | (a\_famtyp==5 & hhdfmx==51);

/\* head of unrelated secondary family \*/

replace head=1 if a\_famtyp==4 & hhdfmx==46 ;

/\* head of subfamily \*/

replace head=1 if (a\_famtyp==3 & hhdfmx==3) | (a\_famtyp==3 & hhdfmx==5) | (a\_famtyp==3 & hhdfmx==8) | ///

(a\_famtyp==3 & hhdfmx==10) | (a\_famtyp==3 & hhdfmx==13) | (a\_famtyp==3 & hhdfmx==15) | (a\_famtyp==3 & hhdfmx==18) | ///

(a\_famtyp==3 & hhdfmx==20) | (a\_famtyp==3 & hhdfmx==23) | (a\_famtyp==3 & hhdfmx==26) | (a\_famtyp==3 & hhdfmx==30) | ///

(a\_famtyp==3 & hhdfmx==32) | (a\_famtyp==3 & hhdfmx==35) | (a\_famtyp==3 & hhdfmx==38) | (a\_famtyp==3 & hhdfmx==41) | ///

(a\_famtyp==3 & hhdfmx==43);

/\* primary individual \*/

replace head=1 if (a\_famtyp==2 & hhdfmx==49) | (a\_famtyp==2 & hhdfmx==51);

/\* starting in 1987 tax year can identify foster kids under age 18 who

can be listed as heads but are then claimed as children, so set their head = 0 \*/

replace head=0 if (a\_famtyp==5 & hhdfmx==50 & a\_exprrp==11 & a\_age < 18);

sum head;

/\* compute total number of heads per unique unit..should be 1...but can be 0 for those in

group quarters and can't be uniquely identified with uniqfam \*/

bysort uniqfam: egen tothd=total(head);

tab tothd;

/\* Create identifier for spouse \*/

gen t\_spouse=0;

/\* spouse of primary family \*/

replace t\_spouse=1 if a\_famtyp==1 & hhdfmx==2;

/\* spouse of unrelated secondary family \*/

replace t\_spouse=1 if a\_famtyp==4 & hhdfmx==47 ;

/\* spouse of related subfamily \*/

replace t\_spouse=1 if (a\_famtyp==3 & hhdfmx==6) | (a\_famtyp==3 & hhdfmx==11) | (a\_famtyp==3 & hhdfmx==16) | ///

(a\_famtyp==3 & hhdfmx==21) | (a\_famtyp==3 & hhdfmx==27) | (a\_famtyp==3 & hhdfmx==33) | (a\_famtyp==3 & hhdfmx==39) | ///

(a\_famtyp==3 & hhdfmx==44);

sum t\_spouse;

/\* define qualifying child. Depending on the version of the ASEC this code may need to be modified for tax years prior to 1987. EITC defines qualifying child as those 18 and younger, or ages 19-23 (inclusive) if their primary activity is school \*/

gen child = 0;

/\* primary family child \*/

replace child = 1 if (a\_famtyp==1 & hhdfmx==4) | ///

(a\_famtyp==1 & hhdfmx==7) | ///

(a\_famtyp==1 & hhdfmx==9 & a\_age > 18 & a\_age < 24 & a\_ftpt==1) | ///

(a\_famtyp==1 & hhdfmx==9 & a\_age < 19) | ///

(a\_famtyp==1 & hhdfmx==12 & a\_age > 18 & a\_age < 24 & a\_ftpt==1) | ///

(a\_famtyp==1 & hhdfmx==12 & a\_age < 19) | ///

(a\_famtyp==1 & hhdfmx==14) | ///

(a\_famtyp==1 & hhdfmx==17) | ///

(a\_famtyp==1 & hhdfmx==19 & a\_age > 18 & a\_age < 24 & a\_ftpt==1) | ///

(a\_famtyp==1 & hhdfmx==19 & a\_age < 19) | ///

(a\_famtyp==1 & hhdfmx==22 & a\_age > 18 & a\_age < 24 & a\_ftpt==1) | ///

(a\_famtyp==1 & hhdfmx==22 & a\_age < 19) | ///

(a\_famtyp==1 & hhdfmx==25) | ///

(a\_famtyp==1 & hhdfmx==29) | ///

(a\_famtyp==1 & hhdfmx==31 & a\_age > 18 & a\_age < 24 & a\_ftpt==1) | ///

(a\_famtyp==1 & hhdfmx==31 & a\_age < 19) | ///

(a\_famtyp==1 & hhdfmx==34 & a\_age > 18 & a\_age < 24 & a\_ftpt==1) | ///

(a\_famtyp==1 & hhdfmx==34 & a\_age < 19) | ///

(a\_famtyp==1 & hhdfmx==37) | ///

(a\_famtyp==1 & hhdfmx==40) | ///

(a\_famtyp==1 & hhdfmx==42 & a\_age > 18 & a\_age < 24 & a\_ftpt==1) | ///

(a\_famtyp==1 & hhdfmx==42 & a\_age < 19) | ///

(a\_famtyp==1 & hhdfmx==45 & a\_age > 18 & a\_age < 24 & a\_ftpt==1) | ///

(a\_famtyp==1 & hhdfmx==45 & a\_age < 19) ;

/\* unrelated secondary family child \*/

replace child = 1 if (a\_famtyp==4 & hhdfmx==48);

/\* related subfamily child \*/

replace child = 1 if (a\_famtyp==3 & hhdfmx==24) | (a\_famtyp==3 & hhdfmx==36);

/\* related totally disabled child regardless of age is eligible for EITC qualifying child \*/

replace child=1 if (dis\_hp==1 & a\_famtyp==1 & rsnnotw==1 & a\_lfsr==6) & ///

(hhdfmx==9 | hhdfmx==12 | hhdfmx==19 | hhdfmx==22 | hhdfmx==31 | ///

hhdfmx==34 | hhdfmx==42 | hhdfmx==45);

/\* try to separate nonqualifying adult children into own units \*/

gen tempfiler=1 if child==0 & head==0 & t\_spouse==0;

sort uniqfam tempfiler;

by uniqfam: gen nfiler=\_n if tempfiler==1;

egen caseid=group(uniqfam nfiler) if tempfiler==1;

sum caseid uniqfam,detail;

replace uniqfam=caseid if tempfiler==1;

sum tempfiler;

/\* define total dependents \*/

bysort uniqfam: egen depx=total(child);

/\* to make sure we get foster kids...assign them to DEPX in primary family or individual

Depending on the version of the ASEC this code may need to be modified for tax years prior to 1985 \*/

gen foster=0;

replace foster = 1 if (a\_famtyp == 5 & hhdfmx == 50 & a\_exprrp==11 & a\_age <= 18) | ///

(a\_famtyp == 5 & hhdfmx == 50 & a\_exprrp==11 & a\_age > 18 & a\_age < 24 & a\_ftpt==1);

bysort ph\_seq: egen totfoster=total(foster);

replace totfoster=0 if tempfiler==1;

replace depx=depx+totfoster if a\_famtyp==1 | a\_famtyp==2;

sum depx child foster totfoster;

/\* # of dependents is capped in Tax Code and Taxsim \*/

replace depx=15 if depx > 15;

label var depx “[TS7] Number of dependents”;

/\* dependent children for EITC (dep18), other child credits (dep17), and dependent care (dep13) \*/

gen dep18=depx;

label var dep18 “[TS10] Number of EITC dependents”;

gen child17 = 0;

replace child17 = 1 if child==1 & a\_age < 17;

sum child17;

bysort uniqfam: egen dep17=total(child17);

/\* to make sure we get foster kids < age 17 ...assign them to dep17 in primary family or individual \*/

gen foster17=0;

replace foster17 = 1 if foster==1 & a\_age < 17;

bysort ph\_seq: egen totfoster17=total(foster17);

replace totfoster17=0 if tempfiler==1;

replace dep17=dep17+totfoster17 if a\_famtyp==1 | a\_famtyp==2;

/\* # of dependents is capped in Tax Code and Taxsim \*/

replace dep17=15 if dep17 > 15;

label var dep17 “[TS9] Number of child tax credit dependents”;

gen child13 = 0;

replace child13 = 1 if child==1 & a\_age < 13;

sum child13;

bysort uniqfam: egen dep13=total(child13);

/\* to make sure we get foster kids < age 13 ...assign them to dep13 in primary family or individual \*/

gen foster13=0;

replace foster13 = 1 if foster==1 & a\_age < 13;

bysort ph\_seq: egen totfoster13=total(foster13);

replace totfoster13=0 if tempfiler==1;

replace dep13=dep13+totfoster13 if a\_famtyp==1 | a\_famtyp==2;

/\* # of dependents is capped in Tax Code and Taxsim \*/

replace dep13=15 if dep13 > 15;

label var dep13 “[TS8] Number of child care dependents”;

/\* taxsim command mstat to determine filing status \*/

gen married=0;

replace married=1 if a\_spouse>0;

bysort uniqfam: egen sppres=max(t\_spouse);

replace married=1 if married==0 & sppres==1;

gen mstat=0 if child==1; /\*equivalent to FILESTAT = 6;\*/

replace mstat=1 if child==0 & married==0;

replace mstat=2 if child==0 & married==1;

label def mstat 0 "Nonfiler" 1 "Single" 2 "Joint" ;

label val mstat mstat ;

label var mstat "[TS4] Tax filing marital status " ;

tab mstat,mi;

tab mstat a\_spouse,mi;

/\* Prepping income data for TAXSIM. (these will aggregate in next program)

Depending on the version of the ASEC used some of the income terms change names in the 1980s and will need to be modified \*/

/\*Income: wage & salary, nonfarm self-employ, farm/nonincorporated SE.

taxsim will not accept negative wages - put negatives in otherprop \*/

gen wages\_i= ern\_val + ws\_val + se\_val + frm\_val if se\_val>=0 & frm\_val>=0;

replace wages\_i=ern\_val + ws\_val if se\_val<0 | frm\_val<0 ;

/\* private disability retirement income is considered earned income if under minimum retirement age, assumed 56 \*/

g prdisab1=0;

replace prdisab1=1 if (dis\_sc1 >=2 & dis\_sc1 <=5);

g prdisab2=0;

replace prdisab2=1 if (dis\_sc2 >=2 & dis\_sc2 <=5);

Replace wages\_i=wages\_i + dis\_val1 if a\_age < 56 & prdisab1==1;

Replace wages\_i=wages\_i + dis\_val2 if a\_age < 56 & prdisab1==2;

label var wages\_i "Individual wages" ;

/\*Income: dividends \*/

gen dividends\_i=div\_val ;

label var dividends\_i "Individual dividends" ;

/\* Income: interest \*/

gen intrec\_i=int\_val;

label var intrec\_i “Individuals interest received”;

/\*Income: other property income \*/

gen otherprop\_i=rnt\_val if se\_val>=0 & frm\_val>=0;

replace otherprop\_i=rnt\_val+se\_val+frm\_val if semp\_val<0 | frse\_val<0;

label var otherprop\_i "Individual otherprop" ;

/\*Income: non-property income such as alimony \*/

gen nonprop\_i=alm\_val ;

label var nonprop\_i "Individual nonprop" ;

/\*Income: private retirement funds \*/

gen pensions\_i= rtm\_val;

label var pensions\_i "Individual pensions" ;

/\* Income: social security, survivors, and disability –if under age 56 only assign non-retirement disability to pension \*/

gen gssi\_i= ss\_val + srvs\_val if a\_age < 56;

replace gssi\_i= gssi\_i + dis\_val1 if a\_age < 56 & prdisab1==0;

replace gssi\_i= gssi\_i + dis\_val2 if a\_age < 56 & prdisab2==0;

replace gssi\_i = ss\_val + srvs\_val + dsab\_val if a\_age > 55;

label var gssi\_i "Individual gross social security" ;

/\*Income: public assist/welfare, workers comp, veterans payments, child support, supplemental security income \*/

gen transfers\_i= paw\_val + wc\_ + vet\_val + csp\_val + ssi\_val;

label var transfers\_i "Individual transfer income" ;

/\*Income: unemployment compensation \*/

gen ui\_i= uc\_val;

label var ui\_i "Individual UI income" ;

/\*Ignored sources of income

inced Income: educational assistance

incoth Income: other sources

NOTE: cpgain and cploss are imputed by CPS - attached to "tax filing head"

prptax imputation is also available and attached to every member of a household

BOTH OF THESE VARS WILL BE PICKED UP WHEN DATA ARE AGGREGATED TO TAX FILING UNIT

\*/

/\* Any self-employment income - pos or neg \*/

gen selfemp=(se\_val!=0) ;

label var selfemp "Person had any self-emp income (pos or neg)" ;

gen selfemp2=(se\_val!=0 | frm\_val!=0);

label var selfemp2 "Person had any self-emp or farm income (pos or neg)" ;

/\* Rename State for Taxsim. Variable name for STATE depends on version of ASEC used \*/

gen cpsstate=state;

drop state;

gen state=0;

replace state=1 if cpsstate==63; replace state=2 if cpsstate==94; replace state=3 if cpsstate==86;

replace state=4 if cpsstate==71; replace state=5 if cpsstate==93; replace state=6 if cpsstate==84;

replace state=7 if cpsstate==16; replace state=8 if cpsstate==51; replace state=9 if cpsstate==53;

replace state=10 if cpsstate==59; replace state=11 if cpsstate==58; replace state=12 if cpsstate==95;

replace state=13 if cpsstate==82; replace state=14 if cpsstate==33; replace state=15 if cpsstate==32;

replace state=16 if cpsstate==42; replace state=17 if cpsstate==47; replace state=18 if cpsstate==61;

replace state=19 if cpsstate==72; replace state=20 if cpsstate==11; replace state=21 if cpsstate==52;

replace state=22 if cpsstate==14; replace state=23 if cpsstate==34; replace state=24 if cpsstate==41;

replace state=25 if cpsstate==64; replace state=26 if cpsstate==43; replace state=27 if cpsstate==81;

replace state=28 if cpsstate==46; replace state=29 if cpsstate==88; replace state=30 if cpsstate==12;

replace state=31 if cpsstate==22; replace state=32 if cpsstate==85; replace state=33 if cpsstate==21;

replace state=34 if cpsstate==56; replace state=35 if cpsstate==44; replace state=36 if cpsstate==31;

replace state=37 if cpsstate==73; replace state=38 if cpsstate==92; replace state=39 if cpsstate==23;

replace state=40 if cpsstate==15; replace state=41 if cpsstate==57; replace state=42 if cpsstate==45;

replace state=43 if cpsstate==62; replace state=44 if cpsstate==74; replace state=45 if cpsstate==87;

replace state=46 if cpsstate==13; replace state=47 if cpsstate==54; replace state=48 if cpsstate==91;

replace state=49 if cpsstate==55; replace state=50 if cpsstate==35; replace state=51 if cpsstate==83;

label var state “[TS3] State ID”;

/\*Age has been replaced in Taxsim27 with page and sage \*/

gen age\_head=a\_age if head==1;

bysort uniqfam: egen mn\_age\_head=max(age\_head);

replace mn\_age\_head=a\_age if tempfiler==1;

gen page = mn\_age\_head;

gen age\_spouse=0 if t\_spouse==0;

replace age\_spouse=a\_age if t\_spouse==1;

bysort uniqfam: egen mn\_age\_spouse=max(age\_spouse);

gen sage = mn\_age\_spouse if head==1;

label var page "[TS5] Age of head " ;

label var sage "[TS6] Age of spouse " ;

/\* create wages for head, spouse, and child, but do not assign child earnings.

Some children probably need to file separately but this will be missed here.

Aggregate up using uniqfam \*/

gen hwages=wages\_i if head==1 | tempfiler==1;

bysort uniqfam: egen pwages=total(hwages);

gen spwages=wages\_i if t\_spouse==1;

bysort uniqfam: egen swages=total(spwages);

gen othwages\_i=wages\_i if child==1;

bysort uniqfam: egen othwages=total(othwages\_i);

\*replace swages=swages+othwages;

label var pwages "[TS11] Primary taxpayer wage income" ;

label var swages "[TS12] Secondary taxpayer wage income" ;

/\*Note: In a joint filing tax unit, there are two records, one for each as a primary taxpayer \*/

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ;

\* Sum up other income sources by uniqfam

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ;

bysort uniqfam: egen dividends=total(dividends\_i) ;

bysort uniqfam: egen intrec=total(intrec\_i) ;

bysort uniqfam: egen otherprop=total(otherprop\_i) ;

bysort uniqfam: egen nonprop=total(nonprop\_i) ;

bysort uniqfam: egen pensions=total(pensions\_i) ;

bysort uniqfam: egen gssi=total(gssi\_i) ;

bysort uniqfam: egen transfers=total(transfers\_i) ;

bysort uniqfam: egen ui=total(ui\_i) ;

label var dividends "[TS13] Dividend income" ;

label var intrec "[TS14] Interest income" ;

label var otherprop "[TS17] Rental income" ;

label var nonprop "[TS18] Alimony income" ;

label var pensions "[TS19] Pensions/retirement income" ;

label var gssi "[TS20] Social sec, supp sec, survivors, disability bens" ;

label var transfers "[TS22] Welfare, WC, Vets, child supp" ;

label var ui "[TS21] Unemployment compensation" ;

\*The following vars are unavailable ;

gen rentpaid=0 ;

gen otheritem=0 ;

gen childcare=0 ; /\* care\_val is available starting with 2010 tax year \*/

gen mortgage=0 ;

label var rentpaid "[TS23] Rent paid - NOT AVAIL FOR CPS" ;

label var otheritem "[TS25] Oth itemized deductions - NOT AVAIL FOR CPS" ;

label var childcare "[TS26] Child care expenses - AVAIL FOR CPS starting 2011 survey" ;

label var mortgage "[TS27] Mortgage interest paid - NOT AVAIL FOR CPS" ;

/\* The following vars are CPS imputations

Property taxes paid.

These are originally attached to all persons in the HOUSEHOLD

Fix so that values are assigned only to one tax unit within hhold

Order of assignment: joint filers, head of hhold, single filers

If there are multiple singles in a hhhold with no joint or hhead filers, then split proptax across singles \*/

gen joint=(mstat==2) ;

gen hhead=(mstat==1 & depx > 0) ;

gen single=(mstat==1 & depx==0) ;

by ph\_seq, sort: egen anyjoint=max(joint) ;

by ph\_seq, sort: egen anyhhead=max(hhead) ;

by ph\_seq, sort: egen numsingle=sum(single) ;

gen proptax=prop\_tax;

replace proptax=0 if mstat==0 ; /\* nonfilers \*/

replace proptax=0 if (mstat==1 | mstat==3) & anyjoint==1 ; /\* hheads or single filers in hholds with joint filers \*/

replace proptax=0 if mstat==1 & anyhhead==1 ; /\* single filers in hholds with hhhead filers \*/

replace proptax=proptax/numsingle if mstat==1 & anyjoint==0 & anyhhead==0 ; /\* single filers in hhold with no hheads or joint filers \*/

label var proptax "[TS24] Property tax paid (imputed by CPS)" ;

/\* capital gains not collected in ASEC \*/

gen stcg=0 ;

label var stcg "[TS15] Short-term cap gains - NOT AVAIL FOR CPS" ;

gen ltcg=0 ;

label var ltcg "[TS16] Long-term cap gains - NOT AVAIL FOR CPS" ;

/\* TAXSIM will not take negative values \*/

replace pwages=0 if pwages < 0;

replace swages=0 if swages < 0 | swages==.;

replace otherprop=0 if otherprop < 0;

replace nonprop=0 if nonprop < 0;

replace intrec=0 if intrec < 0;

replace childcare=0 if childcare < 0;

replace dividends=0 if dividends < 0;

replace pensions=0 if pensions < 0;

replace gssi=0 if gssi <0;

replace transfers=0 if transfers <0;

replace ui=0 if ui <0;

replace proptax=0 if proptax < 0;

replace depx=0 if depx < 0;

replace dep18=0 if dep18 < 0;

replace dep17=0 if dep17 < 0;

replace dep13=0 if dep13 < 0;

/\* \*\*\*\*RUN TAXSIM year by year and then reassemble \*\*\*\* \*/

count;

taxsim27, full replace;

tempfile dump`j';

save "`dump`j''";

restore;

};

use "`dump1979'",clear;

local newstart 1980;

forvalues i=`newstart'/`endyear' {;

quietly append using "`dump`i''";

};