

**ALBANIA:  
EMPLOYMENT AND WELFARE SURVEY  
1996**

**DATA CHECKING PROGRAM**

Dictionary: ALBANIA

Program: ALBANIA

```
1 . DATA CHECKING PROGRAM
2 .
3 CONTROL
4 Questionnaire
5 hhnum
6
7 max-records sec011 18
8 sec02 18
9 sec031 15
10 sec033 15
11 sec061 15
12 sec10 45
13
14 REPORTS summary
15 questionnaire
16
17 VARIABLES
18 ind
19 agediff
20 prev
21 pnum
22 ind031 miss031
23 ind032 miss032
24 ind033 miss033
25 ind041 miss041
26 ind051 miss051
27 ind061 miss061
28 flag
29 LARGE
30 valdenim
31 fqpur fqrec fqpro totfood
32
33 For-new-quest
34
35 let ind = 0
36
```

Dictionary: ALBANIA

Program: ALBANIA

```
37 ..*****
38 . ** Section 0 COVER PAGE *****
39 . *****
40 For-each (sec00)
41
42 .sec 00 err01 - Check location codes against the HH code
43   if hhidcode(1) <> hhnum
44     msg 'Sec 00 Hh ID does not correspond to location ID'
45     bashki sector hhold hhnum
46   end-if
47
48 .sec 00 err02 - Time of Interview
49   if timeend = Blank or timest = Blank
50     msg "Sec 00 Interview time has not been reported"
51   end-if
52
```

Dictionary: ALBANIA

Program: ALBANIA

```
53 ..*****
54 . ** Section 1.1 ROSTER *****
55 . *****
56 For-each (sec011)
57
58 let ind = ind + 1 . Count the number of people in the Roster
59
60 .sec 1.1 err01 - Make sure that there is one individual per record
61   if pcode011 <> rec-number
62     msg 'Sec 1.1 Person Number does not Correspond '
63       rec-number pcode011
64   end-if
65
66 .sect 1.1 err02 - One head reported in line 01
67   if relation = 1 and rec-number <> 1
68     msg 'Sec 1.1 Household Head is not reported as number 1'
69       rec-number pcode011 relation age
70   end-if
71
72 .sect 1.1 err03 - Age of Head
73   if relation = 1 and age < 16
74     msg 'Sec 1.1 The household head is too young'
75       pcode011 relation age
76   end-if
77
78 .sec 1.1 err04 - Check age and gender of married people
79   if civil=1
80     if age < 15
81       msg 'Sec 1.1 This person is too young to be married'
82       pcode011 civil age
83     else
84       if spouse=NA or spouse>type-count(sec011)
85         msg 'Sec 1.1 The code of the spouse is not valid'
86         pcode011 civil spouse
87       else
88         if spouse<>99
89           if spouse = pcode011
90             msg 'Sec 1.1 This person is married to him/her self !!!'
91             pcode011 spouse
92           end-if
93           if civil(spouse)<>1
94             msg 'Sec 1.1 The spouse is not married'
95             pcode011(spouse) gender(spouse) civil(spouse)
96           end-if
97           if gender = gender(spouse)
98             msg 'Sec 1.1 Same gender marriage !!!!'
99             pcode011 gender pcode011(spouse) gender(spouse)
100          end-if
101          let agediff = age - age(spouse)
102          if agediff > 15
103            msg 'Sec 1.1 Age difference too large'
104            pcode011 pcode011(spouse)
105          end-if
106        end-if
107      end-if
108    end-if
109  end-if
110
```

Dictionary: ALBANIA

Program: ALBANIA

```
111 .sec 1.1 err04 - Check age difference between head and their children
112   if rec-number >0
113     let agediff = age(1) - age
114     if relation = 3 and agediff < 16
115       msg 'Sec 1.1 Age difference is too small'
116       pcode011 age(1) age
117     end-if
118   end-if
119
```

Dictionary: ALBANIA

Program: ALBANIA

```
120 ..*****
121 . ** Section 1.2 MIGRATED *****
122 . *****
123 For-each (sec012)
124
125 .sec 1.2 - err01 - Check if Members exists
126     if pcode012 = blank OR pcode012 = 0
127         msg 'Sec 1.2 This person does not exist '
128             pcode012
129     end-if
130
131 .sec 1.2 - err02 - Check if Person has not been entered twice
132     if type-count(sec012)>0
133         DO varying prev from 1 by 1
134             until prev = rec-number
135                 if pcode012 = pcode012(prev)
136                     msg 'Sec 1.2 This person is reported more than once'
137                         pcode012 pcode012(prev)
138                 end-if
139             END-DO
140         end-if
141
142
```

Dictionary: ALBANIA

Program: ALBANIA

```
143 ..*****
144 . ** Section 2. EDUCATION *****
145 . *****
146 For-each (sec02)
147
148 .sec 2 - err01 - Check if Members exists
149     if pcode020 = blank OR pcode020 = 0 OR pcode020 >ind
150         msg 'Sec 2 This person does not exist in Sec 1.1'
151         pcode020 ind
152     end-if
153
154 .sec 2 - err02 - Make sure that all individuals have been entered
155     let flag=0
156     if type-count(sec02) <> ind or pcode020 <> rec-number
157         let flag=flag + 1
158     end-if
159     if flag>=1
160         msg "Sec 2 Individuals have not been reported in the right order"
161         rec-number pcode020 flag
162     end-if
163
164
```

Dictionary: ALBANIA

Program: ALBANIA

```
165 ..*****
166 . ** Section 3.1 EMPLOYMENT *****
167 . *****
168 For-each (sec031)
169
170 .sec 3.1 - err01 - Check if Members exist and that are of the right age
171     if pcode031 = blank OR pcode031 = 0 OR pcode031 >ind
172         msg 'Sec 3.1 This person does not exist in Sec 1.1'
173         pcode031 ind
174     else
175         if age(pcode031) <= 15
176             msg 'Sec 3.1 This person is too young'
177             pcode031 age(pcode031)
178         end-if
179     end-if
180
181 .sec 3.1 - err02 - Check if Member is not a duplicate
182     if type-count(sec031)>0
183         DO varying prev from 1 by 1
184             until prev = rec-number
185                 if pcode031 = pcode031(prev)
186                     msg 'Sec 3.1 This person has been entered more than once'
187                     pcode031 pcode031(prev)
188                 end-if
189             END-DO
190         end-if
191
192 .sec 3.1 - err03 - Check if has job and is working
193     if working7 = 1 and notwork <> Blank
194         msg 'Sec 3.1 Is this person working or not ?'
195         pcode031 working7 notwork
196     end-if
197
198
```



Dictionary: ALBANIA

Program: ALBANIA

```
199 ..*****
200 . ** Section 3.2 UNEMPLOYMENT *****
201 . *****
202 For-each (sec032)
203
204 .sec 3.2 err01 - Check if Members exist and that are of the right age
205     if pcode032 = blank OR pcode032 = 0 OR pcode032 >ind
206         msg 'Sec 3.2 This person does not exist in Sec 1.1'
207         pcode032 ind
208     else
209         if age(pcode032) <= 15
210             msg 'Sec 3.2 This person is too young'
211             pcode032 age(pcode032)
212         end-if
213     end-if
214
215 .sec 3.2 - err02 - Check if Member is Looking for a job in 3.1
216     if pcode032 <= type-count(sec031)
217         if lookwork(pcode032)=0 and lookjobm <> BLANK
218             msg 'Sec 3.2 This person is not looking for a job in Sec 3.1'
219             pcode032 lookwork(pcode032)
220         end-if
221     end-if
222
223 .sec 3.2 - err03 - Check if Member is not a duplicate
224     if type-count(sec032)>0
225         DO varying prev from 1 by 1
226             until prev = rec-number
227                 if pcode032 = pcode032(prev)
228                     msg 'Sec 3.2 This person has been entered more than once'
229                     pcode032 pcode032(prev)
230                 end-if
231             END-DO
232         end-if
233
234
```

Dictionary: ALBANIA

Program: ALBANIA

```
235 ..*****
236 . ** Section 3.3 HYSTORY of EMPLOYMENT *****
237 . *****
238 For-each (sec033)
239
240 .sec 3.3 err01 - Check if Members exist and that are of the right age
241     if pcode033 = blank OR pcode033 = 0 OR pcode033 >ind
242         msg 'Sec 3.3 This person does not exist in Sec 1.1'
243         pcode033 ind
244     else
245         if age(pcode033) <= 15
246             msg 'Sec 3.3 This person is too young'
247             pcode033 age(pcode033)
248         end-if
249     end-if
250
251 .sec 3.3 - err02 - Check if Member is not a duplicate
252     if type-count(sec033)>0
253         DO varying prev from 1 by 1
254             until prev = rec-number
255                 if pcode033 = pcode033(prev)
256                     msg 'Sec 3.3 This person has been resported more than once'
257                     pcode033 pcode033(prev)
258                 end-if
259             END-DO
260         end-if
261
262 .sec 3.3 - err03 - Check if Changed Job or Activity last 5 years
263     if pcode033 <= type-count(sec031)
264         if difjobel = 0 and worktime(pcode033) <60
265             msg "Sec 3.3 The activity in Sec 3.1 lasted less than 5 years"
266             pcode033 worktime(pcode033)
267         end-if
268     end-if
269
```

Dictionary: ALBANIA

Program: ALBANIA

```
270 ..*****
271 . **      Section 3.4 A - PUBLIC WORKS      *****
272 . *****
273 For-each (sec034a)
274
275 .sec 3.4a err01 - Check if Members exist and that are of the right age
276     if pcode034 = blank OR pcode034 = 0 OR pcode034 >ind
277         msg 'Sec 3.4a This person does not exist in Sec 1.1'
278         pcode034 ind
279     else
280         if age(pcode034) <= 15
281             msg 'Sec 3.4b This person is too young'
282             pcode034 age(pcode034)
283         end-if
284     end-if
285
286 .sec 3.b4a - err02 - Check if Member is not a duplicate
287     if type-count(sec034a)>0
288         DO varying prev from 1 by 1
289             until prev = rec-number
290                 if pcode034 = pcode034(prev)
291                     msg 'Sec 3.4a This person has been reported more than once'
292                     pcode034 pcode034(prev)
293                 end-if
294             END-DO
295         end-if
296
```

Dictionary: ALBANIA

Program: ALBANIA

```
297 ..*****
298 . ** Section 3.4 B - TRAINING *****
299 . *****
300 For-each (sec034b)
301
302 .sec 3.4b err01 - Check if Members exist and that are of the right age
303     if pcode035 = blank OR pcode035 = 0 OR pcode035 >ind
304         msg 'Sec 3.4b This person does not exist in Sec 1.1'
305         pcode035 ind
306     else
307         if age(pcode035) <= 15
308             msg 'Sec 3.4b This person is too young'
309             pcode035 age(pcode035)
310         end-if
311     end-if
312
313 .sec 3.b4b - err02 - Check if Member is not a duplicate
314     if type-count(sec034b)>0
315         DO varying prev from 1 by 1
316             until prev = rec-number
317                 if pcode035 = pcode035(prev)
318                     msg 'Sec 3.4b This person has been reported more than once'
319                     pcode035 pcode035(prev)
320                 end-if
321             END-DO
322         end-if
323
324
```

Dictionary: ALBANIA

Program: ALBANIA

```
325 ..*****
326 . ** Section 4.1 MAIN JOB *****
327 . *****
328 For-each (sec041)
329
330 .sec 4.1 err01 - Check if Members exist and that are of the right age
331     if pcode041 = blank OR pcode041 = 0 OR pcode041 >ind
332         msg 'Sec 4.1 This person does not exist in Sec 1.1'
333         pcode041 ind
334     else
335         if age(pcode041) <= 15
336             msg 'Sec 4.1 This person is too young'
337             pcode041 age(pcode041)
338         end-if
339     end-if
340
341 .sec 4.1 - err02 - Check if Member is Looking for a job in 3.1
342     if pcode041 <= type-count(sec031)
343         if mainwork(pcode041) = 4:6 or mainwork(pcode041)=BLANK
344             msg 'Sec 4.1 This person has no dependent job in Sec 3.1'
345             pcode041 mainwork(pcode041)
346         end-if
347     end-if
348
349 .sec 4.1 - err03 - Check if Member is not a duplicate
350     if type-count(sec041)>0
351         DO varying prev from 1 by 1
352             until prev = rec-number
353                 if pcode041 = pcode041(prev)
354                     msg 'Sec 4.1 This person has been reported more than once'
355                     pcode041 pcode041(prev)
356                 end-if
357             END-DO
358         end-if
359
360 .sec 4.1 - err04 - Check wage levels
361     if wlpaytim = Blank or wlnetpay = Blank
362         msg "Sec 4.1 Wage variables are missing"
363         pcode041 wlpaytim wlnetpay
364     else
365         if ( wlpaytim = 1 and (wlnetpay < 50 or wlnetpay > 500 )) OR
366            ( wlpaytim = 2 and (wlnetpay < 100 or wlnetpay > 2000 )) OR
367            ( wlpaytim = 3 and (wlnetpay < 500 or wlnetpay > 3000 )) OR
368            ( wlpaytim = 4 and (wlnetpay <1000 or wlnetpay > 5000 )) OR
369            ( wlpaytim = 5 and (wlnetpay <3000 or wlnetpay >10000 ))
370             msg "Sec 4.1 Wage rate seems to high or too low"
371             pcode041 wlpaytim wlnetpay
372         end-if
373     end-if
374
375
```

Dictionary: ALBANIA

Program: ALBANIA

```
376 ..*****
377 . ** Section 4.2 SECOND JOB *****
378 . *****
379 For-each (sec042)
380
381 .sec 4.2 err01 - Check if Members exist and that are of the right age
382     if pcode042 = blank OR pcode042 = 0 OR pcode042 >ind
383         msg 'Sec 4.2 This person does not exist in Sec 1.1'
384         pcode042 ind
385     else
386         if age(pcode042) <= 15
387             msg 'Sec 4.2 This person is too young'
388             pcode042 age(pcode042)
389         end-if
390     end-if
391
392 .sec 4.2 - err02 - Check if Member has a job in 3.1
393     if pcode042 <= type-count(sec031)
394         if mainwork(pcode042) = 4:6 or mainwork(pcode042)=BLANK
395             msg 'Sec 4.2 This person has no dependent job in Sec 3.1'
396             pcode042 mainwork(pcode042)
397         end-if
398     end-if
399
400 .sec 4.2 - err03 - Check if Member is not a duplicate
401     if type-count(sec042)>0
402         DO varying prev from 1 by 1
403             until prev = rec-number
404                 if pcode042 = pcode042(prev)
405                     msg 'Sec 4.2 This person has been reported more than once'
406                     pcode042 pcode042(prev)
407                 end-if
408             END-DO
409         end-if
410
411 .sec 4.2 - err04 - Check wage levels
412     if w2paytim = Blank or w2netpay = Blank
413         msg "Sec 4.2 Wage Variables are missing"
414         pcode042 w2paytim w2netpay
415     else
416         if ( w2paytim = 1 and (w2netpay < 50 or w2netpay > 500 )) OR
417            ( w2paytim = 2 and (w2netpay < 100 or w2netpay > 2000 )) OR
418            ( w2paytim = 3 and (w2netpay < 500 or w2netpay > 3000 )) OR
419            ( w2paytim = 4 and (w2netpay <1000 or w2netpay > 5000 )) OR
420            ( w2paytim = 5 and (w2netpay <3000 or w2netpay >10000 ))
421             msg "Sec 4.2 Wage rate seems to high or too low"
422             pcode042 w2paytim w2netpay
423         end-if
424     end-if
425
426
```

Dictionary: ALBANIA

Program: ALBANIA

```
427 ..*****
428 . ** Section 5.1 BUSINESS ACTIVITY ****
429 . *****
430 For-each (sec051)
431
432 .sec 5.1 err01 - Check if Members exist and that are of the right age
433     if pcode051 = blank OR pcode051 = 0 OR pcode051 >ind
434         msg 'Sec 5.1 This person does not exist in Sec 1.1'
435         pcode051 ind
436     else
437         if age(pcode051) <= 15
438             msg 'Sec 5.1 This person is too young'
439             pcode051 age(pcode051)
440         end-if
441     end-if
442
443 .sec 5.1 - err02 - Check if Member has a Business in 3.1
444     if pcode051 <= type-count(sec031)
445         if mainwork(pcode051) <> 4
446             msg 'Sec 5.1 This person does not have a business in Sec 3.1'
447             pcode051 mainwork(pcode051)
448         end-if
449     end-if
450
451 .sec 5.1 - err03 - Check if Member is not a duplicate
452     if type-count(sec051)>0
453         DO varying prev from 1 by 1
454             until prev = rec-number
455                 if pcode051 = pcode051(prev)
456                     msg 'Sec 5.1 This person has been reported more than once'
457                     pcode051 pcode051(prev)
458                 end-if
459             END-DO
460         end-if
461
462
```

Dictionary: ALBANIA

Program: ALBANIA

```
463 ..*****
464 . ** Section 6.1 AGRICULTURE *****
465 . *****
466 For-each (sec061)
467
468 .sec 6.1 - err01 - Check if Members exist and that are of the right age
469     if pcode061 = blank OR pcode061 = 0 OR pcode061 >ind
470         msg 'Sec 6.1 This person does not exist in Sec 1.1'
471         pcode061 ind
472     else
473         if age(pcode061) <= 15
474             msg 'Sec 6.1 This person is too young'
475             pcode061 age(pcode061)
476         end-if
477     end-if
478
479 .sec 6.1 - err02 - Check if Member is not a duplicate
480     if type-count(sec061)>0
481         DO varying prev from 1 by 1
482             until prev = rec-number
483                 if pcode061 = pcode061(prev)
484                     msg 'Sec 6.1 This person has been reported more than once'
485                     pcode061 pcode061(prev)
486                 end-if
487             END-DO
488         end-if
489
490 .sec 6.1 - err03 - Check if has agriculture in 3.1 and is reported here
491     if pcode061 <= type-count(sec031)
492         if mainwork(pcode061) = 5 and agritime <> 1
493             msg 'Sec 6.1 This person reports agricultural activity in Sec 3.1'
494             pcode061 mainwork(pcode061) agritime
495         end-if
496     end-if
497
```



Dictionary: ALBANIA

Program: ALBANIA

```
498 ..*****
499 . ** Section 6.2B AGRICULTURE LAND ****
500 . *****
501 For-each (sec062b)
502
503 .Sec 6.2b - err01 - Check the value of Land Owned
504
505     DO varying prev from 1 by 1
506         until prev > 3
507         if alowna(# prev) > 0 and alownv(# prev) = Blank
508             msg "Sec 6.2b What is the value of the land? "
509         else
510             if alownv(# prev) > 0
511                 let valdenim = alownv(# prev)/alowna(# prev)
512                 if valdenim < 200 or valdenim > 1900
513                     msg "Sec 6.2b Land values seems too high or too low"
514                     alownv(# prev)
515                 end-if
516             end-if
517         end-if
518     END-DO
519
520
```

Dictionary: ALBANIA

Program: ALBANIA

```
521 ..*****
522 . **      Section 7.1b REMITTANCES SENT ****
523 . *****
524 For-each (sec071b)
525
526 .sec 7.1b - err01 - Check if Entry is not a duplicate
527     if type-count(sec071b)>0
528         DO varying prev from 1 by 1
529             until prev = rec-number
530             if percontr = percontr(prev)
531                 msg 'Sec 7.1b This person has been reported more than once'
532                 percontr percontr(prev)
533             end-if
534         END-DO
535     end-if
536
537 .sec 7.1b - err02 - Check if Individual exists in Sec 1.1 or 1.2
538     if rrcode <> Blank and rrcode < 30 and rrcode > ind
539         msg 'Sec 7.1b This person does not exist in Sec 1.1'
540         rrcode ind
541     end-if
542     if rrcode <> Blank and rrcode > 30 and rrcode < 40
543         if type-count(sec012)>0
544             let flag = rrcode - 30
545             if flag > type-count(sec012)
546                 msg 'Sec 7.1b This person does not exist in Sec 1.2'
547                 rrcode
548             end-if
549         else
550             msg 'Sec 7.1b Should have Section 1.2'
551         end-if
552     end-if
553
554
```

Dictionary: ALBANIA

Program: ALBANIA

```
555 ..*****
556 . ** Section 7.2b REMITTANCES RECEV ***
557 . *****
558 For-each (sec072b)
559
560 .sec 7.2b - err01 - Check if Entry is not a duplicate
561     if type-count(sec072b)>0
562         DO varying prev from 1 by 1
563             until prev = rec-number
564             if perrec = perrec(prev)
565                 msg 'Sec 7.2b This person has been reported more than once'
566                 perrec perrec(prev)
567             end-if
568         END-DO
569     end-if
570
571 .sec 7.2b - err02 - Check if Individual exists in Sec 1.1 or 1.2
572     if rrcode <> Blank and rrcode < 30 and rrcode > ind
573         msg 'Sec 7.2b This person does not exist in Sec 1.1'
574         rrcode ind
575     end-if
576     if rrcode <> Blank and rrcode > 30 and rrcode < 40
577         if type-count(sec012)>0
578             let flag = rrcode - 30
579             if flag > type-count(sec012)
580                 msg 'Sec 7.2b This person does not exist in Sec 1.2'
581                 rrcode
582             end-if
583         else
584             msg 'Sec 7.2b Should have Section 1.2'
585         end-if
586     end-if
587
588
```

Dictionary: ALBANIA

Program: ALBANIA

```
589 ..*****
590 . **      Section 8.1 SOCIAL ASSISTANCE ****
591 . *****
592 For-each (sec081)
593
594 .sec 8.1 - err01 - Check if Code is a duplicate
595     if type-count(sec081)>0
596         DO varying prev from 1 by 1
597             until prev = rec-number
598                 if sacode = sacode(prev)
599                     msg 'Sec 8.1 This code has been reported more than once'
600                     sacode sacode(prev)
601                 end-if
602             END-DO
603         end-if
604
605 .sec 8.1 - err02 - Check if it is complete
606     if saany = blank or
607         sa95any = blank or
608         sa94any = blank or
609         sa93any = blank
610         msg 'Sec 8.1 This record is incomplete'
611         sacode
612     end-if
613
614
```

Dictionary: ALBANIA

Program: ALBANIA

```
615 ..*****
616 . ** Section 9.1 HOUSING *****
617 . *****
618 For-each (sec091)
619
620 .sect 9.1 err01
621     if electhrs >5 and mxelect = blank
622         msg "If they have electricity thay must pay for it "
623     end-if
624
625
```

Dictionary: ALBANIA

Program: ALBANIA

```
626 ..*****
627 . ** Section 10 FOOD CONSUMPTION ****
628 . *****
629 For-each (sec10)
630
631 .sec 10 - err01 - Check if Code is a duplicate
632     if type-count(sec10)>0
633         DO varying prev from 1 by 1
634             until prev = rec-number
635             if fooditem = fooditem(prev)
636                 msg 'Sec 10 This food code has been reported more than once'
637                 fooditem fooditem(prev)
638             end-if
639         END-DO
640     end-if
641
642 .sec 10 - err02 - Check if it is complete
643     if fpurqua = Blank and fpurval = Blank
644         msg "Sec 10 Quantity or values are missing"
645         fooditem
646     else
647         if fdprod = Blank or fdreceiv = Blank
648             msg 'Sec 10 Quantity produced or received are missing'
649             fooditem
650         end-if
651     end-if
652
653 .sec 10 - err03 - Check if prices have been reported
654     if fpurval > 0 and (fdprice = Blank or fdprice = 0)
655         msg "Sec 10 Price is missing"
656         fooditem
657     end-if
658
659 .sec 10 - err04 - Check prices
660     if fdprice <> Blank and fdprice > 0
661         if ( fooditem = 101:107 and (fdprice < 20 or fdprice > 99)) OR
662             ( fooditem = 201:202 and (fdprice < 20 or fdprice > 99)) OR
663             ( fooditem = 301:306 and (fdprice < 10 or fdprice > 50)) OR
664             ( fooditem = 406:411 and (fdprice < 10 or fdprice > 99)) OR
665             ( fooditem = 501:504 and (fdprice < 50 or fdprice >150)) OR
666             ( fooditem = 601:605 and (fdprice < 100 or fdprice >400)) OR
667             ( fooditem = 606 and (fdprice < 5 or fdprice > 35)) OR
668             ( fooditem = 608:703 and (fdprice < 100 or fdprice >400)) OR
669             ( fooditem = 801:806 and (fdprice < 20 or fdprice >400)) OR
670             ( fooditem = 201:202 and (fdprice < 20 or fdprice > 99)) OR
671             ( fooditem = 901:906 and (fdprice < 20 or fdprice >400)) OR
672             ( fooditem =1102:1103 and (fdprice < 9 or fdprice > 99)) OR
673             ( fooditem =1104:1105 and (fdprice < 100 or fdprice >1000)) OR
674             ( fooditem =1201:1206 and (fdprice < 20 or fdprice >300))
675             msg "Sec 10 Prices are too high or too low"
676             fooditem fdprice
677         end-if
678     end-if
679
680
```

Dictionary: ALBANIA

Program: ALBANIA

```
681 ..*****
682 . ** Section 12.1 HEALTH - CHRONIC ****
683 . *****
684 For-each (sec121)
685
686 .sec 12.1 - err01 - Check if Members exist in Sec 1.1
687     if pcode121 = blank OR pcode121 = 0 OR pcode121 >ind
688         msg 'Sec 12.1 This person does not exist in Sec 1.1'
689         pcode121 ind
690     end-if
691
692 .sec 12.1 - err02 - Check if Member is not a duplicate
693     if type-count(sec121)>0
694         DO varying prev from 1 by 1
695             until prev = rec-number
696                 if pcode121 = pcode121(prev)
697                     msg 'Sec 12.1 This person has been reported more than once'
698                     pcode121 pcode121(prev)
699                 end-if
700             END-DO
701         end-if
702
```

Dictionary: ALBANIA

Program: ALBANIA

```
703 ..*****
704 . ** Section 12.2 HEALTH - ACUTE *****
705 . *****
706 For-each (sec122)
707
708 .sec 12.2 - err01 - Check if Members exist in Sec 1.1
709     if pcode122 = blank OR pcode122 = 0 OR pcode122 >ind
710         msg 'Sec 12.2 This person does not exist in Sec 1.1'
711         pcode122 ind
712     end-if
713
714 .sec 12.2 - err02 - Check if Member is not a duplicate
715     if type-count(sec122)>0
716         DO varying prev from 1 by 1
717             until prev = rec-number
718                 if pcode122 = pcode122(prev)
719                     msg 'Sec 12.2 This person has been reported more than once'
720                     pcode122 pcode122(prev)
721                 end-if
722             END-DO
723         end-if
724
725
```



Dictionary: ALBANIA

Program: ALBANIA

```
726 ..*****
727 . *** Across Sections *****
728 . *****
729 at-end-quest
730
731 . err00 - If the questionnaire was a refusal do not check it
732     if nointerv(1) = 1:3 or nointerv(1) = 5
733     msg 'Sec 00 - The questionnaire is incomplete'
734     stop-quest
735     end-if
736
737
738 .*****
739 . err01 - Make sure that the required individuals are Reported
740
741 . Sec 3.1 - Employment
742 if type-count(sec031)>0
743     DO varying pnum from 1 by 1
744         until pnum > ind
745         if age(pnum) >15
746             let miss031=0
747             DO varying ind031 from 1 by 1
748                 until ind031 > type-count(sec031)
749                 if pcode011(pnum) = pcode031(ind031)
750                     let miss031=1
751                 end-if
752             END-DO
753             if miss031<>1
754                 msg 'Sec 3.1 - This person should be reported in this section'
755                 pcode011(pnum)
756             end-if
757         end-if
758     END-DO
759 end-if
760
761 .Sec 3.2 - Unemployment - People looking for job in sec 3.1
762 if type-count(sec032)>0
763     DO varying ind031 from 1 by 1
764         until ind031 > type-count(sec031)
765         if lookwork(ind031)=1
766             let miss032=0
767             DO varying ind032 from 1 by 1
768                 until ind032 > type-count(sec032)
769                 if pcode031(ind031) = pcode032(ind032)
770                     let miss032=1
771                 end-if
772             END-DO
773             if miss032<>1
774                 msg 'Sec 3.2 - This person should be reported in this section'
775                 pcode031(ind031)
776             end-if
777         end-if
778     END-DO
779 end-if
780
781 . Sec 3.3 - Hystory of Employment - All people age 16 and up
782 if type-count(sec033)>0
783     DO varying pnum from 1 by 1
```

Dictionary: ALBANIA

Program: ALBANIA

```
784         until pnum > ind
785     if age(pnum) >15
786         let miss033=0
787         DO varying ind033 from 1 by 1
788             until ind033 > type-count(sec033)
789             if pcode011(pnum) = pcode033(ind033)
790                 let miss033=1
791             end-if
792         END-DO
793         if miss033<>1
794             msg 'Sec 3.3 - This person should be reported in this section'
795             pcode011(pnum)
796         end-if
797     end-if
798 END-DO
799 end-if
800
801 .Sec 4.1 - Main Dependent Job - People with a job in sec 3.1
802 if type-count(sec041)>0
803     DO varying ind031 from 1 by 1
804         until ind031 > type-count(sec031)
805         if mainwork(ind031)=1:3
806             let miss041=0
807             DO varying ind041 from 1 by 1
808                 until ind041 > type-count(sec041)
809                 if pcode031(ind031) = pcode041(ind041)
810                     let miss041=1
811                 end-if
812             END-DO
813             if miss041<>1
814                 msg 'Sec 4.1 - This person should be reported in this section'
815                 pcode031(ind031)
816             end-if
817         end-if
818     END-DO
819 end-if
820
821 .Sec 5.1 - Business activity - People with a job in sec 3.1
822 if type-count(sec051)>0
823     DO varying ind031 from 1 by 1
824         until ind031 > type-count(sec031)
825         if mainwork(ind031)=4
826             let miss051=0
827             DO varying ind051 from 1 by 1
828                 until ind051 > type-count(sec051)
829                 if pcode031(ind031) = pcode051(ind051)
830                     let miss051=1
831                 end-if
832             END-DO
833             if miss051<>1
834                 msg 'Sec 5.1 - This person should be reported in this section'
835                 pcode031(ind031)
836             end-if
837         end-if
838     END-DO
839 end-if
840
841 . Sec 6.1 - Agricultural Activity
```

Dictionary: ALBANIA

Program: ALBANIA

```
842  if type-count(sec061)>0
843      DO varying pnum from 1 by 1
844          until pnum > ind
845          if age(pnum) >15
846              let miss061=0
847              DO varying ind061 from 1 by 1
848                  until ind061 > type-count(sec061)
849                  if pcode011(pnum) = pcode061(ind061)
850                      let miss061=1
851                  end-if
852              END-DO
853          if miss061<>1
854              msg 'Sec 6.1 - This person should be reported in this section'
855              pcode011(pnum)
856          end-if
857      end-if
858  END-DO
859 end-if
860
861
862 .*****
863 . err02 - Make sure that the required records exists
864
865 .Sec 1.1
866     if type-count(sec011) = 0
867         msg 'Sec 1.1 Not completed ?'
868     end-if
869
870 .Sec 2
871     if type-count(sec02) = 0
872         msg 'Sec 2 Not completed ?'
873     end-if
874
875 .Sec 3.1
876     if type-count(sec031) = 0
877         msg 'Sec 3.1 Not completed ?'
878     end-if
879
880 .Sec 3.2
881     DO varying pnum from 1 by 1
882         until pnum>type-count(sec031)
883         if lookwork(pnum)=1 and type-count(sec032) = 0
884             msg 'Sec 3.2 Remember to report those looking for a job in Sec 3.2'
885             pcode031(pnum)
886         end-if
887     END-DO
888
889 .Sec 3.3
890     if type-count(sec033) = 0
891         msg 'Sec 3.3 Not completed ?'
892     end-if
893
894 .Sec 4.1
895     DO varying pnum from 1 by 1
896         until pnum>type-count(sec031)
897         if mainwork(pnum)=1:3 and type-count(sec041) = 0
898             msg 'Sec 4.1 Remember to report those who have a job in Sec 4.1'
899             pcode031(pnum)
```

```
900         end-if
901     END-DO
902
903 .Sec 5.1
904     DO varying pnum from 1 by 1
905         until pnum>type-count(sec031)
906         if mainwork(pnum)=4 and type-count(sec051) = 0
907             msg 'Sec 5.1 Remember to report those who have a business in Sec 5.1'
908             pcode031(pnum)
909         end-if
910     END-DO
911
912 .Sec 5.2a
913     if (type-count(sec051) >=1 and type-count(sec052a) = 0) OR
914         (type-count(sec051) >=1 and type-count(sec052b) = 0)
915         msg 'Sec 5.2a o 5.2b Not completed ?'
916     end-if
917
918 .Sec 6.1
919     if type-count(sec061) = 0
920         msg 'Sec 6.1 Not completed ?'
921     end-if
922
923 .Sec 6.2a
924     if type-count(sec062a) = 0
925         msg 'Sec 6.2A Not completed ?'
926     end-if
927
928 .Sec 6.2b and 6.2c
929     if type-count(sec062a) = 1
930         if (accesagl(1) = 1 and type-count(sec062b) = 0) OR
931             (accesagl(1) = 1 and type-count(sec062c) = 0) OR
932             (accesagl(1) = 1 and type-count(sec063) = 0)
933             msg 'Sec 6.2a o 6.2b o 6.3 Not completed ?'
934         end-if
935     end-if
936
937 .Sec 7.1b
938     if type-count(sec071a) = 1
939         if remrecev(1) = 1 and type-count(sec071b) = 0
940             msg 'Sec 7.1b Not completed ?'
941         end-if
942     end-if
943
944 .Sec 7.2b
945     if type-count(sec072a) = 1
946         if remsend(1) = 1 and type-count(sec072b) = 0
947             msg 'Sec 7.2b Not completed ?'
948         end-if
949     end-if
950
951 .Sec 8.1
952     DO varying pnum from 1 by 1
953         until pnum>type-count(sec011)
954         if age(pnum)>60 and type-count(sec081) = 0
955             msg 'Sec 8.1 There should be at least one pensioner in Sec 8.1'
956             pcode011(pnum)
957         end-if
```

Dictionary: ALBANIA

Program: ALBANIA

```
958         END-DO
959
960     .Sec 9.2b
961         if type-count(sec092a) = 1
962             if realasset(1) = 1 and type-count(sec092b) = 0
963                 msg 'Sec 9.2b Not completed ?'
964             end-if
965         end-if
966
967     .Sec 10, 11.1, 11.2
968         if type-count(sec10) = 0 OR
969             type-count(sec111) = 0 OR
970             type-count(sec112) = 0
971             msg 'Sec 10, 11 Not completed ?'
972         end-if
973
974
975     .*****
976     . err03 - Check the quantities purchased of food
977
978     if type-count(sec10) >0 or type-count(sec111) >0
979
980     .Cereals and Potatoes
981         call setfood
982         DO varying pnum from 1 by 1
983             until pnum > type-count(sec10)
984                 if fooditem(pnum) > 100 and fooditem(pnum) <300
985                     call clctfood
986                 end-if
987     END-DO
988         let totfood = (fqpur + fqpro + fqrec)
989     if totfood = 0
990         msg "Sec 10 Do not consume CEREALS !! "
991     else
992         let totfood = totfood / ind
993         if totfood > 27
994             msg "Sec 10 Cereal consumption seems excessive"
995             fqpur fqpro fqrec totfood
996         end-if
997     endif
998
999     .Vegetables, Fruits and Canned Vegetables
1000     call setfood
1001     DO varying pnum from 1 by 1
1002         until pnum > type-count(sec10)
1003             if fooditem(pnum) > 300 and fooditem(pnum) <407 OR
1004                 fooditem(pnum) > 407 and fooditem(pnum) <600
1005                 call clctfood
1006             end-if
1007     END-DO
1008         let totfood = (fqpur + fqpro + fqrec)
1009     if totfood = 0
1010         msg "Sec 10 Do not consume  FRUITS and VEGETABLES !! "
1011     else
1012         let totfood = totfood / ind
1013         if totfood > 35
1014             msg "Sec 10 The consumption of Fruits and Vegetables seems excessive"
1015             fqpur fqpro fqrec totfood
```

```
1016         end-if
1017     endif
1018
1019 .Meat and Fish
1020     call setfood
1021     DO varying pnum from 1 by 1
1022         until pnum > type-count(sec10)
1023         if (fooditem(pnum) > 600 and fooditem(pnum) < 606) OR
1024             (fooditem(pnum) > 607 and fooditem(pnum) < 800)
1025             call clctfood
1026         end-if
1027     END-DO
1028     let totfood = (fqpur + fqpro + fqrec)
1029     if totfood = 0
1030         msg "Sec 10 Do not consume  MEAT or FISH !! "
1031     else
1032         let totfood = totfood / ind
1033         if totfood > 15
1034             msg "Sec 10 The consumption of Meat and Fish seems excessive"
1035             fqpur fqpro fqrec totfood
1036         end-if
1037     endif
1038
1039 .Eggs
1040     call setfood
1041     DO varying pnum from 1 by 1
1042         until pnum > type-count(sec10)
1043         if fooditem(pnum) = 606
1044             call clctfood
1045         end-if
1046     END-DO
1047     let totfood = (fqpur + fqpro + fqrec)
1048     if totfood = 0
1049         msg "Sec 10 Do not consume  EGGS !! "
1050     else
1051         let totfood = totfood / ind
1052         if totfood > 60
1053             msg "Sec 10 The consumption of Eggs seems excessive"
1054             fqpur fqpro fqrec totfood
1055         end-if
1056     endif
1057
1058 .Milk and Cheese
1059     call setfood
1060     DO varying pnum from 1 by 1
1061         until pnum > type-count(sec10)
1062         if fooditem(pnum) > 800 and fooditem(pnum) < 900
1063             call clctfood
1064         end-if
1065     END-DO
1066     let totfood = (fqpur + fqpro + fqrec)
1067     if totfood = 0
1068         msg "Sec 10 Do not consume  DAIRY !! "
1069     else
1070         let totfood = totfood / ind
1071         if totfood > 45
1072             msg "Sec 10 The consumption of Dairy productds seems excessive"
1073             fqpur fqpro fqrec totfood
```

```
1074         end-if
1075     endif
1076
1077 .Fats
1078     call setfood
1079     DO varying pnum from 1 by 1
1080         until pnum > type-count(sec10)
1081         if fooditem(pnum) > 900 and fooditem(pnum)< 1000
1082             call clctfood
1083         end-if
1084     END-DO
1085     let totfood = (fqpur + fqpro + fqrec)
1086     if totfood = 0
1087         msg "Sec 10 Do not consume FATS or OILS !! "
1088     else
1089         let totfood = totfood / ind
1090         if totfood > 12
1091             msg "Sec 10 The consumption of Fats and Oils seems excessive"
1092             fqpur fqpro fqrec totfood
1093         end-if
1094     endif
1095
1096 end-if
1097
1098
```

Dictionary: ALBANIA

Program: ALBANIA

```
1099 . *****
1100 . *****      END OF PROGRAM      *****
1101 . *****
1102
1103     output questionnaire
1104
1105
```



Dictionary: ALBANIA

Program: ALBANIA

```
1106 ..*****
1107 . *** Subroutines *****
1108 . *****
1109
1110 SUBR setfood
1111     let fqpur = 0
1112     let fqpro = 0
1113     let fqrec = 0
1114 END-SUBR
1115
1116 SUBR clctfood
1117     if fpurqua(pnum) <> blank and fpurqua(pnum) >0
1118         let fqpur = fqpur + fpurqua(pnum)/100
1119     end-if
1120     if (fpurqua(pnum) = Blank or fpurqua(pnum) = 0) and
1121         (fpurval(pnum) <> Blank and fpurval(pnum) >0) and
1122         (fdprice(pnum) <> Blank and fdprice(pnum) >0)
1123         let fqpur = fqpur + fpurval(pnum)/fdprice(pnum)
1124     end-if
1125     if fdprod(pnum) <> blank and fdprod(pnum) >0
1126         let fqpro = fqpro + fdprod(pnum)/100
1127     end-if
1128     if fdreceiv(pnum) <> blank and fdreceiv(pnum) >0
1129         let fqrec = fqrec + fdreceiv(pnum)/100
1130     end-if
1131 END-SUBR
1132
1133
```

Number of Source Lines

1133