CLS Cohort Studies

Data Note 2008/1

The BCS70 16-year
Arithmetic Dataset

Brian Dodgeon

Centre for Longitudinal Studies

Centre for Longitudinal Studies
Institute of Education
20 Bedford Way
London WC1H 0AL
Tel: 020 7612 6860
Fax: 020 7612 6880
Email cls@ioe.ac.uk
Web http://www.cls.ioe.ac.uk

A. Introduction

The administration of the BCS70 cohort study was taken over by the Centre for Longitudinal Studies (formerly the Social Statistics Research Unit, City University) at the end of the 1980s. It had previously been housed at Bristol University, who had been involved from the inception of the study in 1970.

Unfortunately, after the move from Bristol, it was found that the data for the sweeps up to and including age 16 were not stored and documented in a systematic way, and certain elements appeared to be missing, or else hidden on a vast mainframe, with no easy means of location.

One such element was the dataset containing the results of the Arithmetic test at age 16. This was part of Document C, the rest of which can be viewed at the CLS website at:

http://www.cls.ioe.ac.uk/studies.asp?section=00010002000200070001

The vocabulary test data and two spelling tests results at age 16 have been available for many years, but the arithmetic, reading and matrices results had not previously been found. However, in 2007 the arithmetic dataset came to light when the contents of the mainframe were copied to a more easily-searchable windows environment.

The dataset was not in an easy-to-use format: it did not contain the longitudinal link variable KEY, but instead had the obsolete individual case identifier from the 'Youthscan' era, known as the y-number (YNUM). Some cases had duplicate y-numbers, indicating the presence of twins, but it was not always straightforward to interpret which twin was which, in order to match against a look-up table of YNUM against KEY. The test item results were keyed in a form where one could not tell whether each answer was right or wrong, and a large number of cases on the dataset had no valid test data at all.

This Data Note explains the process of cleaning up the dataset and deriving variables which facilitate its analysis.

B. Step-by-Step process of cleaning data

The list of 60 questions which form the arithmetic test appears in Appendix 1. Each question has a format of five possible answers A-E, only one of which is correct.

The raw dataset contained the individual item responses to the test, each stored in a 5-character string variable. These were named 'car1' to 'car60'. If the respondent chose answer 'A', this would be coded as an 'A' followed by four blanks; if 'B' it was coded with 'B' in the second position, with blanks in the other four positions, and so on. If the question was not attempted, the variable was coded as 5 blank spaces.

Surprisingly, four of these 5-character string variables had been 'split' into two variables: a 2-character string which could contain an 'A' or a 'B'; and a 3-character variable which could contain a 'C', 'D', or 'E' (or blanks of course). These variables were named car7.1 and car7.2, car23.1 and car23.2, car39.1 and car39.2, and car55.1 and car55.2. As they occur in a regular sequence, each sixteen numbers further on from the previous one, it seems likely this split is a consequence of the data being stored on 80-character punched cards in the original mainframe environment (16x5=80).

So there were 64 variables denoting the item responses, and two variables enabling the cohort members to be longitudinally linked: 'yno' denoting the y-number, and 'tc' denoting the 'twin' code, (distinguishing twins, which always have identical y-numbers).

Step 1: Eliminate cases where no data recorded

The dataset contained 6,351 cases, but it was obvious that many cohort members had blanks for every single question, i.e. they hadn't attempted the test.

'Eyeballing' the data, these cases were not easy to distinguish from those who had attempted only a minimal number of questions: against expectations, some cohort members had answered one or two questions late on (e.g. question 40), but none of the earlier ones. So to eliminate those with no data, it was necessary to test every one of the 64 character strings for consistent blanks (see syntax in Appendix 2).

2,543 cases contained no data at all, so after deleting these we were left with 3,808 valid cases.

Step 2: Construct y-number & twin code combination, and eliminate duplicates

Concatenating the y-number and twin code variables, we would hope to arrive at a unique case identifier, which could then be matched against a corresponding key number, enabling longitudinal linkage.

Although every case had a valid twin code (tc) to go with the y-number, there were 25 y-number/twin-code combinations which were duplicated in the dataset. One would expect the y-numbers to be duplicated in the case of twins, where the tc variable resolves which twin is which. Unfortunately, in each of these 25 cases the tc variable was coded zero.

Looking up the corresponding KEY numbers, it seems unlikely these 25 pairs of cases correspond to twins (where a twin code might have been erroneously typed): if so, one would expect the KEY number to end in a 1 or 2, but in all cases they ended in zero, and the 'CHESno' part of the KEY number (i.e. everything except the last digit) is unique for all 25 cases. All 25 were present at the age 16 survey (i.e. they have data for the other age 16 questionnaires), but there is no record of a twin for any of those KEY numbers.

Eyeballing the dataset, it is clear that the arithmetic scores themselves are not, in general, duplicated: with two exceptions, there are two distinct sets of results for each pair. One explanation might be that there was a series of typo errors in entering the y-numbers, and that in each of these 25 pairs of cases, one of them should have been typed as a different number. But if so, one might expect one of the adjacent numbers to be missing, but they're not.

Another possibility is that these 25 cohort members somehow did the arithmetic test twice, and these duplicates represent two sets of results for the same person. But in the absence of any further evidence, we are unable to discern which of the two is the more valid. So regrettably, 23 pairs (i.e. 46 records) had to be discarded. In the case of the other two, the item scores were identical, so we could arbitrarily delete one member of each pair, and leave the other.

Step 3: Match y-number/twin-code variable against longitudinal link variable KEY

The look-up table derived from the confidential BCS70 Address Database enables a unique match to be made between ynum_tc and KEY in most cases. However, there are problems: not every y-number is present in that look-up. It contains 16,396 cases out of the total of 18,736 who have ever been the subject of an interview at any sweep. As the address database was constructed some years later than the 16-year survey, those missing from the look-up tend to be cohort members who dropped out of the study after the 16-year sweep.

There were 90 yno_tc combinations in the dataset which did not appear in that exact form in the look-up table. Of these, 21 appeared to be cases of twins where (unlike those referred to in Step 2) only one was present in the dataset. In all 21 cases, there was a match on the y-number in the look-up table, but the twin code was discrepant. In 6 of these 21 cases there was only one twin present in the look-up table, so despite the discrepant twin code, it was assumed we had a match, based on the y-number.

In the remaining 15 cases of twins, there were two records with that y-number in the look-up: usually the twin code in the dataset was zero, but the two twin codes in the look-up table were 1 & 2. In all these cases a check was carried out on the two corresponding KEY numbers, to see if perhaps only one of the twins was present at the 16-year survey. This occurred in just one case, so there it could be reasonably assumed it was this twin who did the arithmetic test, and a match could be made; but in the

other 14 cases, both twins were present at age 16, so (as with those referred to in Step 2) there was no way of knowing which it was, and those cases had to be discarded.

In the remaining 69 cases, there was not even a corresponding y-number element in the look-up table (i.e. they were not present in the address database: probably dropping out of the survey after the 16-year survey), so no match was possible.

So a total of 83 cases had to be discarded, leaving us with a dataset containing 3,677 cases.

Step 4: Check that all cases were present at age 16 survey

Now that each of the remaining 3,677 cases had a unique KEY number, it was possible to do a check against the 'presence' file which indicates which cases were present at which BCS70 sweep.

All but one were found to have been present at the age 16 sweep, which was a reassuring endorsement that the process of linkage to KEY number was correct.

In the case of the one case not present, a check was performed which showed they were present at every other sweep up to age 34. So it doesn't seem to be such an anomalous case: it is plausible that this cohort member did the arithmetic test, but was absent for the home visit and other tests at age 16.

Step 5: Derive variables summarising results of test

In response to each of the 60 arithmetic questions, the cohort members were asked to choose between 5 possible answers, denoted by alternatives A-D. The variables corresponding to these responses are 'car1' to 'car60'.

These are stored as 5-character strings, with 'A' always appearing as the first character followed by four blanks, 'B' as the second character followed by three blanks, etc.

It was necessary to derive a corresponding series of 60 variables coded 1 for 'correct' and 0 for 'incorrect'. These were then totalled in a variable 'mathscore', and a corresponding variable 'mathincorrect' added up the incorrect responses.

Normally only one character had been keyed in each item score, but for about 2-3% of cases, two or three characters appear, which seems to indicate the student 'hedged their bets'. All these multiple answers were coded as 'incorrect'.

An additional complication was that, for questions 7, 23, 39 and 55 only, the 5-character string is broken into two variables, a 2-character string for the 'A' and 'B' possible answers (car7.1 etc) and a 3-char string for the 'C', 'D' and 'E' responses (e.g. car7.2). This is presumably because, back in 1986 when the test was performed, the data would have been stored on 80-character punched cards.

They were therefore coded as correct if the right answer appeared in the appropriate part and the other part was blank (e.g. for q7 the correct answer is 'A', so we look for car7.1 to be 'A '. car7.2 can only contain 'C', 'D' and 'E', and should therefore be blank).

It was noticeable that very few people got correct answers to questions 48-50. This appears to be caused by the fact that the histogram they're asked to analyse has no labels on the X axis, making it very difficult to fathom the question!

Appendix 1 The Arithmetic Test Booklet

ARITHMETIC TEST

YOU WILL BE TOLD WHEN TO START – please read the instructions on this page first.

INSTRUCTIONS

This is a test of arithmetic. You can work out the answers in your head or use spare paper for any rough work.

Please remember to enter your choice of answer on the separate Student Score Form.

Please look first at the two examples shown below. You will see that for each question there are five answers to choose from. Only one answer is correct.

1st example

	(a)	(b)	(c)	(d)	(e)
$2 \times 2 =$	1	2	3	4	5

The correct answer is 4, so the **d** space has been filled in for you on the Score Form opposite "1st example". Look at the answer sheet and check this.

Now proceed to do the next example for yourself.

2nd example

	(a)	(b)	(c)	(d)	(e)
$4 \div 4 =$	2	4	8	10	15

You should have filled in space **c** opposite the "2nd example" on the Score Form, since the correct answer is 8 in column **c**.

There are 60 questions in this test. Try to answer as many of them as you can. If you cannot answer a question don't waste too much time on it but go on to the next one. When you have finished one page, go on to the next page. There are three pages of questions altogether. You have 30 minutes to do this test. If you finish before then, you can go back over your answers to check them.

DO NOT TURN OVER THIS PAGE UNTIL YOU ARE TOLD TO DO SO

		(a)	(b)	(c)	(d)	(e)
1.	$2 + 3 =$	1/5	3	4	5	7
2.	$2 \times 4 =$	6	8	12	16	24
3.	$12 \div 3 =$	4	6	9	12	36
4.	$359 - 126 =$	103	113	133	213	233
5.	$57 + 135 =$	78	182	192	642	705
6.	What number multiplied by itself gives 81?	7	8	9	10	11
7.	Add half of 26 to twice 24	61	74	100	101	124
8.	Choose in figures "fourteen hundred and three"	103	143	403	1403	1430
9.	$1.85 - 0.45 =$	1·2	1·4	1·45	1·65	2·3
10.	$4 \times 6 \div 3 =$	3	4	5	6	8
11.	$44 \times 11 =$	55	88	444	484	584
12.	Subtract a quarter of 12 from half of 12	3	4	5	6	8
13.	$27.85 + 15.32 =$	33·17	43·17	43·27	44·17	44·77
14.	$196 \div 14 =$	6	8	10	12	14
15.	$6 \times 0.33 =$	1·88	1·98	2	2·05	2·19
16.	$1.25 + 0.875 =$	1·875	1·9	2·125	2·215	2·225
17.	What is the square root of 36?	3	4	6	12	16
18.	What is the average of 2, 4 and 12?	6	7	8	9	10
19.	What is the average of 43, 37, 125 and 35?	30	35	40	45	60
20.	**25, 24, 22, 19?** What number comes next?	16	15	14	13	12
21.	$12^2 =$	24	122	144	148	156
22.	$2^4 =$	48	32	24	16	8
23.	$4^3 =$	12	43	56	64	143

24-26. A letter is to be chosen at random from the word "ARITHMETIC".
         What is the probability that it will be:

		(a)	(b)	(c)	(d)	(e)
24.	The letter 'R'?.	0·1 or 1/10	0·2 or 1/5	0·3 or 3/10	0·4 or 2/5	0·5 or 1/2
25.	The letter 'T'?.	0·1 or 1/10	0·12 or 3/25	0·15 or 3/20	0·18 or 9/50	0·2 or 1/5
26.	A vowel?.	0·2 or 1/5	0·4 or 2/5	0·5 or 1/2	0·6 or 3/5	1·0 or 1

		(a)	(b)	(c)	(d)	(e)
27.	20% of 50 =.	10	15	20	25	30
28.	$0.5 \times 0.25 =$.	0·105	0·115	0·125	0·75	1·25
29.	If we multiply a number by 9, subtract 8 and divide by 7 we get 13. What is the number?.	8	9	10	11	13
30.	What is the cube root of 27?	1	2	3	4	9

31-33. A box contains 25 black marbles, 35 white marbles and 40 red marbles. If one marble is taken at random from the box, what is the probability that it will be:

		(a)	(b)	(c)	(d)	(e)
31.	A red marble?	0·35 or 7/20	0·4 or 2/5	0·45 or 9/20	0·5 or 1/2	0·75 or 3/4
32.	A white marble? . . .	0·2 or 1/5	0·25 or 1/4	0·3 or 3/10	0·35 or 7/20	0·4 or 2/5
33.	A marble that is not black?	0·35 or 7/20	0·4 or 2/5	0·65 or 13/20	0·7 or 7/10	0·75 or 3/4

		(a)	(b)	(c)	(d)	(e)
34.	What is $\frac{17}{1000}$ expressed as a decimal fraction? . .	10·7	1·7	0·7	0·17	0·017
35.	$(7 \times 1000) + (8 \times 100) + (9 \times 10) = 100 \times ?$ . .	78·9	79·8	789	798	71890
36.	10% of the seats in a cinema with 1700 seats were empty. How many people were in the cinema?	170	1170	1530	1570	1630

37.
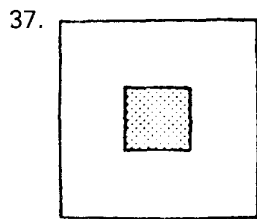
Look at this figure, then work out what fraction of the whole area is shaded. . . . .

	(a)	(b)	(c)	(d)	(e)
	$\frac{1}{10}$	$\frac{1}{9}$	$\frac{1}{5}$	$\frac{1}{4}$	$\frac{1}{3}$

38.
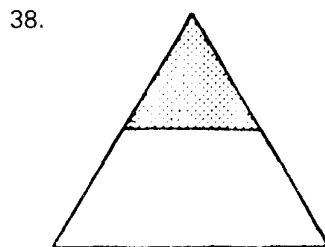
Now look at this figure and work out what fraction of the whole area is shaded. . . . . . .

(a)	(b)	(c)	(d)	(e)
$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{7}$	$\frac{1}{8}$

		(a)	(b)	(c)	(d)	(e)
39.	**99, 97, 93, 85, 69,** ? What number comes next?	53	43	39	37	35
40.	If we cube a number greater than 0 and divide the result by 16 we get the number we started with. What is the number?	0	1	2	3	4

41-42. Two children had to be chosen from a group of three boys and four girls:

		(a)	(b)	(c)	(d)	(e)
	B₁ B₂ B₃ G₁ G₂ G₃ G₄	(a)	(b)	(c)	(d)	(e)
41.	In how many ways could two girls be chosen? .	3	4	6	9	10
42.	In how many ways could a boy and a girl be chosen?	3	4	6	9	12
43.	27% of 30 − 17% of 30 = ...	2	3	13	17	25
44.	The sum of two numbers is 89 and the difference between them is 23. What is the **smaller** number?	(a) 16	(b) 29	(c) 31	(d) 33	(e) 57

45-47.

The total in the circle is 16
The total in the triangle is 16
The total in the square is 14
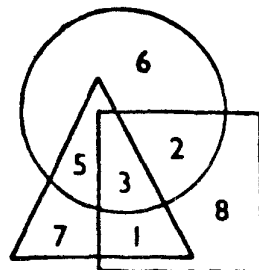
		(a)	(b)	(c)	(d)	(e)
45.	Which number must be increased if all the totals are to be equal?	1	2	3	7	8
46.	Which number must be changed if the totals in the circle, in the triangle and in the square are all to be changed equally?.	3	5	6	7	8
47.	Which number must be increased if the totals in the circle and the triangle are to be 6 greater than the total in the square?.	2	5	6	7	8

48-50. A group of 100 people were asked to estimate the length of a line.
It's true length was 5 cms.
The histogram shows the distribution of their estimates. . . . .

Estimated length of line in cms

	(a)	(b)	(c)	(d)	(e)
48. How many people guessed correctly?	5	15	20	25	30
49. What percentage of the group were only one centimetre out in their estimate?.	35	40	45	50	55
50. What is the mean of the group's estimates? . .	4·4	4·5	4·7	4·9	5
51. 0, 3, 10, 21, 36, ? What number comes next? . .	41	45	55	57	61
52. 14·9 × 5·4 ÷ 2·7 = ...	29·8	31·2	32·8	39·8	46·8

53. Divide 200 in the ratio 7:3 . . . . 110:90 120:80 130:70 140:60 150:50

54. Divide 600 in the ratio 3:2:1

(a)	(b)	(c)	(d)	(e)
200:150:100	200:200:100	200:200:200	300:100:100	300:200:100

55. Divide 300 in the ratio 3:4:2:6

(a)	(b)	(c)	(d)	(e)
50:80:40:120	60:80:40:120	60:70:50:120	60:80:50:110	70:80:30:120

	(a)	(b)	(c)	(d)	(e)
56. How many rectangular blocks 30 cms long, 20 cms wide and 6 cms deep will fit into a rectangular box 90 cms long, 80 cms wide and 72 cms deep?	144	154	156	164	166
57. The line round a football pitch is 350 metres long. The pitch is 35 metres longer than it is broad. How many metres long is it?	85	95	105	125	135
58. What is the radius in cms of a circle of 62·8 cms circumference? (Take π = 3·14)	5	6	8	9	10
59. What is the area in cm² of a circle of 3 cms radius? (Take π = 3·14)	31·46	31·4	30·28	30·26	28·26
60. If a two figure number is written backwards it makes a number that is 72 smaller. What is the number?.	68	75	86	91	97

END OF TEST 3

Appendix 2 Syntax used to derive scores from the BCS70 16-year arithmetic dataset

* Read in original datafile as found on Unix archive (6,351 cases).

get file='j:\mathsc.sav'.

* Display frequencies of raw item scores.

fre car1 to car60.

* Delete cases where all 60 item scores are blank (i.e. test not attempted at all).

```
select if (not(car1='   ' and car2='   ' and car3='   ' and car4='   ' and car5='   '
and car6='   ' and car7.1=' ' and car7.2=' ' and car8='   ' and car9='   ' and car10='   '
and car11='   ' and car12='   ' and car13='   ' and car14='   ' and car15='   '
and car16='   ' and car17='   ' and car18='   ' and car19='   ' and car20='   '
and car21='   ' and car22='   ' and car23.1=' ' and car23.2=' ' and car24='   '
and car25='   ' and car26='   ' and car27='   ' and car28='   ' and car29='   '
and car30='   ' and car31='   ' and car32='   ' and car33='   ' and car34='   '
and car35='   ' and car36='   ' and car37='   ' and car38='   ' and car39.1=' '
and car39.2=' ' and car40='   ' and car41='   ' and car42='   ' and car43='   '
and car44='   ' and car45='   ' and car46='   ' and car47='   ' and car48='   '
and car49='   ' and car50='   ' and car51='   ' and car52='   ' and car53='   '
and car54='   ' and car55.1=' ' and car55.2=' ' and car56='   ' and car57='   '
and car58='   ' and car59='   ' and car60='   ')).
```

execute.

* Derive arithmetic 'total score' variables from the item scores.
* The raw arithmetic variables car1 to car60 are coded with the alphabetic digit (A-E) of the
* alternative that was selected as correct, which varies with each question.
* Need to recode the correct alternative in each case to '1' and incorrect to '0', then total up
* the correct and incorrect responses ('not answered'=-1).

* Create vector MATHS, with 60 slots for the arithmetic items.

vector #maths (60).

* Initialise all 60 slots to zero (i.e. incorrect), with intention of replacing by '1' if found to be
* correct, or' -1' if no answer (i.e. 5-char string is blank).

```
loop #i=1 to 60.
compute #maths(#i)=0.
end loop.
```

* Proceed through all 60 test items, re-coding slot to '1' if found to be correct, or' -1' if no
* answer (i.e. 5-char string is blank)

```
If (car1='   D ')#maths(1)=1.
If (car1='     ')#maths(1)=-1.

If (car2=' B   ')#maths(2)=1.
If (car2='     ')#maths(2)=-1.

If (car3='A    ')#maths(3)=1.
If (car3='     ')#maths(3)=-1.

If (car4='    E')#maths(4)=1.
If (car4='     ')#maths(4)=-1.
```

```
If (car5=' C ')#maths(5)=1.
If (car5='    ')#maths(5)=-1.

If (car6=' C ')#maths(6)=1.
If (car6='    ')#maths(6)=-1.

If (car7.1='A ' and car7.2='  ')#maths(7)=1.
If (car7.1=' ' and car7.2='  ')#maths(7)=-1.

If (car8='  D ')#maths(8)=1.
If (car8='    ')#maths(8)=-1.

If (car9=' B  ')#maths(9)=1.
If (car9='    ')#maths(9)=-1.

If (car10='   E')#maths(10)=1.
If (car10='    ')#maths(10)=-1.

If (car11='  D ')#maths(11)=1.
If (car11='    ')#maths(11)=-1.

If (car12='A   ')#maths(12)=1.
If (car12='    ')#maths(12)=-1.

If (car13=' B  ')#maths(13)=1.
If (car13='    ')#maths(13)=-1.

If (car14='   E')#maths(14)=1.
If (car14='    ')#maths(14)=-1.

If (car15=' B  ')#maths(15)=1.
If (car15='    ')#maths(15)=-1.

If (car16=' C  ')#maths(16)=1.
If (car16='    ')#maths(16)=-1.

If (car17=' C  ')#maths(17)=1.
If (car17='    ')#maths(17)=-1.

If (car18='A   ')#maths(18)=1.
If (car18='    ')#maths(18)=-1.

If (car19='   E')#maths(19)=1.
If (car19='    ')#maths(19)=-1.

If (car20='B   ')#maths(20)=1.
If (car20='    ')#maths(20)=-1.

If (car21=' C  ')#maths(21)=1.
If (car21='    ')#maths(21)=-1.

If (car22='  D ')#maths(22)=1.
If (car22='    ')#maths(22)=-1.

If (car23.2=' D ' and car23.1='  ')#maths(23)=1.
If (car23.2='  ' and car23.1='  ')#maths(23)=-1.

If (car24='A   ')#maths(24)=1.
If (car24='    ')#maths(24)=-1.
```

```
If (car25='   E')#maths(25)=1.
If (car25='    ')#maths(25)=-1.

If (car26=' B  ')#maths(26)=1.
If (car26='    ')#maths(26)=-1.

If (car27='A   ')#maths(27)=1.
If (car27='    ')#maths(27)=-1.

If (car28='  C ')#maths(28)=1.
If (car28='    ')#maths(28)=-1.

If (car29='  D ')#maths(29)=1.
If (car29='    ')#maths(29)=-1.

If (car30='  C ')#maths(30)=1.
If (car30='    ')#maths(30)=-1.

If (car31=' B  ')#maths(31)=1.
If (car31='    ')#maths(31)=-1.

If (car32='   D ')#maths(32)=1.
If (car32='    ')#maths(32)=-1.

If (car33='   E')#maths(33)=1.
If (car33='    ')#maths(33)=-1.

If (car34='   E')#maths(34)=1.
If (car34='    ')#maths(34)=-1.

If (car35='A   ')#maths(35)=1.
If (car35='    ')#maths(35)=-1.

If (car36='  C ')#maths(36)=1.
If (car36='    ')#maths(36)=-1.

If (car37=' B  ')#maths(37)=1.
If (car37='    ')#maths(37)=-1.

If (car38='A   ')#maths(38)=1.
If (car38='    ')#maths(38)=-1.

If (car39.2=' D ' and car39.1='  ')#maths(39)=1.
If (car39.2='   ' and car39.1='  ')#maths(39)=-1.

If (car40='   E')#maths(40)=1.
If (car40='    ')#maths(40)=-1.

If (car41='  C ')#maths(41)=1.
If (car41='    ')#maths(41)=-1.

If (car42='   E')#maths(42)=1.
If (car42='    ')#maths(42)=-1.

If (car43=' B  ')#maths(43)=1.
If (car43='    ')#maths(43)=-1.

If (car44='  D ')#maths(44)=1.
If (car44='    ')#maths(44)=-1.
```

```
If (car45='    E')#maths(45)=1.
If (car45='     ')#maths(45)=-1.

If (car46='A    ')#maths(46)=1.
If (car46='     ')#maths(46)=-1.

If (car47=' B   ')#maths(47)=1.
If (car47='     ')#maths(47)=-1.

If (car48='   D ')#maths(48)=1.
If (car48='     ')#maths(48)=-1.

If (car49='  C  ')#maths(49)=1.
If (car49='     ')#maths(49)=-1.

If (car50='A    ')#maths(50)=1.
If (car50='     ')#maths(50)=-1.

If (car51='  C  ')#maths(51)=1.
If (car51='     ')#maths(51)=-1.

If (car52='A    ')#maths(52)=1.
If (car52='     ')#maths(52)=-1.

If (car53='   D ')#maths(53)=1.
If (car53='     ')#maths(53)=-1.

If (car54='    E')#maths(54)=1.
If (car54='     ')#maths(54)=-1.

If (car55.1=' B' and car55.2='   ')#maths(55)=1.
If (car55.1=' ' and car55.2='   ')#maths(55)=-1.

If (car56='A    ')#maths(56)=1.
If (car56='     ')#maths(56)=-1.

If (car57='  C  ')#maths(57)=1.
If (car57='     ')#maths(57)=-1.

If (car58='    E')#maths(58)=1.
If (car58='     ')#maths(58)=-1.

If (car59='    E')#maths(59)=1.
If (car59='     ')#maths(59)=-1.

If (car60='   D ')#maths(60)=1.
If (car60='     ')#maths(60)=-1.

* Initialise and label two variables which will keep track of the number of correct and incorrect
* answers.

compute mathscore=0.
compute mathincorrect=0.
format mathscore mathincorrect (f3.0).
missing values mathscore mathincorrect(-1).

variable lables mathscore 'BCS70 16-year Arithmetic scores (out of 60)'.
value labels mathscore
-1 'Test not attempted'.
```

variable lables mathincorrect 'BCS70 16-year Arithmetic - no. incorrect scores'.
value labels mathincorrect
-1 'Test not attempted'.

* Loop through all 60 test items, increasing total each time a correct answer is encountered,
* whilst correspondingly logging incorrect responses.

loop #i=1 to 60.
if (#maths(#i)=1)mathscore=mathscore+1.
if (#maths(#i)=0)mathincorrect=mathincorrect+1.
end loop.

* If no correct nor incorrect replies, assume test was not attempted and set total to missing.

if (mathscore=0 and mathincorrect=0)mathscore=-1.

* Derive total number of questions attempted (i.e. correct plus incorrect).

compute mathanswered=-1.
if (mathscore + mathincorrect ge 1)mathanswered=mathscore+mathincorrect.
format mathanswered (f3.0).
missing values mathanswered (-1).

variable lables mathanswered 'BCS70 16-year Arithmetic test - no. of questions attempted'.
value labels mathanswered
-1 'Test not attempted'.

* Express total correct answers as a percentage of all questions in test (i.e. 60).

compute mathspc=mathscore*100/60.

variable labels mathspc 'BCS70 16-year Arithmetic percentages'.

* Display frequency counts of total score, percentage score, and total questions attempted.

fre mathscore mathanswered mathspc.

* Sort the file by variable KEY, so as to be able to merge with other longitudinal data.

sort cases by key (A) .

save outfile='j:\Cleaned BCS70 16-year arithmetic dataset (3,677 cases).sav'/keep=key
yno_tc car1 to mathspc.