# CONSTRUCTING RASCH SCALES

Table of Contents

[CONSTRUCTING RASCH SCALES 1](#_Toc430702665)

[PRE-REQUISITES 2](#_Toc430702666)

[SOFTWARE REQUIRED 2](#_Toc430702667)

[DATA PREPARATION 2](#_Toc430702668)

[Figure 1: Format of the data for scale construction 3](#_Toc430702669)

[Figure 2: Saving the file in csv format 3](#_Toc430702670)

[Figure 3: Snapshot of key.csv 4](#_Toc430702671)

[DATA CLEANING 4](#_Toc430702672)

[Figure 4: Checking for valid responses 5](#_Toc430702673)

[UNDERSTANDING THE CODEFRAME 5](#_Toc430702674)

[Figure 5: Snapshot of a code frame 6](#_Toc430702675)

[CONSTRUCTION OF SCALES 6](#_Toc430702676)

[Collating the files 6](#_Toc430702677)

[Preparing the anchor file 7](#_Toc430702678)

[Running the script 9](#_Toc430702679)

[Running Quest 11](#_Toc430702680)

[Figure 6: The Quest window 11](#_Toc430702681)

## PRE-REQUISITES

### SOFTWARE REQUIRED

The following software is required for the development of scales.

1) R/ R Studio\* or SPSS

2) Quest (ACER)

3) Microsoft Excel (2010 and above)

\* R and R-Studio scripts are optional as they help expedite the task of converting the raw data (files with responses) to suitable formats and automated build of script files for the Quest software. The format conversion of raw data for Quest compatible software can be done using SPSS also (Manual prepared by Kumaresh and Manjunath).

##### INSTALLATION

**For R installation**

https://cran.r-project.org/doc/manuals/r-release/R-admin.html#Installing-R-under-Windows

**R-Studio installation**

http://www.dummies.com/how-to/content/how-to-install-and-configure-rstudio.html

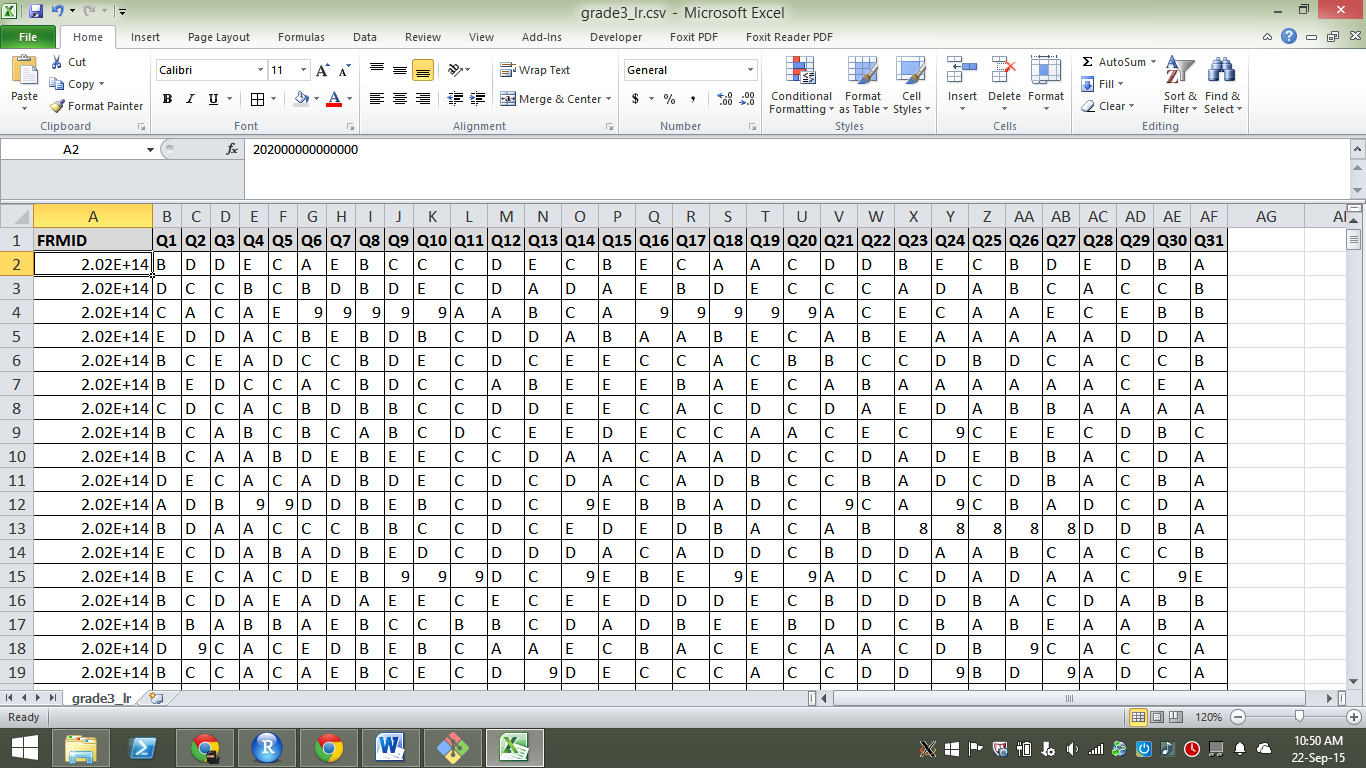
## DATA PREPARATION

The data for construction of scales should be prepared in the prescribed format.

1) Copy all the raw responses (either through OMR reading or manual tabulation) into a single sheet by grade and subject. Usually the scales are constructed for each subject by linking the grades from the lowest to the highest.

2) The spreadsheet should only have the formID(or any other identifier) and the responses to all questions. The responses on one student should be in one row.

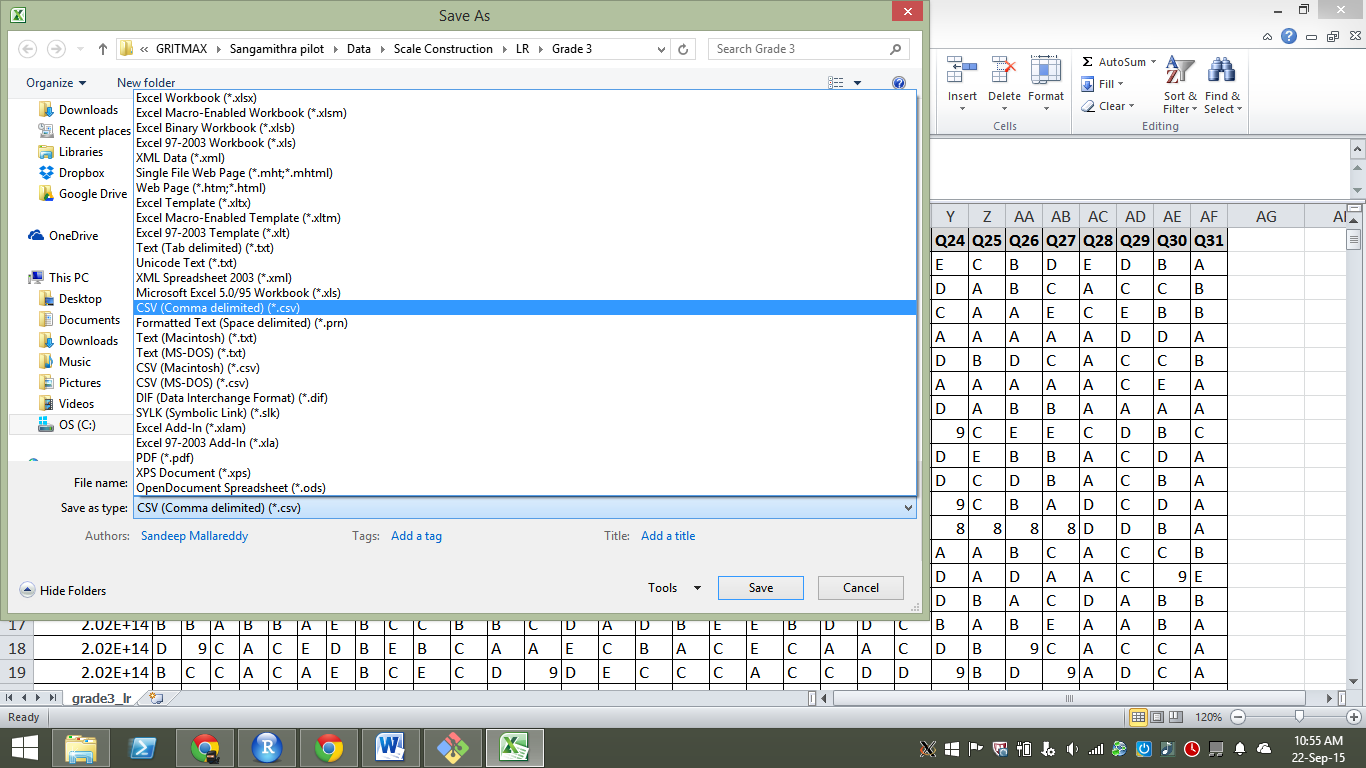
### Figure 1: Format of the data for scale construction



3) Save the file in the .**csv (comma separated values)**.

The file name should be ideally “grade {number}\_{subject}.csv”. *e.g.* grade3\_maths.csv

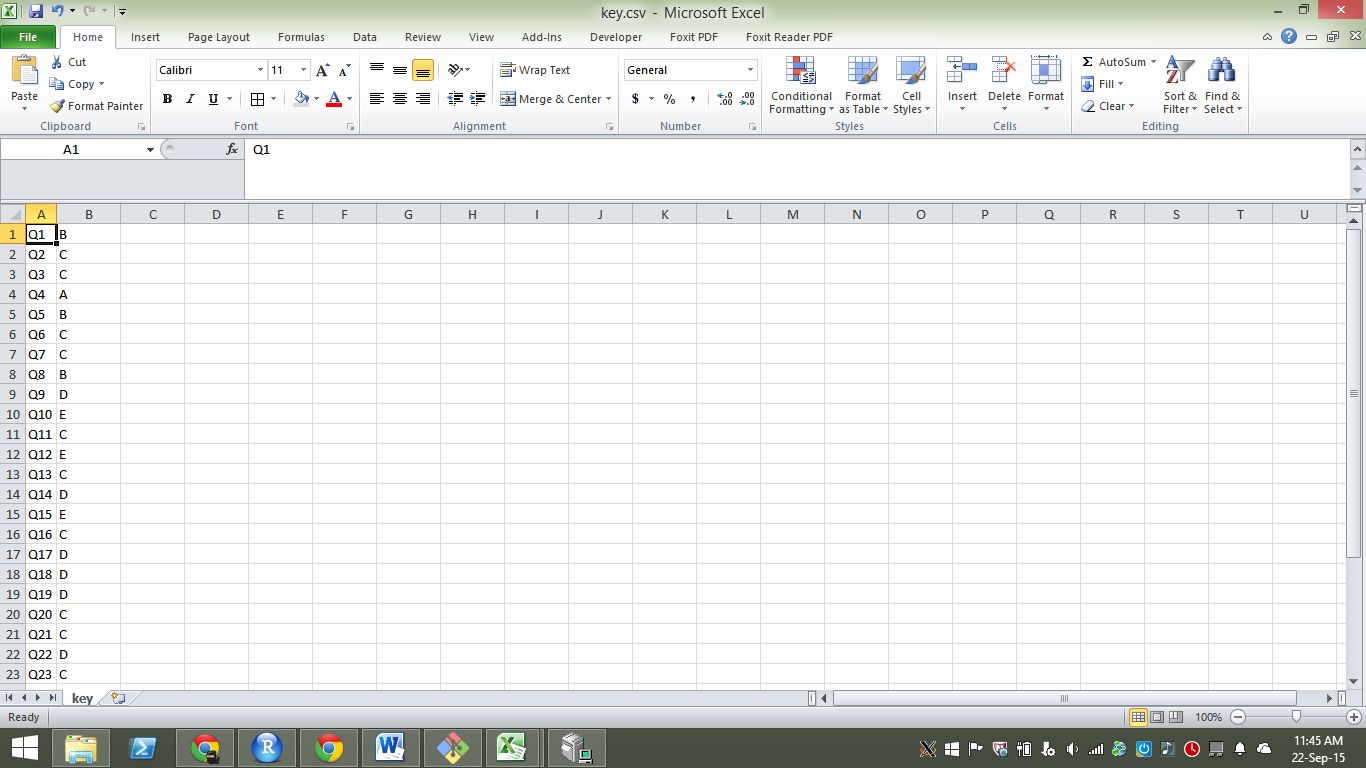
### Figure 2: Saving the file in csv format



**Note: The .csv format doesn’t allow more than one sheet to be stored.**

4) Create a **key.csv** which contains the questions and the correct solution key. There can be only one solution key and it is case sensitive. Use upper case alphabets for the solution key.

### Figure 3: Snapshot of key.csv



Note: The question names should be exactly the same as the column names in the data sheet containing the responses. The names are case sensitive, hence the best way to avoid any errors is to copy the row names and paste them as values and transpose.

## DATA CLEANING

The next step entails cleaning the data. The data cleaning step entails the following

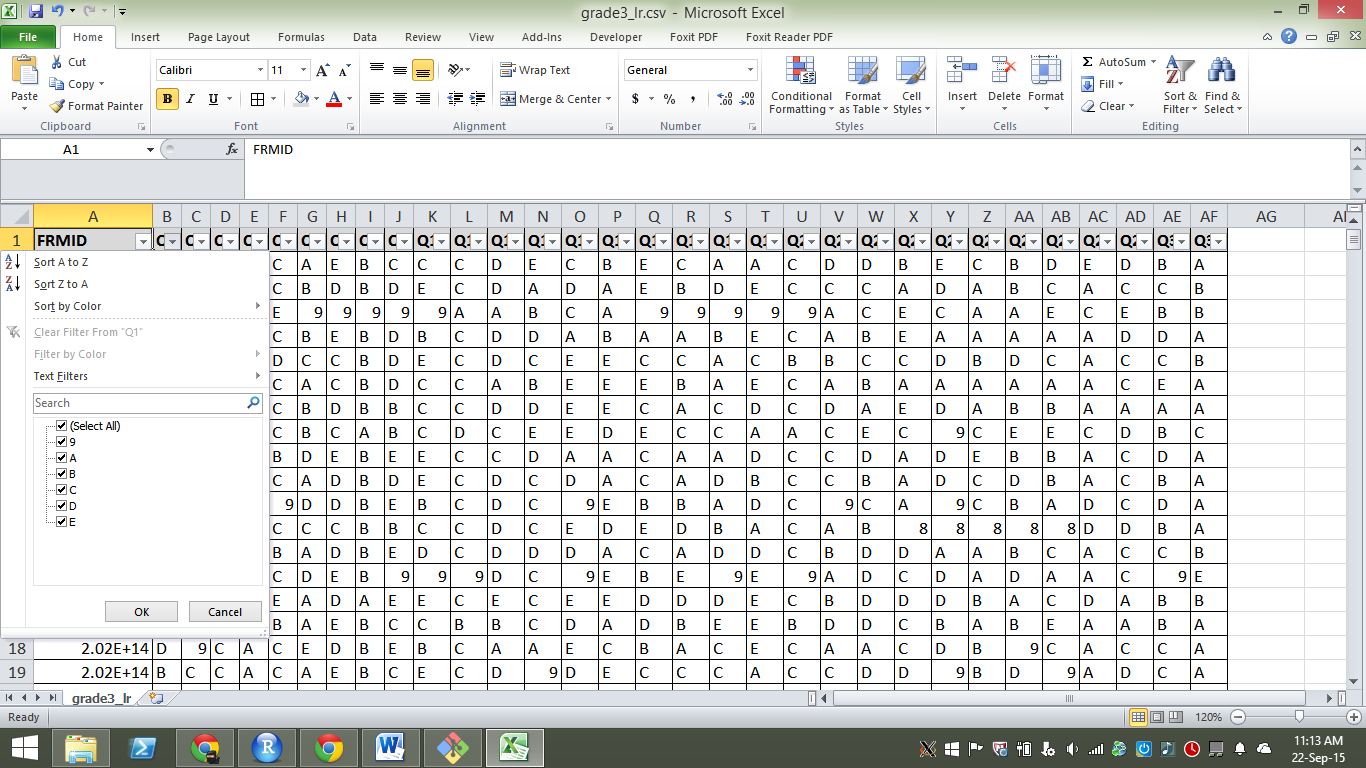
1) Convert all blank responses to number ‘9’

2) Replace multiple responses for a question (AB,ABC,ABCD, ABCDE etc) to’8’

3) For subjects which have only 4 options, if responses contain ‘E’, convert them to 8

4) Check all responses, question by question to ensure that only A,B,C,D,E,8,9 are the only responses in each column

### Figure 4: Checking for valid responses



## UNDERSTANDING THE CODEFRAME

The code frame is document listing the items used for testing. The code frame is a master list of the items used along with the Grade, Subject, Descriptor, Unique ID, question number in a particular question paper. This information helps us in identifying the links between various question papers and is very crucial for constructing RASCH scales spanning multiple grades.

**Figure 5** shows a part of the code frame. Question with S.no 2, is meant for Grade 4, 5 and belongs to Numbers/Algebra strand and the item tests the skill of ‘Adding a 5 digit number with a 4 digit number’.

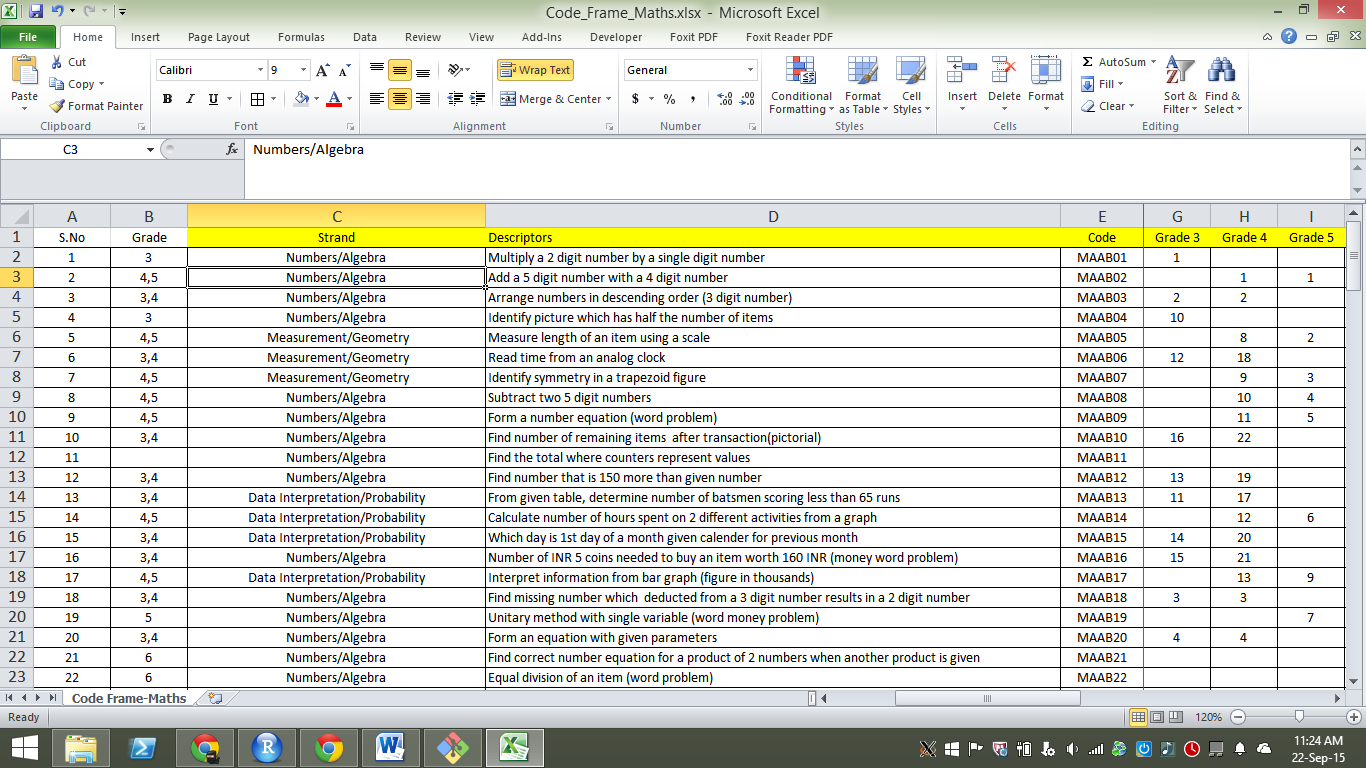
This item also serves as a link item between Grade 4 & 5 question papers. This can be confirmed by the fact that this item record appears under Grade 4 and Grade 5 columns (extreme right). The number in each of the columns shows the question sequence number in the respective question papers. This question is Q.no 1 in both Grade 4 and Grade 5.

Similarly, S.no 6 serves as a link between 3,4. The item is Q.no12 in Grade 3 and Q.no18 in Grade 4.

Note: As a best practice, it is good to list down the link items and their sequence number before linking across grades in the following format

|  |  |
| --- | --- |
| Grade 3 | Grade 4 |
| 2 | 2 |
| 12 | 18 |
| 16 | 22 |

### Figure 5: Snapshot of a code frame



## CONSTRUCTION OF SCALES

### Collating the files

This step involves getting all the files required for the scales in one folder. Since linking is done by linking one grade data to a base scale, each grade will have a folder. The recommended structure for the folder is as following

grade4\_maths.csv

key.csv

Q.exe

scaleconstructionscript.R

anchorgrade3\_maths.doc

GRADE 3

GRADE 4

GRADE 5

MATHS

1) Copy the files (scaleconstructionscript.R & Q.exe) from Rasch\_Scale\_Construction / LinkScaleFiles to Grade folder

2) Copy data file and key to the Grade folder. Also copy the anchor file from the previous grade folder. E.g. If you are constructing scales for Maths Grade 4, copy anchorgrade3\_maths.doc

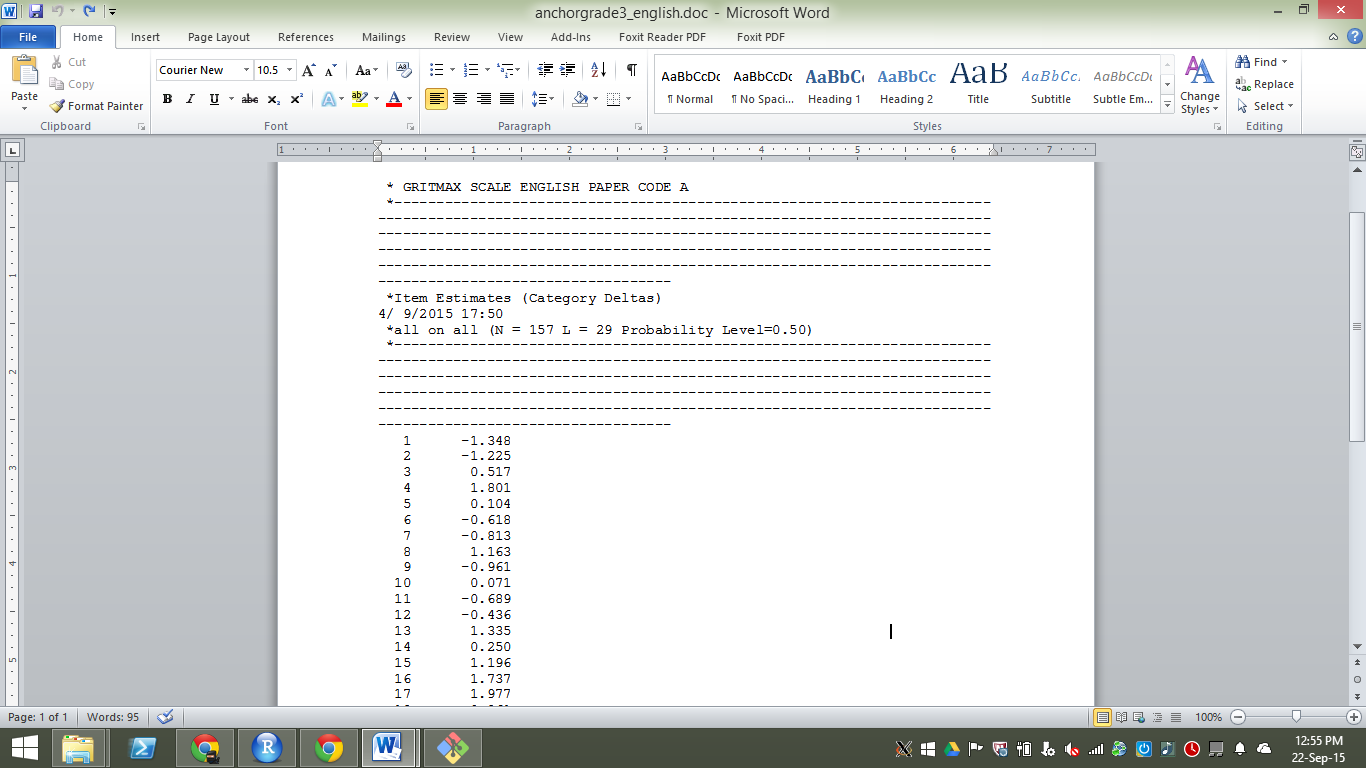
Note: For the lowest grade there will not be an anchor file, this is the base scale

### Preparing the anchor file

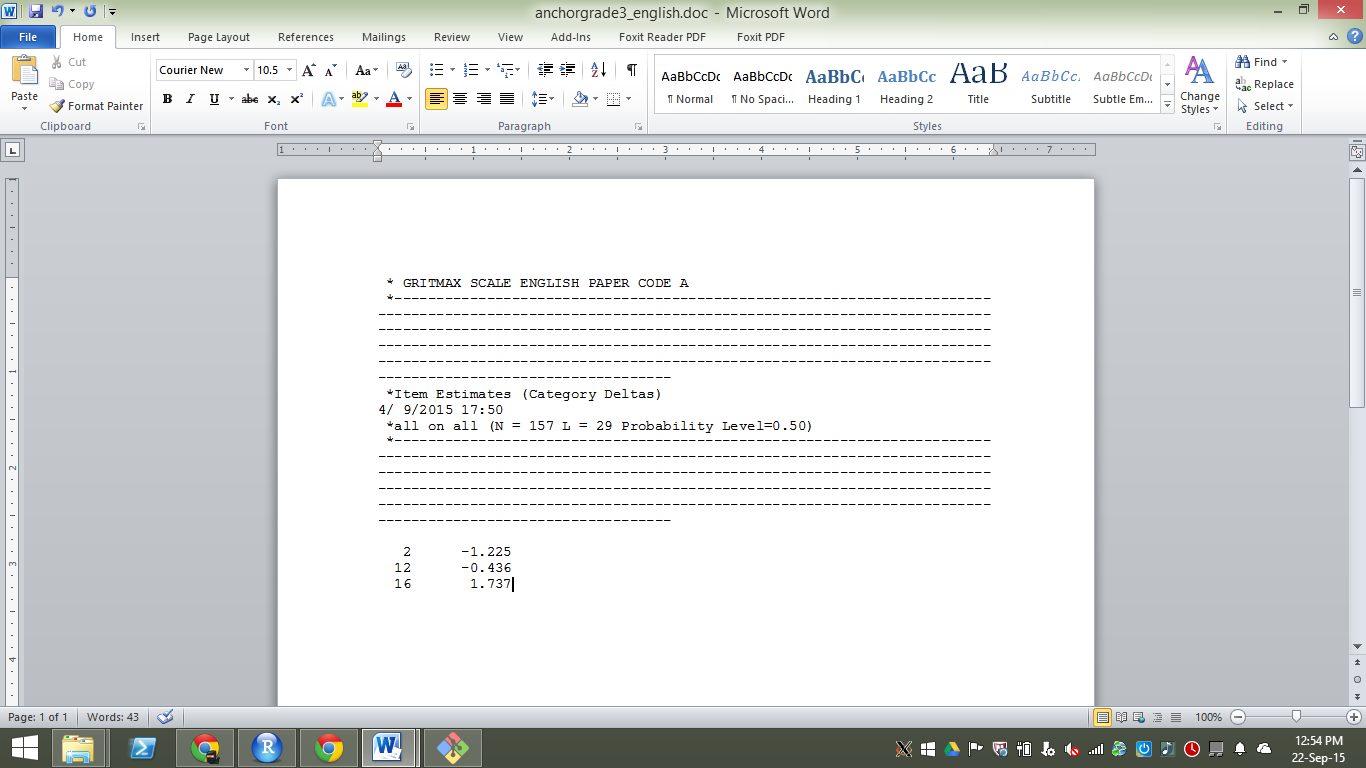
Example: Let’s use the list from previous example to do this. Let us consider that there are 3 items in Grade 3 & 4 which are linked. The following is the list of question numbers in respective question papers.

|  |  |
| --- | --- |
| Grade 3 | Grade 4 |
| 2 | 2 |
| 12 | 18 |
| 16 | 22 |

3) Open the anchor file



4) Remove items which are not links. Here since this is Grade 3 anchor file, remove all items except for **2,12,16**

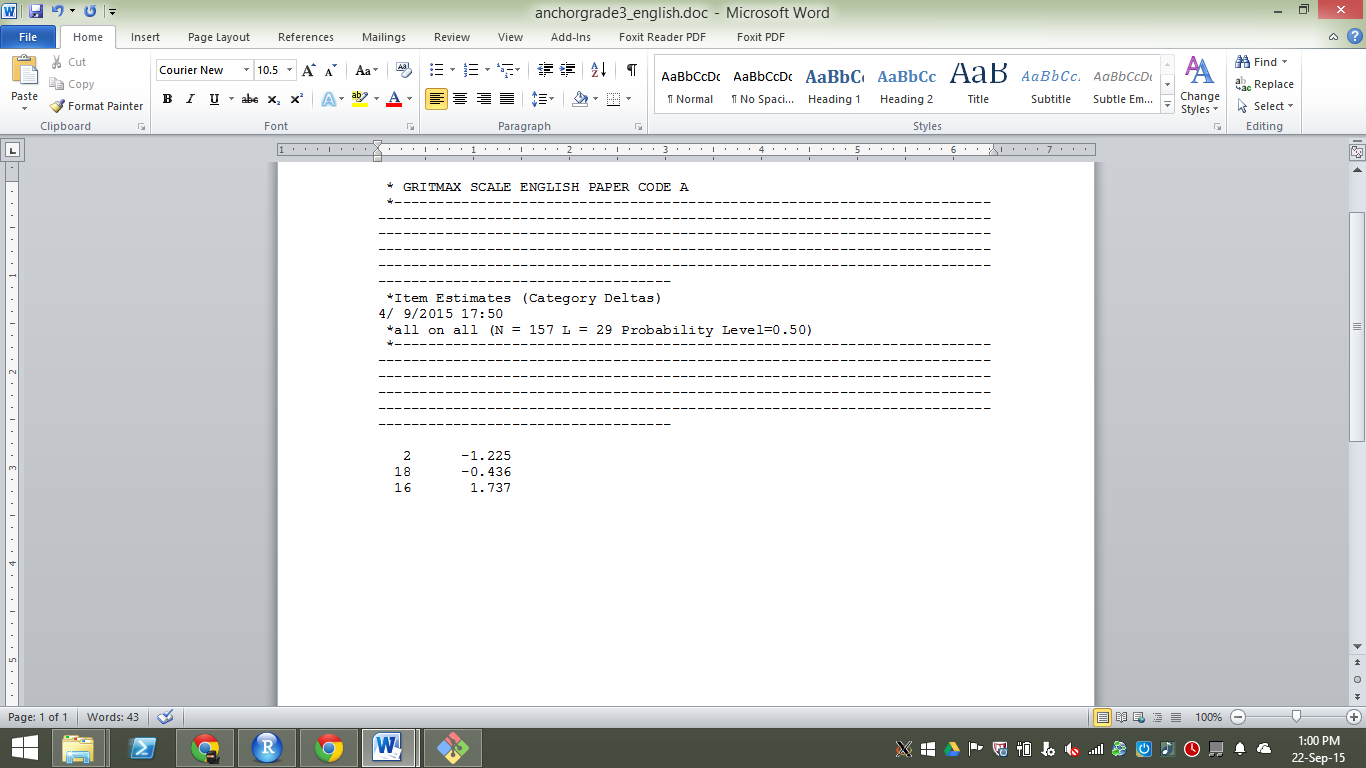


5) Rename the items to the question numbers in Grade 4.

2 -> 2

12 -> 18

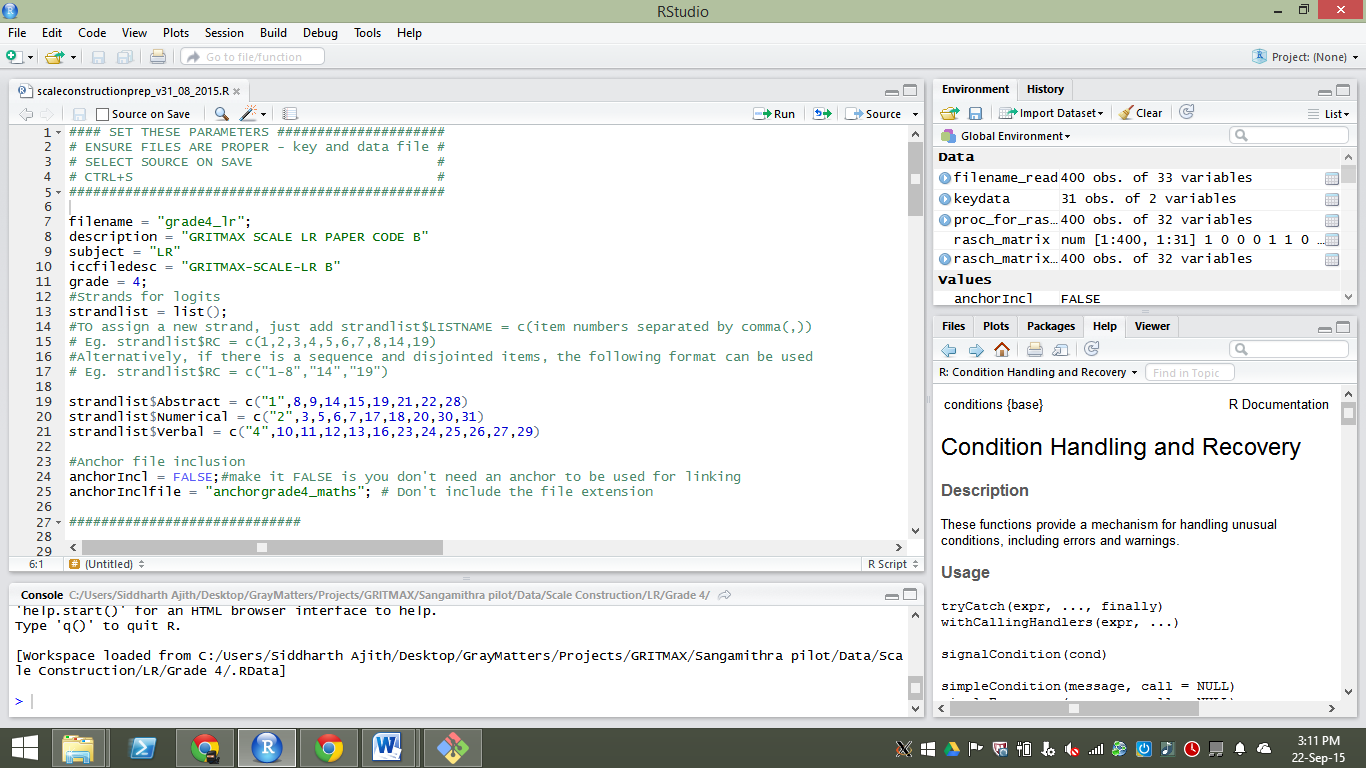
16 -> 22



6) Save the file and exit. Do not change the name of the file.

### Running the script

7) Double click and open scaleconstructionscript.R



8) Fill in the following parameters

|  |  |
| --- | --- |
| Variable | Description |
| filename | The filename of the data file(E.g. grade4\_maths). Ensure the file name is exactly the same. Don’t include the extension(“.csv”) |
| description | The name you would want to see in the Quest file outputs generated |
| subject | LR,MATHS,ENGLISH,SCIENCE |
| iccfiledesc | The name of the file generally the short form of description |
| grade | The grade for which the scale is being generated |

9) Fill in the strand wise details. Here the strand names are flexible. In the script file, you can define the strand wise names and the questions which define the strands.

strandlist = list(); # This line should be ignored

To define a strand name, in a new line below strandlist =list(); with

strandlist${STRANDNAME} = c(question numbers);

Example:  
The following lines define 3 strands- Abstract, Numerical and Verbal,

strandlist$Abstract = c("1",8,9,14,15,19,21,22,28)

strandlist$Numerical = c("2",3,5,6,7,17,18,20,30,31)

strandlist$Verbal = c("4",10,11,12,13,16,23,24,25,26,27,29)

Note: The first number in each of the strand definition should be in double quotes. The rest of the following question numbers need not be in double quotes.

Note: The question numbers defined here are in sequential order as defined by columns in data file.

Example: If suppose, there are 5 questions , Q1,Q2,Q3,Q4,Q5 and all of them are part of a hypothetical strand “TestStrand”. If Q4 is deleted from analysis. The data file would consist of the Q1, Q2, Q3, Q5 in this order. Thus “TestStrand” is defined in the following way:-

strandlist$TestStrand = c(“1”,2,3,4)

9) Enable or disable anchoring.

If you want to link the scales from the previous test, then enable anchors by the following command

anchorIncl = TRUE;

The above case is always applicable when constructing multi grade scales. However, for the base scale(the scale of the lowest grade), there wouldn’t be any anchors to refer to and hence the anchors should be disabled.

anchorIncl = FALSE;

10) Set the anchor file

If you enable anchoring by setting “anchorIncl = TRUE” , then set

anchorInclfile = “name of the anchor file”;

Example:

anchorInclfile = “anchorgrade3\_maths”;

11) Select “**Source on Save**” and then press **Ctrl key + S.** The script takes some time to run. In case of any errors, the script would stop and show the errors in the console below. If successful, the script should create the following files

somename\_ICC.csv – The file for ICC curves. This is not important

Q.txt – The Quest config file

key.txt – The key file for Quest

somename.dat – The data file for Quest

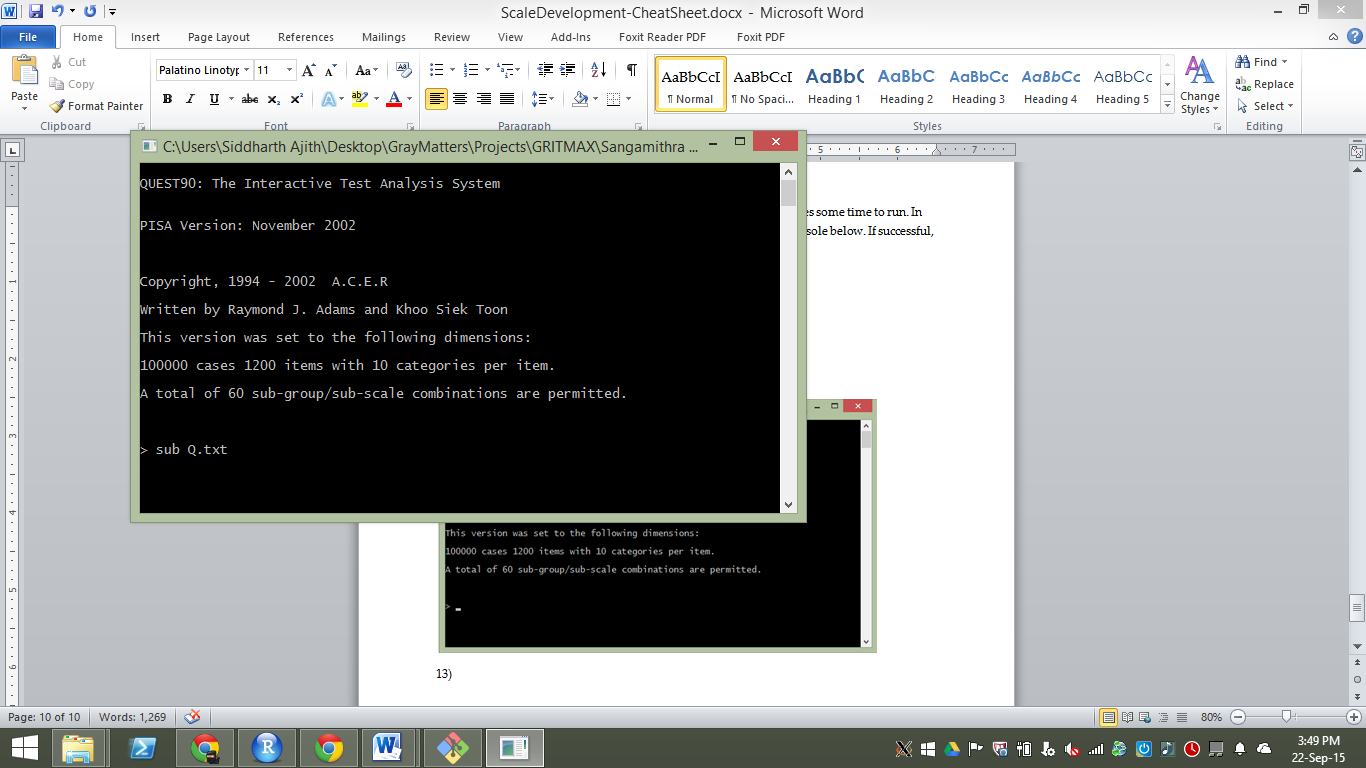
### Running Quest

12) Double click and open Q.exe

13) Type the following command at the prompt

*sub Q.txt*

### Figure 6: The Quest window



14) The Quest program would run and generate the following files

|  |  |
| --- | --- |
| Files | Description |
| Itn{filename}.doc | The item characterstics file |
| Itnshort{filename}.doc | The shorter item characterstics file |
| show{filename}.doc | The file containing the model file, Item person maps |
| Logit{filename}.doc | The file containing the test score to scale score conversions |

Depending on the number of stands defined, each strand gets the following files generated

|  |  |
| --- | --- |
| Files | Description |
| {strandname}Logit.doc | The file contains the test score to scale score conversions for each strands |

15) The linked scale for a grade is complete. Repeat steps for each grade.

**NOTE:** The TS to SS conversion are not automatically generated by the QUEST software. The scale score for 0 and max score can be generated in the following way

If T1 to TN-1 scores are the scale scores for 1 to (N-1) total score then

min score T0 is T1+(T1-T2)+((T1-T2)-(T2-T3))

max score TN is TN-1+(TN-1-TN-2)+(( TN-1-TN-2)-(TN-2-TN-3))

Example:

|  |  |
| --- | --- |
| Test score | SS |
| 6 (Max score) | (65+(65-45)+((65-45)-(45-33)) |
| 5 | 65 |
| 4 | 45 |
| 3 | 33 |
| 2 | 22 |
| 1 | 10 |
| 0 (Min score) | =(10+(10-22)+((10-22)-(22-33)) |