

Processing Standard University Course Evaluations on OITUNIX October, 2003

These instructions apply ONLY to course evaluations that use either:

- "University of Massachusetts Course and Teaching Evaluation" blue bubble sheets with 12 pre-printed questions with responses coded 1=Hopelessly Inadequate to 7=Unusually Effective
- "University Evaluation General Answer Sheet" green bubble sheets with 42 questions, responses 1-7.

We will henceforth refer to these as "blue" and "green" evaluation forms. Do not confuse the green course evaluation forms with the 200 item "General Purpose Answer Sheet", with responses 1-5, which also happens to be green, but are not normally used for course evaluations, and will not work with the programs described here.

SRTI (Student Response to Instruction) course evaluations are processed by the Office of Academic Planning and Assessment.

If you are using any other forms to collect course evaluation information, the programs provided here will not work "off the shelf", though someone thoroughly familiar with SPSS syntax (not just menus) may be able to customize them for you.

These programs use SPSS on OITUNIX. They will not run on SPSS Windows "as is", but could be converted to run under SPSS Production Facility on Windows. Converting to SPSS (Interactive) on Windows would be somewhat more painful, as many of the commands do not have a period at the end, as is required by SPSS Windows. The Production Facility does not require periods at the end of each command.

Preparing evaluations for scanning

The evaluation forms for each class must be preceded by a "Header" sheet. Regardless of whether you use blue or green forms, the class "Header" must be a green form, filled out with the instructor's name, course number, section, semester, year, department code, AND it must have a 999 in the "batch number" area. This 999 code identifies the sheet as a header, and is critical to correct processing. Students using the green evaluation form should not put any marks in the "batch number".

Put the evaluations for all the courses (each with its header sheet) together in the order that you want the report to print the courses, and take them, with a blank diskette, to the Data Processing center at Whitmore for scanning. They will put a file with the results of the scanning on the diskette.

What you need to have

- An OITUNIX username and password. Most people already have this, but may not be aware of it. If you get your email through OIT, or use the UDRIVE, you have an OITUNIX username. Unless you've changed it, the OITUNIX password should be the same as your email password was when you first got your email set up. If you are unable to log on, go to OIT Help Desk. They can check whether you have a username, and change the password if necessary.

- An Internet or telephone connection to OITUNIX
- SecureCRT, or similar software to log on to OITUNIX (available from OIT)
- Secure FTP software to transfer files between your local machine and OITUNIX (available from OIT)
- Diskette from Whitmore with scanned evaluations.

What you need to know

- How to log on to OITUNIX with SecureCRT
- How to transfer files from your PC to OITUNIX
- Some basic unix commands and concepts, at a minimum
 - ✓ Filenaming conventions
 - ✓ `ls` list names of files
 - ✓ `more` display file on screen
 - ✓ `cp` copy file
 - ✓ `rm` delete files
 - ✓ `chmod` change file permissions
 - ✓ text editing with `emacs`, `pico` or some other editor. You could use WordPad or similar text editor on your PC, but remember to copy your edited file to OITUNIX each time to edit it.

If you don't know these commands, contact the OIT Help Desk. They should be able to get you started. You need a reasonable level of comfort with UNIX before embarking on this project.

Getting Started

The course evaluation reports are created using SPSS. Go to <http://www-unix.oit.umass.edu/~statdata/software>, click on "Processing Course Evaluations" and download the zipped program files. Move the zip file to your OITUNIX account using binary file transfer. Log on to your OITUNIX account and unzip the file there:

```
unzip eval_progs
```

This should unzip four files:

<code>check.sps</code>	spss program to check data validity
<code>evalblue.sps</code>	spss program for reports based on blue forms
<code>evalgreen.sps</code>	spss program for reports based on green forms
<code>printeval</code>	shell script to print results on OIT laser printers

If you will be working with standard blue course evaluations, you can delete `evalgreen.sps`. Conversely, if you are working with green course evaluations, you can delete `evalblue.sps`.

Customizing Programs for Green forms or for Additional Items on Blue forms

If you are using only the pre-printed questions on the blue evaluation forms, you do not need to customize any programs. Proceed to the section ["Put Your Data File on OITUNIX"](#).

The standard report for blue forms prints results for the 9 items on the top half of the form, and the 12 pre-printed questions on the bottom half, for a total of 21 items. If you

used the additional items (numbered 13 to 32 on the form), you will need to customize the program to print these additional items and label them appropriately.

The standard report for green forms prints 42 questions, labeled "Question 1" to "Question 42". If you want more meaningful labels, you will need to customize the program. If you used substantially fewer questions you can save a lot of paper by also customizing the program to print fewer items.

Assigning Labels to Questions

Open the report program you will be using (evalblue.sps or evalgreen.sps) with a text editor. Scroll down till you find the line:

```
COMMENT set up labels for N variables
```

(N is 41 or 42, depending on whether you are working with the blue or green program).

Below this comment you will find the labels for the questions. For green forms, they begin with:

```
compute ql1="Question 1"  
compute ql2="Question 2"
```

Substitute the labels you want printed instead of "Question 1", "Question 2", etc. Labels may be no more than 35 characters, and may NOT include internal quotes. Be sure to leave the quotes around each label. If you have less than 42 questions, leave the excess questions as is.

For blue forms, scroll down a bit further to:

```
COMMENT For additional items, substitute descriptions here.
```

```
compute ql22="Question 13"  
compute ql23="Question 14"
```

Substitute the labels you want printed for "Question 13", "Question 14", etc., using the same rules as for green forms.

Label Responses (Green only)

For Blue forms, the 1-7 response scale means "Hopelessly Inadequate" to "Unusually Effective". Green forms have no pre-assigned scale, so you need to explain what the 1-7 response codes mean. Scroll down the green report program some more until you find the line:

```
COMMENT Put something here to indicate what the 1-7 coding means.  
print / "Responses are coded 1=... to 7=..."/
```

Substitute appropriate text for the ... to label the scale extremes in the print command.

Set Number of Questions to Print

Blue

For Blue forms, the standard report prints 21 questions, 9 from the top half of the page, and 12 from the bottom. If you used additional questions, you must increase the number of questions to be printed. Look for the lines:

```
COMMENT For additional questions, change all 21s on the next 12 lines  
to the number of questions to be printed.
```

```
e.g., for standard blue form with 3 additional questions,  
change all 21s to 24s. (Max is 41; i.e 20 add'l questions.)
```

```
do repeat n=n1 to n21/  
i=1 to 21/  
t1=ta1 to ta21/ t2=tb1 to tb21/
```

To print 3 additional questions, the beginning of the DO REPEAT should be:

```
do repeat n=n1 to n24/  
    i=1 to 24/  
    t1=ta1 to ta24/ t2=tb1 to tb24/
```

The command continues for 12 lines. Be sure to change them all.

Green

Green forms print 42 questions. To reduce this number and save a lot of paper, look for the lines:

```
COMMENT To print fewer than 42 questions, change all 42s in the next  
    12 lines to the number of questions you wish to print.
```

```
do repeat n=n1 to n42/  
    i=1 to 42/  
    t1=ta1 to ta42/ t2=tb1 to tb42/
```

Change all the 42s on the DO REPEAT command to the number of questions you need printed. The command continues over 12 lines, so be sure to change them all. For example, to print only 10 questions, the beginning of the DO REPEAT will be:

```
do repeat n=n1 to n10/  
    i=1 to 10/  
    t1=ta1 to ta10/ t2=tb1 to tb10/
```

Save Customized Report Program

When you've made all the necessary changes, save your customized program. As long as you do not change the evaluation forms, you will be able to continue to use this customized program without having to repeat the customization.

Put your Data file on OITUNIX

Using ftp, with text transfer mode, copy the scanned data file you got from Whitmore to your OITUNIX account. If you don't like the name Whitmore assigned to your file, rename it before the transfer, so it will have a meaningful name on OITUNIX. Remember that the UNIX system distinguishes upper and lower case in filenames.

Check the Data

Before running the reports, it is important to make sure the each class has a header sheet. If a header is missing, the student evaluations for that class will be mixed in with the evaluations for the previous class.

Run the Check Program

Open the check program file, `check.sps`, with a text editor. Near the beginning of the file you will see the lines:

```
COMMENT Substitute the name of your data file for yourdata.dsa. Make  
    Sure it is spelled correctly, and has the correct upper/lower case.  
FILE TYPE NESTED FILE='yourdata.dsa' RECORD=BATCH 25-27 (A) WILD=WARN
```

Substitute the name of your data file for 'yourdata.dsa'. Be sure the filename is typed correctly, including upper/lower case. Keep the apostrophes around the filename. Save the edited `check.sps` file. To process your data with spss, type:

```
spss -m <check.sps>check.out
```

This command runs spss using the `check.sps` program (which examines your data file), and puts the results in file `check.out`. The end of the response should indicate no errors:

```
End of job: 33 command lines 0 errors 0 warnings 0 CPU seconds
```

You may possibly get this warning:

```
>Warning # 10950
>The working file read by AGGREGATE was not in ascending sequence on
>the break variables. It may be impossible to match the aggregated
> output file with the MATCH FILES command using the keyword BY.
End of job: 33 command lines 0 errors 1 warnings 0 CPU seconds
```

This indicates that the courses were not arranged in increasing sequence by course and section prior to scanning. The report will be correct, but will not be ordered by course/section. If you don't mind having the classes out of order, you do not need to do anything about this.

If you get error messages, scroll back to the first error. If it includes the phrase "file not found", you did not type the correct data filename in `check.sps`:

```
>Error # 31 on line 15 in column 23. Text: yourfilename.dsa
>File not found.
```

```
>No further commands will be executed. Error scan continues.
```

You need to edit `check.sps` again, and make sure the filename is right, including upper/lower case. Then run the spss program again, as shown above.

Make Data Corrections

After the `check.sps` program runs successfully, examine the output file, `check.out`, to see whether there are any problems with missing or incorrect headers.

Blue forms

For blue forms, the only available information is a list of all classes included in the data file. Look at the contents of the output file, `check.out`. Scroll down past the spss command listing, almost to the end, where you will find the list of courses. Here is the beginning of what this list might look like:

DEPT	COURSE	SECTION	INSTRUCT	SEMESTER	YEAR
12	565	1	SMITH	FALL	99

Look through the list to see whether any course/section/instructor is missing. If all is well, proceed to "Run Report Program".

If a course/section/instructor is missing, there are several possible reasons:

- The header sheet for the "missing" course did not have 999 for the batch number. This is a fairly common mistake, and you should always look for it first. To check for this possibility, you will need to use a text editor to scan the data file visually. *Make sure your editor does not wrap long lines!* Blue data files look something like this:

```
502001 2 00011 1 #000>999012565 001FALL 99SMITH $
502001 4 00021 1 #000> 213141122 5677675577$
502001 6 00031 1 #000> 112151231 3522251465$
```

Ignore everything before the > on each line. Notice that the first line has data in the middle of the line, starting with 999 just after the >. This is a header, and the 999 just after the > is the batch number. You recognize headers by the fact that they show the instructor's name, and have nothing on the right side. The next several

lines do NOT show the instructor's name, but have data on the right. These are the students' evaluation records. They have no identifying information on them (the middle section), but have the evaluation information on the right side. Now that you can distinguish headers from student data, search for the header of the "missing" class. Assuming the instructor's name was filled in, it should not be hard to find. If it is missing the 999 after the >, put in the 999. Make sure to remove the three blanks where you add the 999, so the rest of the line is not shifted right. Save the corrected data file, and **run check.sps again**.

- The course did not have a header sheet when it was scanned. This is bad news, since there is no way to correct it. There is nothing on the blue evaluation forms to indicate what class the student is evaluating. Therefore, if there is no header, there is no way to tell which evaluations were intended for the "missing" course. They will be reported for the preceding course that has a header. The only clue will be that some other course may have more evaluations than the number of students in the class.
- The instructor never handed in a set of evaluation forms, or they were misplaced and never scanned. Depending on the importance of getting all evaluations, you may decide to pursue this, or ignore it. If you ignore it and later get the forms, you can get them scanned and run them separately.

If you made any corrections to the data file, **run check.sps again**, to make sure it is now correct. If it is, proceed to "Run Report Program".

Green forms

For green forms, the course number, section and instructor listed on each header is compared with the information entered by each student in the class. If the two do not agree, they are printed so you can peruse the differences and decide whether or not it is a problem. This list of "mismatches" might look something like this:

```

HEADER :      201      1          SMITH      STUDENT : 2 1      1          SMITH
HEADER :      201      1          SMITH      STUDENT : 201      1          JOHN SMITH
HEADER :      201      1          SMITH      STUDENT : 201      0          SMITH

```

On the left side of each line the information on the header is displayed, in this case course 201, section1, instructor Smith; on the right is what the student entered. In these cases (as in the vast majority), the header is correct, and the "mismatch" is due to a student error. On the first record the student did not enter the course number correctly (2 1 rather than 201); on the next the student spelled the instructor's name differently from the header; on the third the student did not enter the section number.

As long as the header information is correct, you do not need to be concerned with "mismatches" due to erroneous entries by the students. The final report only uses identifying information from the headers. What you are looking for in this listing is a consistent pattern of mismatches, for example, a series of students that list their instructor as Jones, while the header says it is Smith. This could indicate that Jones' class is missing a header, or that his header does not have 999 in the batch number. If so, this must be corrected before you run the report, otherwise Jones' evaluations will be mixed in with Smith's.

Open the data file with a text editor and search for Jones. Make sure your editor does not wrap long lines! The relevant section of the file might look like this:

```

534001 1 02711 1 #000>      201          SMITH          1131211121
534001 2 00011 1 #000>      187    FALL    02JONES
534001 4 00021 1 #000>      187    FALL    JONES          2242222232
534001 6 00031 1 #000>      187          JONES          1121111131

```

You can distinguish headers from student records by the fact that headers have complete identifying information (course, semester, year, instructor) and DON'T have evaluation data to the right of the instructor's name. Most tellingly, they should be the first record with a new course, section or instructor listed. Student records may or may not have all the identifying information, but should have data to the right of the instructor's name. In the example above (which is not the beginning of the data file), the second record shown is the first one for Jones, and has no data on the right. This is Jones' header record, but it does not have the 999 batch number, which should be immediately following the >. Remove three blanks and add the 999. It should look like this:

```

534001 1 02711 1 #000>      201          SMITH          1131211121
534001 2 00011 1 #000>999  187    FALL    02JONES
534001 4 00021 1 #000>      187    FALL    JONES          2242222232
534001 6 00031 1 #000>      187          JONES          1121111131

```

Whenever you edit the data file, make sure the fields are in the correct locations. They must line up with the records above and below. If the header record appears to be missing completely, you can copy and paste a header record from another class, and edit the course number, section and instructor's name to the one you need.

In addition to the list of mismatches between header and student records, the end of the check output file lists all the courses found on the data file. You can use this to see whether any classes are missing, which may also indicate missing headers or headers without the 999 batch number.

When you've finished making any necessary data corrections, save the corrected data file and **run check.sps again** until all errors have been eliminated. When the check output is satisfactory, you are ready to run the report.

Run Report Program

Open your (possibly customized) report program with a text editor. Right near the beginning you will see the following lines:

```

TITLE          Put your department name here - Course Evaluations
SUBTITLE       Put Semester here.
set printback=none
COMMENT substitute the name of your data file for yourdata.dsa below.
  Make sure it is spelled correctly, upper/lower case counts.
file type nested file='yourdata.dsa' record=batch 25-27 (a) wild=warn

```

Edit the title and subtitle with appropriate text. Substitute the name of your (corrected) data file for yourdata.dsa. Be sure to it is spelled correctly, and that upper/lower case match the file name. The filename should be in apostrophes. Save the edited program.

To process your data with spss, type:

```
spss -m <reportfile.sps>report.out
```

where `reportfile.sps` is the name of your report program, `evalblue.sps` or `evalgreen.sps`. (If you saved a customized report program under a different name, then use that file.) The output is written to file `report.out`.

The normal response should indicate no errors but may have warnings:

```
End of job: 174 command lines 0 errors 0 warnings 1 CPU seconds
```

If the courses in your data file are not ordered sequentially, you may get a warning:

```
>Warning # 10950
>The working file read by AGGREGATE was not in ascending sequence on
> the break variables. It may be impossible to match the aggregated
> output file with the MATCH FILES command using the keyword BY.
End of job: 180 command lines 0 errors 1 warnings 1 CPU seconds
```

This is not a problem, as long as you don't mind that the report will not be ordered by course number.

Any other error messages indicate a problem with either the data or the program. Data problems should have been corrected during the data checking step. If you did not customize the report program then the most likely error is that you did not enter the correct data file name in the report program. This can generate a LOT of errors, so do not be alarmed. Scroll back to look at the FIRST error. If you made a mistake in the data file name, the first error will include the phrase "file not found":

```
>Error # 31 on line 17 in column 23. Text: yourdatafile.dsa
>File not found.
>No further commands will be executed. Error scan continues.
```

Check the spelling of the filename, including upper/lower case. Make sure it's in apostrophes, and try again.

If you customized the report program, you may have made a mistake, which will generate other errors. The error message may or may not give you a hint of the area where the error is. In any case, go back over the instructions for customizing the program. Look for unmatched apostrophes or quotes in any labels you edited.

After making any needed corrections to the report program, run it through spss again. When you get no errors, you are ready to print the evaluation report.

Print the Report

The evaluation report is too wide to print lengthwise on 8.5x11 paper. You can print it on the wide green and white fanfold paper in OIT's I/O room, or you can print it "landscape" on white paper. The latter requires using the "printeval" script, rather than the ordinary UNIX `lpr` print command. The output can be picked up in the I/O room, A106 LGRC, filed under your username. To print on green and white fanfold paper, type:

```
lpr report.out
```

To print landscape on white paper, type:

```
printeval report.out hppss
```

where `report.out` is the name of the report output file. `hppss` is the name of OIT's HP postscript laser printer.