Stat 604
Assignment 13 - SAS

This assignment reinforces the concepts covered in lectures SAS01 through SAS13. Specifically, it focuses on combining data, sorting data, and producing reports. You will get some additional practice using the match merge process. You will also need to use some of the data manipulation techniques covered earlier in the semester.
Perform each of the exercises listed below. To the extent you have been taught to control it, your output should match that in the PDF file posted on eCampus. Download the files **runners.sas7bdat** and **offences.sas7bdat** from the **Assignment Data Files** on eCampus to your personal SAS library folder for use in this assignment (Be sure this libref is assigned as readonly).

The first data set contains rushing statistics on the 100 NCAA Football players who had the most yards running with the football during the season. The second data set contains information about the 120 NCAA Football teams that had the best offences during the same season. You will be combining the observations in these datasets based on the name of the College or University in each dataset.

Since you are saving these files to a folder for which you are assigning a "read only" libref, you may not alter the original downloaded data sets. You may create and alter temporary versions of these data sets in any way that is necessary to perform the required merge. Even though the merge is the primary objective, it will probably not be the first step in your program. The cardinal rule of "Know Thy Data" will be especially important as you work to combine these two data sets.

1. Begin your program with the required header, filename, and libname statements. As always, your program must include comments in the appropriate places.
2. Using a single merge step create three data sets as described below. Leave all variables in your new data sets unless otherwise specified. Make the common variable that is required for merging the same length in both input data sets. Some variables have the same name in both data sets but do not contain exactly the same data. Add "Team" to the beginning of any variable name from the **offences** data set if the variable has the same name in both input data sets. You will not be required to use the exact same length as the professor for created variables but the lengths should be reasonable to both conserve space and not truncate information. The variable TIES is not used in any computations and is not in any output data sets.
    a. In your permanent personal library create a data set named ALLDATA that contains matching and non-matching rows along with all variables from both data sets. Create a variable that contains a value of Yes if the row has data from the **runners** input data set and No if the row does not contain data from **runners**. This variable will only appear in the ALLDATA data set. Create a variable that represents the percentage of team average yards contributed by each player in the **runners** data set. This percentage is computed by dividing YDSPGM by TEAMYDSPGM.
    b. Create a temporary data set named TEAMDATA containing only the matching rows from both input data sets.
    c. Create a temporary data set named NORUNNERS that contains rows from the **offences** data set with no matching rows from the **runners** data set. This data set must not

contain any of the variables from the **runners** data set and does not need the new percent of team average yards variable either.

A portion of the log from this step is shown below:

```
NOTE: The data set MYLIB.ALLDATA has 136 observations and 22 variables.
NOTE: The data set WORK.NORUNNERS has 36 observations and 10 variables.
NOTE: The data set WORK.TEAMDATA has 100 observations and 21 variables.
```

3. Set up your program to ensure that the output is printed with a landscape layout (wider than it is tall). Ensure the date is printed but that the printed date/time resets instead of using the system time. Suppress the printing of page numbers on the page.
4. Without creating a new dataset, change the order of the observations in the NORUNNERS data set by teamrank.
5. Print the "top 10" teams from this dataset with the variables in the order shown in the sample output on eCampus. Use the same titles and footnotes as shown on eCampus. Note that there is a blank line between the two titles. Add temporary labels to match the eCampus output. Use a temporary format to remove decimal places from the teamydspgm data.
6. Suppress the printing of the date and time on the remaining pages. Note: There is also no footnote on the remaining pages. Research in SAS Help or online the NOPROCTITLE and PROCLABEL commands and use the appropriate one to replicate the output shown on eCampus.
7. Use the ALLDATA data set to print a frequency count and percentage of players from each position (POS) in each class (CL). Include only the statistics shown on eCampus. Supply temporary labels for the two variables.
8. Use the MEANS procedure to reproduce the third page of output as posted on eCampus based on the new variable you created in the TEAMDATA data set created above.
    a. There are two titles. There is a blank line between the two titles.
    b. Analysis statistics are only carried out to two decimal places.
9. Use the TABULATE procedure to create essentially the same report as you did in the previous step as shown on the fourth page of the posted output. Add a line showing the overall statistics at the bottom of the table.
10. Convert the program and log to PDF files and submit them to WebAssign along with your SAS output.