Complete the following tasks using SAS. Paste whatever is requested after each question and submit the completed file via the drop box prior to the assigned due date and time. Be sure to use appropriate comments in the SAS program, and that output has appropriate titles.

1. Using PROC SQL, create a table (SAS data set) that lists each patient in the patients database, along with the name of their primary physician. Use PROC PRINT to show the SAS data set.

/\*Q1 PROC SQL, create a table (SAS data set) that lists each patient in the patients database, along with the name of their primary physician\*/

proc sql ;

create table procdoc as select distinct(patients.id) as id ,

patients.firstname as patient\_name, doctors.lastname as docname from sasuser.patients left

join sasuser.doctors on patients.primmd=doctors.md\_id order by patients.id;

quit;

/\*PROC PRINT to show the SAS data set.\*/

proc print data=work.procdoc (drop=id) ;

title "Table showing Patients name and Doctors name made using PROC SQL";

run;

NOTE: Data file SASUSER.PATIENTS.DATA is in a format that is native to another host, or the file encoding does not match the

session encoding. Cross Environment Data Access will be used, which might require additional CPU resources and might reduce

performance.

NOTE: Data file SASUSER.DOCTORS.DATA is in a format that is native to another host, or the file encoding does not match the session

encoding. Cross Environment Data Access will be used, which might require additional CPU resources and might reduce

performance.

NOTE: Table WORK.PROCDOC created, with 20 rows and 3 columns.

**Table showing Patients name and Doctors name made using PROC SQL**

| Obs | patient\_name | docname |
| --- | --- | --- |
| 1 | Hugh | Fitzhugh |
| 2 | Susan | Fitzhugh |
| 3 | Michael | Premnath |
| 4 | Geoffrey | Colantonio |
| 5 | Celeste | Fitzhugh |
| 6 | Anthony | MacArthur |
| 7 | Lars | Hanratty |
| 8 | Marianne | Colantonio |
| 9 | Frances | Hanratty |
| 10 | Josephine | MacArthur |
| 11 | Karen | Fitzhugh |
| 12 | Albert | Avitable |
| 13 | Riley | Colantonio |
| 14 | Henrik |  |
| 15 | Hannah | Hanratty |
| 16 | Antonio | Fitzhugh |
| 17 | Adam | MacArthur |
| 18 | Shelby | Fitzhugh |
| 19 | Judith | Colantonio |
| 20 | Sarah | Avitable |

1. Using a DATA step, with a MERGE statement, create a SAS data set that lists each patient in the patients database, along with the name of their primary physician. Use PROC PRINT to show the SAS data set.

/\*Q2 DATA step, with a MERGE statement, create a

SAS data set that lists each patient in the patients database,

along with the name of their primary physician. \*/

/\*PROC SORT to sort the Patients Dataset by primmd\*/

proc sort data=sasuser.patients out=work.patients;

by primmd;

run;

/\*PROC SORT to sort the Doctors Dataset by primmd\*/

proc sort data=sasuser.doctors out=work.sorteddoctors;

by md\_id;

run;

/\*MERGE statement, create a SAS data set that lists each patient in the patients database, \*/

data work.docmerge;

merge patients(in=p drop = dob sex zipcode)

sorteddoctors (in=d rename=(md\_id=primmd) keep= md\_id lastname);

by primmd;

if p and d;

run;

/\*PROC PRINT to show the SAS data set.\*/

proc print data=work.docmerge (obs=10) label ;

label id="Patient Id"

primmd ="Doctor Id"

lastname ="Doctor Name"

firstname = "Patients First Name";

title "Table showing Patients name and Doctors name made using MERGE. Showing 1-10 obs ";

run;

NOTE: There were 20 observations read from the data set SASUSER.PATIENTS.

NOTE: The data set WORK.PATIENTS has 20 observations and 7 variables.

NOTE: PROCEDURE SORT used (Total process time):

NOTE: There were 14 observations read from the data set SASUSER.DOCTORS.

NOTE: The data set WORK.SORTEDDOCTORS has 14 observations and 4 variables.

NOTE: PROCEDURE SORT used (Total process time):

NOTE: MERGE statement has more than one data set with repeats of BY values.

NOTE: There were 20 observations read from the data set WORK.PATIENTS.

NOTE: There were 14 observations read from the data set WORK.SORTEDDOCTORS.

NOTE: The data set WORK.DOCMERGE has 20 observations and 4 variables.

**Table showing Patients name and Doctors name made using MERGE. Showing 1-10 obs**

| Obs | Patient Id | Doctor Id | Doctor Name | Patients First Name |
| --- | --- | --- | --- | --- |
| 1 | 1 | 1972 | Fitzhugh | Hugh |
| 2 | 2 | 1972 | Fitzhugh | Susan |
| 3 | 5 | 1972 | Abbott | Celeste |
| 4 | 11 | 1972 | Erickson | Karen |
| 5 | 16 | 1972 | DeLucia | Antonio |
| 6 | 18 | 1972 | Baker | Shelby |
| 7 | 6 | 2322 | MacArthur | Anthony |
| 8 | 10 | 2322 | MacArthur | Josephine |
| 9 | 17 | 2322 | Cohen | Adam |
| 10 | 7 | 3274 | Hanratty | Lars |

/\*Partial Output \*/

1. Using a DATA step, create a lookup table using hash objects from the doctors database. Then, later in that same DATA step, create a SAS data set that lists each patient in the patients database, along with the name of their primary physician (from the hash table). Use PROC PRINT to show the SAS data set.

/\*Q3 DATA step,to create lookup table using hash objects from the doctors database.

DATA STEP to create a SAS data set that lists each patient in the

patients database, along with the name of their primary physician (from the hash table). \*/

data work.dochash (drop=sex dob zipcode primmd md\_id lastname rc);

if 0 then

set sasuser.doctors (keep=md\_id lastname);

if \_N\_=1 then

do;

declare hash doc (dataset:"sasuser.doctors");

doc.definekey ("md\_id");

doc.definedata ("lastname");

doc.definedone();

end;

set sasuser.patients;

rc=doc.find(key:primmd);

if rc=0 then

do;

doctorname=lastname;

end;

else

do;

doctorname=' ';

end;

run;

/\*PROC PRINT to show the SAS data set.\*/

proc print data=work.dochash (drop=id) ;

title "Table showing Patients name and Doctors name made using HASH TABLE";

run;

NOTE: Data file SASUSER.DOCTORS.DATA is in a format that is native to another host, or the file encoding does not match the session

encoding. Cross Environment Data Access will be used, which might require additional CPU resources and might reduce

performance.

NOTE: There were 14 observations read from the data set SASUSER.DOCTORS.

NOTE: There were 20 observations read from the data set SASUSER.PATIENTS.

NOTE: The data set WORK.DOCHASH has 20 observations and 3 variables.

**Table showing Patients name and Doctors name made using HASH TABLE**

| Obs | firstname | doctorname |
| --- | --- | --- |
| 1 | Hugh | Fitzhugh |
| 2 | Susan | Fitzhugh |
| 3 | Michael | Premnath |
| 4 | Geoffrey | Colantonio |
| 5 | Celeste | Fitzhugh |
| 6 | Anthony | MacArthur |
| 7 | Lars | Hanratty |
| 8 | Marianne | Colantonio |
| 9 | Frances | Hanratty |
| 10 | Josephine | MacArthur |
| 11 | Karen | Fitzhugh |
| 12 | Albert | Avitable |
| 13 | Riley | Colantonio |
| 14 | Henrik |  |
| 15 | Hannah | Hanratty |
| 16 | Antonio | Fitzhugh |
| 17 | Adam | MacArthur |
| 18 | Shelby | Fitzhugh |
| 19 | Judith | Colantonio |
| 20 | Sarah | Avitable |

1. Write a SAS DATA step that reads in the film speed data as described. Use PROC PRINT to show the SAS data set.

/\*Q4 SAS DATA step that reads in the film speed\*/

data work.film;

infile '/home/axb96520/my\_courses/steven.lalonde/filmspeed/film.dat' firstobs=3 obs= 34 missover pad;

input @1 SS 3.

@5 CS1 4.3

@10 CS2 3.2

@14 DOC1 4.1

@19 DOC2 2.

@23 speed1 5.

@29 speed2 5.

@35 speed3 5.

@41 speed4 5.;

run;

/\*PROC PRINT to show the SAS data set.\*/

proc print data = work.film;

title "PROC PRINT showing FILM data set";

run;

NOTE: The infile '/home/axb96520/my\_courses/steven.lalonde/filmspeed/film.dat' is:

Filename=/home/axb96520/my\_courses/steven.lalonde/filmspeed/film.dat,

Owner Name=steven.lalonde,Group Name=oda,

Access Permission=-rw-rw-r--,

Last Modified=23Apr2014:16:08:36,

File Size (bytes)=1596

NOTE: 32 records were read from the infile '/home/axb96520/my\_courses/steven.lalonde/filmspeed/film.dat'.

The minimum record length was 45.

The maximum record length was 45.

NOTE: The data set WORK.FILM has 32 observations and 9 variables.

**PROC PRINT showing FILM data set**

| Obs | SS | CS1 | CS2 | DOC1 | DOC2 | speed1 | speed2 | speed3 | speed4 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 450 | 1.40 | 3.0 | 7.5 | 0 | 255.3 | 252.3 | 252.7 | 253.6 |
| 2 | 450 | 1.40 | 3.0 | 7.5 | 10 | 254.0 | 249.5 | 252.5 | 248.3 |
| 3 | 450 | 1.40 | 3.0 | 10.0 | 0 | 253.4 | 254.4 | 251.9 | 253.0 |
| 4 | 450 | 1.40 | 3.0 | 10.0 | 10 | 250.5 | 253.1 | 251.8 | 250.5 |
| 5 | 450 | 1.40 | 3.5 | 7.5 | 0 | 258.8 | 250.6 | 252.9 | 251.9 |
| 6 | 450 | 1.40 | 3.5 | 7.5 | 10 | 254.8 | 255.1 | 251.1 | 252.6 |
| 7 | 450 | 1.40 | 3.5 | 10.0 | 0 | 252.2 | 255.8 | 253.6 | 250.5 |
| 8 | 450 | 1.40 | 3.5 | 10.0 | 10 | 255.8 | 253.4 | 253.2 | 257.3 |
| 9 | 450 | 1.65 | 3.0 | 7.5 | 0 | 262.0 | 261.9 | 257.6 | 256.8 |
| 10 | 450 | 1.65 | 3.0 | 7.5 | 10 | 266.2 | 261.0 | 255.5 | 251.0 |

/\*Partial Output – 1- 10 observations\*/

1. Write a PROC TRANSPOSE step that creates four new observations for every one original observation. The four observations should contain the four different speed variables, put into one variable, appropriately named. Be sure to carry along all the other variables, and to add another variable that keeps track of which speed the new observation came from. Use PROC PRINT to show the resulting SAS data set.

/\*Q5 PROC TRANSPOSE step that creates four new observations for every one original observation.

The four observations (speed1 speed2 speed3 speed4) should contain the four different speed variables,

put into one variable (finalSpeedTVar),

appropriately named. \*/

proc transpose data=work.film

out=work.filmtranspose

name=Speed

prefix=finalSpeedTVar;

by ss cs1 cs2 doc1 doc2;

var speed1 speed2 speed3 speed4;

run;

/\*PROC PRINT to show the resulting SAS data set.\*/

proc print data = work.filmtranspose (obs=12);

title "Usage of PROC TRANSPOSE to make 4 observations out of 1 observation- Showing 1-3 obs";

run;

NOTE: There were 32 observations read from the data set WORK.FILM.

NOTE: The data set WORK.FILMTRANSPOSE has 128 observations and 7 variables.

**Usage of PROC TRANSPOSE to make 4 observations out of 1 observation- Showing 1-3 obs**

| Obs | SS | CS1 | CS2 | DOC1 | DOC2 | Speed | finalSpeedTVar1 |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 450 | 1.4 | 3 | 7.5 | 0 | speed1 | 255.3 |
| 2 | 450 | 1.4 | 3 | 7.5 | 0 | speed2 | 252.3 |
| 3 | 450 | 1.4 | 3 | 7.5 | 0 | speed3 | 252.7 |
| 4 | 450 | 1.4 | 3 | 7.5 | 0 | speed4 | 253.6 |
| 5 | 450 | 1.4 | 3 | 7.5 | 10 | speed1 | 254.0 |
| 6 | 450 | 1.4 | 3 | 7.5 | 10 | speed2 | 249.5 |
| 7 | 450 | 1.4 | 3 | 7.5 | 10 | speed3 | 252.5 |
| 8 | 450 | 1.4 | 3 | 7.5 | 10 | speed4 | 248.3 |
| 9 | 450 | 1.4 | 3 | 10.0 | 0 | speed1 | 253.4 |
| 10 | 450 | 1.4 | 3 | 10.0 | 0 | speed2 | 254.4 |
| 11 | 450 | 1.4 | 3 | 10.0 | 0 | speed3 | 251.9 |
| 12 | 450 | 1.4 | 3 | 10.0 | 0 | speed4 | 253.0 |

1. Write a PROC TRANSPOSE starts with the transposed data set created in the previous problem, and turns it back into the form of the original data set. Be sure to name variables as they were before. Use PROC PRINT to show the resulting SAS data set.

/\*Q6 PROC TRANSPOSE starts with the transposed data set created in the previous problem (work.filmtranspose),

and turns it back into the form of the original data set.\*/

proc transpose data=work.filmtranspose

out=work.filmtransposeOrg (drop = \_LABEL\_ column\_transposed )

name=column\_transposed;

by ss cs1 cs2 doc1 doc2;

id speed;

var finalSpeedTVar1;

run;

/\*PROC PRINT to show the resulting SAS data set.\*/

proc print data = work.filmtransposeOrg (obs=10);

title "Usage of PROC TRANSPOSE to bring back Original FILM data from transposed data. Showing 1-10 obs";

run;

NOTE: There were 128 observations read from the data set WORK.FILMTRANSPOSE.

NOTE: The data set WORK.FILMTRANSPOSEORG has 32 observations and 9 variables.

**Usage of PROC TRANSPOSE to bring back Original FILM data from transposed data. Showing 1-10 obs**

| Obs | SS | CS1 | CS2 | DOC1 | DOC2 | speed1 | speed2 | speed3 | speed4 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 450 | 1.40 | 3.0 | 7.5 | 0 | 255.3 | 252.3 | 252.7 | 253.6 |
| 2 | 450 | 1.40 | 3.0 | 7.5 | 10 | 254.0 | 249.5 | 252.5 | 248.3 |
| 3 | 450 | 1.40 | 3.0 | 10.0 | 0 | 253.4 | 254.4 | 251.9 | 253.0 |
| 4 | 450 | 1.40 | 3.0 | 10.0 | 10 | 250.5 | 253.1 | 251.8 | 250.5 |
| 5 | 450 | 1.40 | 3.5 | 7.5 | 0 | 258.8 | 250.6 | 252.9 | 251.9 |
| 6 | 450 | 1.40 | 3.5 | 7.5 | 10 | 254.8 | 255.1 | 251.1 | 252.6 |
| 7 | 450 | 1.40 | 3.5 | 10.0 | 0 | 252.2 | 255.8 | 253.6 | 250.5 |
| 8 | 450 | 1.40 | 3.5 | 10.0 | 10 | 255.8 | 253.4 | 253.2 | 257.3 |
| 9 | 450 | 1.65 | 3.0 | 7.5 | 0 | 262.0 | 261.9 | 257.6 | 256.8 |
| 10 | 450 | 1.65 | 3.0 | 7.5 | 10 | 266.2 | 261.0 | 255.5 | 251.0 |

1. Use PROC COMPARE to compare the original film speed data set with the one that was transposed twice.

/\*Q7 PROC COMPARE to compare the original film speed data set(work.film) with the one that was transposed twice (work.filmtransposeOrg).\*/

proc compare data=work.film comp=work.filmtransposeOrg listall;

run;

NOTE: There were 32 observations read from the data set WORK.FILM.

NOTE: There were 32 observations read from the data set WORK.FILMTRANSPOSEORG.

The COMPARE Procedure

Comparison of WORK.FILM with WORK.FILMTRANSPOSEORG

(Method=EXACT)

Data Set Summary

Dataset Created Modified NVar NObs

WORK.FILM 27APR16:02:49:37 27APR16:02:49:37 9 32

WORK.FILMTRANSPOSEORG 27APR16:03:09:51 27APR16:03:09:51 9 32

Variables Summary

Number of Variables in Common: 9.