Questions 1 through 12 are worth 2 points each.

1. The SAS statement that must follow a data statement that describes how to read data values from the data line when using raw data to create a SAS data set is **input**.

2. When raw data is available in a file (say, as a text file), an **in**file statement in the SAS program tells SAS the location of the file and the name of the file.

3. To create a new SAS data set using observations from an existing SAS data set, a data statement must be followed by a **set** statement.

4. Suppose SAS data set named **scores** is available in a folder named in a libname statement with the libref **mylib**. Complete the following data option needed to access this data set in a SAS proc statement: data= **mylib.scores**

5. Give the symbols used on the input statement to move the pointer to the 13th column: **@13**

6. Give a symbol used at the end of an input statement to hold the current data line in order to read more data from the same line: **@**

7. Describe the number stored in the variable DoB when the date 08/24/98 is input to DoB using the informat **mmdyyyy**? **# of days from 1/1/60**

8. The **output** statement in a SAS data step tells SAS to write the current observation from the PDV to the SAS data set at that point in the data step.

9. To use a do loop in a data step to compute, say, the logarithms of several numeric variables, these variables must first be declared in an **array** statement.

10. Two statements that specify whether they are categorical variables or continuous (analysis) variables (not label or format statements) that are needed in most SAS procedure steps are: **var**; **class**

11. To process subsets of observations in a data set by some SAS procedures, without first creating a subset of the data set, use a **where** statement in the proc step.

12. Write SAS programming statements that involve a do-end loop to do the following: increase the value of the variable **Rate** by 20% and add 2.45 to the value of the variable **Score** only when the value of the variable **Type** is equal to 'B'.

```sas
if Type = 'B' then do;
    Rate = 1.2 * Rate;
    Score = Score + 2.45;
end;
```
13. Study the following SAS program and answer the questions given below (4 points per each part). Note that the line numbers are for reference only.

```sas
1 data sales;
2 input Region $ Product $ Quantity Price;
3 Revenue=Quantity*Price;
4 data=sales;
   Eastern ne-101 721 5625
   Southern bl-319 315 24600
   Southern kt-800 182 56750
   Northern ne-101 312 5575
   Eastern ne-101 152 5490
   Eastern ne-101 230 5670
   Northern ne-101 610 5675
   Southern bl-319 95 23675
   Eastern kt-800 320 55250
   Northern kt-800 110 55450
   Eastern bl-319 250 25975
   Southern kt-800 257 57150
   Northern ne-101 312 5675
   Eastern ne-101 521 5690
   Eastern ne-101 325 5670
   Southern bl-319 176 24100
   Northern kt-800 132 56450
   Southern ne-101 115 5750
   Eastern kt-800 415 54750
   Northern ne-101 426 5490
   Eastern bl-319 170 24826
   Southern ne-101 310 5560
;
6 proc means data=sales;
7 class Region Product;
8 var Quantity Revenue;
9 output out=stats1 min=MinQty MinRev max=MaxQty MaxRev;
10 run;
11 proc print data=stats;
12 run;
```

(a) If you know that the data were available in the text file ‘sales.txt’ in the folder U:\Documents\project, write a statement needed in the program in order to read the data from this file (instead of including the data instream).

```
infile 'U:\Documents\project\sales.txt';
```

(b) Is the SAS dataset named sales temporary or permanent? How many variables and observations are in this data set? **Temporary** 5 22

(c) Show a sketch of the output from proc means in line 6. (actual values are not needed; but show what statistics are computed, where they are printed, and for which variables)

<table>
<thead>
<tr>
<th>Region</th>
<th>Product</th>
<th>Variable</th>
<th>N</th>
<th>Mean</th>
<th>StdDev</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eastern</td>
<td>bl-319</td>
<td>Quantity</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Revenue</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td></td>
<td>kt-800</td>
<td>Quantity</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Revenue</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Northern</td>
<td>bl-319</td>
<td>Quantity</td>
<td>x</td>
<td></td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Revenue</td>
<td>x</td>
<td></td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td></td>
<td>kt-800</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(d) Is the SAS data set named stats temporary or permanent? How many observations are in this data set? **Temporary** 16
(e) Following is a line extracted from the output of proc print. Describe what the numbers printed under the 4 variable names: MinQty, MinRev, MaxQty, MaxRev represent.

<table>
<thead>
<tr>
<th>Obs</th>
<th>Region</th>
<th>Product</th>
<th><em>TYPE</em></th>
<th><em>FREQ</em></th>
<th>MinQty</th>
<th>MinRev</th>
<th>MaxQty</th>
<th>MaxRev</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>kt-800</td>
<td></td>
<td>1</td>
<td>6</td>
<td>110</td>
<td>6099500</td>
<td>415</td>
<td>22721260</td>
</tr>
</tbody>
</table>

The statistics are computed for the 6 observations for Product kt-800. Minima for Variables - Quantity - Revenue

Maxima for Variables - Quantity - Revenue

14. Show the values for the variable weight that will be stored in the SAS dataset chicken for each of the 67 observations (6 points):

```sas
data rats;
input Weight 5.2;
datalines;
 125
 125
 1.25
 1.25
 1.25
 1.25
 125
 125;
```

15. A SAS data set named parts is to be created using the input statement:

```sas
input Region $ City $ Product $ Inventory Cost;
```

Data from 4 regions: Midwest, Northeast, South and West were collected for this study. Write SAS statements (using correct syntax) needed in the data step to do the task indicated. (3 points for each part)

(a) To omit observations with missing values for either Inventory or Cost from the data set.

```sas
if Inventory = . | Cost = . then delete;
```

(b) To specify length of 10 bytes for each of the variables City and Product, respectively.

```sas
length City Product $ 10;
```

(c) To create a new numeric variable called Value containing values computed by multiplying value of Inventory by the value of Cost.

```sas
Value = Inventory * Cost;
```

(d) To create labels 'Total Inventory' and 'Current Value (in $1000's)' for the variables Inventory and Value.

```sas
label Inventory = "Total Inventory";
label Value = "Current Value (in $1000's)";
```

(e) To specify appropriate dollar formats for the variables Cost and Value.

```sas
format Cost Value dollar16.2;
```
16. Dr. Hickenlooper has data from a skin disease study where patients receive a new drug or a placebo at several clinics. Skin thickening, skin mobility, and an assessment score were recorded at the beginning of the study and again after 6 months of treatment. The data are:

<table>
<thead>
<tr>
<th>Clinic No.</th>
<th>Patient ID</th>
<th>Treatment</th>
<th>Thick1</th>
<th>Thick6</th>
<th>Mobility1</th>
<th>Mobility6</th>
<th>Assess1</th>
<th>Assess6</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>14</td>
<td>D</td>
<td>17</td>
<td>18</td>
<td>191</td>
<td>192</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>40</td>
<td>18</td>
<td>P</td>
<td>26</td>
<td>24</td>
<td>255</td>
<td>211</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>41</td>
<td>25</td>
<td>D</td>
<td>26</td>
<td>30</td>
<td>210</td>
<td>310</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>50</td>
<td>29</td>
<td>P</td>
<td>19</td>
<td>9</td>
<td>432</td>
<td>398</td>
<td>5</td>
<td>1</td>
</tr>
</tbody>
</table>

Note: Value of Treatment is 'D' if Drug, or 'P' if Placebo

(a) Write SAS statements to create a SAS dataset named skindata. Use variable names as given above and assume that data will be included instream with data values entered separated by blanks. (5 points)

```sas
data skindata;
input Clinic Patient Treatment Thick1 Thick6 Mobility1 Mobility6 Assess1 Assess6;
datalines;
......
data entered in list format exactly as shown above
```

(b) Write SAS statement(s) to be added to the above data step to create a new variable Index for each patient that indicates an improvement measure as the total of the following points. Patients whose skin thickening has decreased get 1 point, skin mobility has increased get 2 points, and assessment score has increased get 3 points. Thus each patient may be assigned 0, 1, 2, 3, 4, 5 or 6 points as the value of Index. (5 points)

```
if Thick6 < Thick1 then Index = Index + 1;
if Mobility6 > Mobility1 then Index = Index + 2;
if Assess6 > Assess1 then Index = Index + 3;
Index = Index + (Thick6 < Thick1) + 2*(Mobility6 > Mobility1) + 3*(Assess6 > Assess1)
```

(c) Dr. Hickenlooper wants to do a separate analysis for patients from clinic 67. Write SAS statements needed to create a SAS data set named clinic67, which is a subset of skindata, containing records of patients from clinic 67 who have at least a value of 3 for Index. (5 points)

```
data clinic67;
set skindata;
if Clinic = 67 & Index >= 3;
run;
```
(d) Dr. Hickenlooper would also like to use a SAS procedure to compute summary statistics (sample size, mean, standard deviation of the mean, and the range), for the variables Thick1, Thick6, Mobility1, Mobility6, Assess1, and Assess6, computed separately for each clinic within each treatment type. Write SAS statements necessary to accomplish this (without sorting the data or creating subsets of the dataset). (5 points)

```
proc means data=skindata n mean stderr range;
class Treatment Clinic;
  var Thick1 Thick6 Mobility1 Mobility6 Assess1 Assess6;
run;
```

17. Display the printed output produced from executing the following SAS program. Show what is in the program data vector immediately after processing the first line of data. (5 points)

```
data carmart;
  input Dept $ Id $ P12 P13 P14;
drop P12 P13 P14;
  if Year=2012 then Year=2012; output;
  if Year=2013 then Year=2013; output;
  if Year=2014 then Year=2014; output;
datalines;
  parts  176  3500 2500  800
  parts  217  2644 3500  3000
  tools  124  5672 6100  7400
  tools  45   1253 4698  9345
  tools  26   9050 5450  8425
  repairs 142 ;
proc print;
run;
```

```
  Dept  Id  P12  P13  P14  Year  Sales  _N_  _ERROR_
parts  176  3500 2500  800  2014   800   1   0
```

---

5
18. Display the printed output produced by executing the following SAS program. Show what is in the program data vector at the point the first observation is to be written to the SAS data set. (5 points)

```sas
data stn6;
input Y1 Y2 @;
Y3=Y2**2-5.0;
Y4=sqrt(Y1)/2+1;
drop Y1 Y2;
datalines;
4 -3 0 2 9 . 16 5 1 12
;
proc print data=stn6;
run;
```

<table>
<thead>
<tr>
<th>Y1</th>
<th>Y2</th>
<th>Y3</th>
<th>Y4</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>-3</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

19. Display the printed output produced by executing the following SAS program. Show what is in the program data vector immediately after processing the first line of data. (5 points)

```sas
data jmart;
input Store $ Item1-Item5;
array Items (5) Item1-Item5 ;
drop ItemNo;
Sales=0;
do ItemNo=1 to 5;
  if Items(ItemNo)=. then Items(ItemNo)=0;
  Sales+Items(ItemNo);
end;
datalines;
PD12 14 6 7 10 2
HJ23 5 . 12 3 14
;
proc print data=jmart;
run;
```

<table>
<thead>
<tr>
<th>Store</th>
<th>Item1</th>
<th>Item2</th>
<th>Item3</th>
<th>Item4</th>
<th>Item5</th>
<th>Sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>PD12</td>
<td>14</td>
<td>6</td>
<td>7</td>
<td>10</td>
<td>2</td>
<td>39</td>
</tr>
<tr>
<td>HJ23</td>
<td>5</td>
<td>.</td>
<td>12</td>
<td>3</td>
<td>14</td>
<td>34</td>
</tr>
</tbody>
</table>

Output -

<table>
<thead>
<tr>
<th>PDV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Store</td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td>PD12</td>
</tr>
<tr>
<td>HJ23</td>
</tr>
</tbody>
</table>