Midterm Examination

For each question below, please enter your answer in the shaded area below the question.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. | SAS programs are comprised of 1 or more “steps”. There are 2 types of “steps”, one type are PROC (i.e. procedure) steps. What is the other type of “step”? | |  |  |  |
|  | **DATA** steps | |  |  | / 1 pt |
|  |  | |  |  |  |
| 2. | Which of the following are ways to submit a SAS program?  For each choice, circle choices that work, and put an X through those that don’t work. | |  |  |  |
|  | Press Press Click Press  F3 F4 ‘Submit’ Button Ctl-Alt-Delete  (Running Man)  Works X Works X | |  |  | / 1 pt |
|  |  | |  |  |  |
| 3. | Write the words “My comment” using the syntax for both (1) a Comment Statement, and (2) a Block Comment. | |  |  |  |
|  | 1. Using Comment Statement syntax: \* My comment ; 2. Using Block Comment syntax: /\* My comment \*/ | |  |  | / 2 pts |
|  |  | |  |  |  |
| 4. | True or False: Every SAS statement must be written on a single line in a program (i.e. it cannot span across multiple lines). | |  |  |  |
|  | False | |  |  | / 1 pt |
|  |  | |  |  |  |
| 5. | Which values below are valid for a numeric variable?  For each choice, circle valid values, and put an X through those that aren’t valid. | |  |  |  |
|  | $7.23 -7 -7.5E2 7,239  X Valid Valid X | |  |  | / 1 pt |
|  |  | |  |  |  |
| 6. | Suppose a SAS variable named ‘BirthDate’ contains a numeric value of -2. What date does this value represent? | |  |  |  |
|  | December 30, 1959 | |  |  | / 1 pt |
|  |  | |  |  |  |
| 7. | Which procedure (i.e. PROC) is used to display the metadata (i.e. the descriptor portion) of a SAS data set? | |  |  |  |
|  | **PROC** **CONTENTS** | |  |  | / 1 pt |
|  |  | |  |  |  |
| 8. | Which SAS data sets named below are considered ‘permanent’ data sets?  Circle those that are permanent, and put an X through those which aren’t permanent. | |  |  |  |
|  | ABC.Sales Stores Finance.Profits WORK.Jobs  Permanent X Permanent X | |  |  | / 1 pt |
|  |  | |  |  |  |
| 9. | Which of the following SAS statements correctly assigns the libref ‘Study’ to the folder named ‘Data’ at the path “C:\Project XYZ\Data”?   1. LIBREF Data "C:\Project XYZ\Study"; 2. LIBNAME Study "C:\Project XYZ\Data"; 3. LIBNAME Data "C:\Project XYZ\Study"; 4. LIBREF Study "C:\Project XYZ\Data"; | |  |  |  |
|  | B | |  |  | / 1 pt |
|  |  | |  |  |  |
| 10. | SAS processes the DATA step in 2 phases. The first phase is the ‘Compilation’ phase. What is the name of the 2nd phase? | |  |  |  |
|  | Execution | |  |  | / 1 pt |
|  |  | |  |  |  |
| 11. | What are the names of the 2 automatic variables which are always in the Program Data Vector (PDV)? | |  |  |  |
|  | \_N\_ and \_ERROR\_ | |  |  | / 1 pt |
|  |  | |  |  |  |
| 12. | The following SAS code will perform which of the following:  **DATA** A.B;  SET C.D;  **RUN**;   1. Creates a data set named ‘A’ in the ‘B’ library from data set ‘C’ in the ‘D’ library 2. Creates a data set named ‘B’ in the ‘A’ library from data set ‘D’ in the ‘C’ library 3. Creates a data set named ‘C’ in the ‘D’ library from data set ‘A’ in the ‘B’ library 4. Creates a data set named ‘D’ in the ‘C’ library from data set ‘B’ in the ‘A’ library   Write the letter of your answer in the box below. | |  |  |  |
|  | B | |  |  | / 1 pt |
|  |  | |  |  |  |
| 13. | In the SAS code below, a temporary variable named ‘LastObs’ is created in the PDV.  **DATA** WORK.Illus;  SET CanSrc.Scans END = LastObs;  **RUN**;  If there are 16 observations in the ‘Scans’ data set, what is the value of ‘LastObs’ for the 7th observation in the PDV? | |  |  |  |
|  | 0 | |  |  | / 1 pt |
|  |  | |  |  |  |
| 14. | When creating an In-Stream data set, what is the name of the last statement before the rows containing the data? | |  |  |  |
|  | DATALINES | |  |  | / 1 pt |
|  |  | |  |  |  |
| 15. | True or False: When creating an In-Stream data set, a “RUN;” statement is needed after the lines of code containing the data value. | |  |  |  |
|  | False | |  |  | / 1 pt |
|  |  | |  |  |  |
| 16. | Data in a Microsoft Excel workbook may be imported into SAS by using either a LIBNAME statement or using PROC IMPORT. If using a LIBNAME statement, an engine should be specified. Write the name of one of the 3 engines available for such a task. | |  |  |  |
|  | XLSX, EXCEL, or PCFILES | |  |  | / 1 pt |
|  |  | |  |  |  |
| 17. | Suppose you are using PROC IMPORT to import data into SAS from a MS Excel workbook containing multiple worksheets. By default the first worksheet will be imported. What is the name of the statement which is needed to specify an import from a different sheet (e.g. the 3rd worksheet)? | |  |  |  |
|  | The SHEET= statement | |  |  | / 1 pt |
|  |  | |  |  |  |
| 18. | Suppose your task is to import data from an Excel workbook named ‘Lab Data.xls’ into a temporary SAS data set named ‘MyLabs’.  PROC IMPORT DATAFILE = "C:\Lab Data.xls"  [missing-syntax-here]  DBMS = XLS REPLACE;  RUN;  Write the missing syntax needed to successfully accomplish the task. | |  |  |  |
|  | OUT=WORK.MyLabs | |  |  | / 1 pt |
|  |  | |  |  |  |
| 19. | What character is used in an INPUT statement to designate that a variable is a character variable? | |  |  |  |
|  | $ | |  |  | / 1 pt |
|  |  | |  |  |  |
| 20. | Column Input and Formatted Input are 2 input styles used to import column-aligned text data. Which of the 2 styles is needed to correctly import non-standard data? | |  |  |  |
|  | Formatted Input | |  |  | / 1 pt |
|  |  | |  |  |  |
| 21. | Write an INPUT statement which will create the displayed output SAS data set from the displayed source raw data file (both shown below).  Recall that numeric variables are displayed as right-justified, and character variables are displayed as left-justified. | |  |  |  |
|  | **Source Raw Data File**   |  | | --- | | 1 1 2 | | 1---5----0----5----0 | | 101518-59-6142FRI | | 102942-27-7358FMI | | 103754-20-4436FVT | | 104007-70-3870MDE | |  | | **Output SAS Data Set**   |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | SubjID | SSN | SexCd | StateCd | | 1 | 101 | 518-59-6142 | F | RI | | 2 | 102 | 942-27-7358 | F | MI | | 3 | 103 | 754-20-4436 | F | VT | | 4 | 104 | 007-70-3870 | M | DE | |  |  |  |
|  | INPUT SubjID **1**-**3**  SSN $ **4**-**14**  SexCd $ **15**  StateCd $ **16**-**17**;  This show one solution:  1 pt for correctly designating the character variables  1 pt for correctly specifying the columns to read for each variable  1 pt for everything else needed (INPUT keyword, variable names, semicolon, etc.) | |  |  | / 3 pts |
|  |  | |  |  |  |
| 22. | From the question above, what are the lengths in bytes of the variables named 1) ‘SubjID’, 2) ‘SSN’, and 3) ‘SexCd’ in the output SAS data set? | |  |  |  |
|  | 1. Length of SubjID: 8 bytes 2. Length of SSN: 11 bytes 3. Length of SexCd: 1 byte | |  |  | / 3 pts |
|  |  | |  |  |  |
| 23. | Write an INPUT statement which will create the displayed output SAS data set from the displayed source raw data file (both shown below).  Note: The values for the variable ‘BirthDt’ are numeric and unformatted. The value shown of -12986 (in the first observation) comes from the characters ‘06-12-1924’.  **Source Raw Data File**   |  | | --- | | 1 1 2 2 3 3 4 4 5 5 6 | | 1---5----0----5----0----5----0----5----0----5----0----5----0 | | 41George H. W. Bush06-12-1924Republican 19891993$200,000 | | 42Bill Clinton 08-19-1946Democratic 19932001$200,000 | | 43George W. Bush 07-06-1946Republican 20012009$400,000 | | 44Barack Obama 08-04-1961Democratic 20092017$400,000 | |  |   **Output SAS Data Set**   |  |  |  |  | | --- | --- | --- | --- | |  | FullName | Salary | BirthDt | | 1 | George H. W. Bush | 200000 | -12986 | | 2 | Bill Clinton | 200000 | -4883 | | 3 | George W. Bush | 400000 | -4927 | | 4 | Barack Obama | 400000 | 581 | | |  |  |  |
|  | INPUT FullName $ **3**-**19**  @**49** Salary DOLLAR8.  @**20** BirthDt MMDDYY10.;  1 pt for correct pointer positioning (e.g. @49)  1 pt for correct informat names and widths  1 pt for correctly ordering variables 1 pt for everything else (INPUT, ;, etc.) | |  |  | / 4 pts |
|  |  | |  |  |  |
| 24. | True or False: When reading delimited data, by default SAS considers multiple delimiters as if they were a single delimiter. | |  |  |  |
|  | True | |  |  | / 1 pt |
|  |  | |  |  |  |
| 25. | Write (1) the INFILE statement and (2) the INPUT statement which will create the displayed output SAS data set from the displayed source raw data file (both shown below). Assume the source data file is stored at the following location: ‘C:\Weight.txt’. | |  |  |  |
|  | **Source Raw Data File**   |  | | --- | | 1 1 2 | | 1---5----0----5----0--- | | Miller|164|||169 | | Johnson|150||154 | | Jackson|113|114|114|112 | | Harris|147|142||149 | | **Output SAS Data Set**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | Name | WtMn0 | WtMn1 | WtMn2 | WtMn3 | | 1 | Miller | 164 | . | . | 169 | | 2 | Johnson | 150 | . | 154 | . | | 3 | Jackson | 113 | 114 | 114 | 112 | | 4 | Harris | 147 | 142 | . | 149 | |  |  |  |
|  | INFILE 'C:\Weight.txt' DELIMITER = '|' DSD MISSOVER;  INPUT Name $  WtMon0 WtMon1 WtMon2 WtMon3;  1 pt for specifying 'C:\Weight.txt' 1 pt for specifying MISSOVER  1 pt for specifying DELIMITER = '|' 1 pt for specifying DSD  1 pt for all else (INFILE and INPUT keywords, variable names, $, ;, etc.) | |  |  | / 5 pts |
|  |  | |  |  |  |
| 26. | Write (1) the INFILE statement and (2) the INPUT statement which will create the displayed output SAS data set from the displayed source raw data file (both shown below).  Assume the source data file is stored at the following location: ‘C:\Plants.txt’.  **Source Raw Data File**   |  | | --- | | 1 1 2 2 3 3 4 4 5 5 6 | | 1---5----0----5----0----5----0----5----0----5----0----5----0 | | Parthenocissus tricuspidata,$4.99,Eastern Asia | | Picea pungens,$109.95,Rocky Mountains | | Epipremnum aureum,$32.59,Mo’orea | |  |   **Output SAS Data Set**   |  |  |  | | --- | --- | --- | |  | Plant | Cost | | 1 | Parthenocissus tricuspidata | 4.99 | | 2 | Picea pungens | 109.95 | | 3 | Epipremnum aureum | 32.59 | | |  |  |  |
|  | One solution is:  INFILE 'C:\Plants.txt' DELIMITER=',';  INPUT Plant :$27.  Cost :DOLLAR.;  1 pt for using the Colon Format Modifier (:)  1 pt for specifying informats correctly  1 pt for all other code correct (e.g. DELIMITER=’,’ etc) | |  |  | / 3 pts |
|  |  | |  |  |  |
| 27. | Write an INPUT statement which will create the displayed output SAS data set from the displayed source raw data file (both shown below). | |  |  |  |
|  | **Source Raw Data File**   |  | | --- | | 1 1 2 2 3 | | 1---5----0----5----0----5----0 | | 101 M 56 102 F 65 103 F 58 | | 201 M 59 202 F 23 203 M 53 | |  | | **Output SAS Data Set**   |  |  |  |  | | --- | --- | --- | --- | |  | SubjID | SexCd | Age | | 1 | 101 | M | 56 | | 2 | 102 | F | 65 | | 3 | 103 | F | 58 | | 4 | 201 | M | 59 | | 5 | 202 | F | 23 | | 6 | 203 | M | 53 | |  |  |  |
|  | One solution is:  INPUT SubjID  SexCd $  Age @@;  1 pt for specifying $; 1 pt for specifying @@, 1 pt for all else | |  |  | / 3 pts |
|  |  | |  |  |  |
| 28. | Write an INPUT statement which will create the displayed output SAS data set from the displayed source raw data file (both shown below).  Assume that the option N = **3** is specified in the INFILE statement. | |  |  |  |
|  | **Source Raw Data File**   |  | | --- | | 1 1 | | 1---5----0----5 | | Phelps | | 8 | | 2008 | | Spitz | | 7 | | 1972 | |  | | **Output SAS Data Set**   |  |  |  |  | | --- | --- | --- | --- | |  | Olympian | Year | Golds | | 1 | Phelps | 2008 | 8 | | 2 | Spitz | 1972 | 7 | |  |  |  |
|  | INPUT Olympian $  // Year  #**2** Golds;  1 pt for using // or # to move down the Input Buffer  1 pt for using # to move up the Input Buffer  1 pt for all other code correct (variable names, etc.) | |  |  | / 3 pts |
|  |  | |  |  |  |
| 29. | Which symbol holds a record in the input buffer until the next iteration of the DATA step?   1. / 2. # 3. @ 4. : | |  |  |  |
|  | @ | |  |  | / 1 pt |
|  |  | |  |  |  |
| 30. | Which of the following SAS statements will successfully change the name of a variable named *Age* to *AgeYrs*?   1. RENAME Age = AgeYrs; 2. RENAME = Age TO AgeYrs; 3. RENAME 'Age' = 'AgeYrs'; 4. RENAME = (Age) = (AgeYrs); | |  |  |  |
|  | A | |  |  | / 1 pt |
|  |  | |  |  |  |
| 31. | Which of the following SAS statements will successfully use a numeric variable named ‘AgeYrs’ to create a character variable named ‘AgeChar’ with values displayed using 1 decimal place (e.g. 35.7):   1. AgeChar = INPUT(AgeYrs, **2.1**); 2. AgeChar = INPUT(AgeYrs, **4.1**); 3. AgeChar = PUT(AgeYrs, **2.1**); 4. AgeChar = PUT(AgeYrs, **4.1**); | |  |  |  |
|  | D | |  |  | / 1 pt |
|  |  | |  |  |  |
| 32. | Suppose you will be creating a variable named *Mountain* which needs have a length of 16 bytes to contain the value ‘Uncompahgre Peak’. Write a LENGTH statement which will successfully accomplish the task. | |  |  |  |
|  | LENGTH Mountain $ **16**; | |  |  | / 1 pt |
|  |  | |  |  |  |
| 33. | Which statement will successfully assign a label of ‘14er Peak’ to the variable named *Mountain*.   1. RENAME Mountain = '14er Peak$'N; 2. LABEL = '14er Peak' TO Mountain; 3. LABEL Mountain = '14er Peak'; 4. RENAME Mountain '14er Peak'; | |  |  |  |
|  | C | |  |  | / 1 pt |
|  |  | |  |  |  |
| 34. | The following VALUE statement is used to create a format named ‘TempFRange’.  VALUE TempFRange  LOW - **49** = 'Cold'  **50** - < **60** = 'Cool'  **60** - < **80** = 'Moderate'  **80** - **100** = 'Hot';  This format is applied to a variable named ‘TempF’ by using the following statement:  FORMAT TempF TempFRange.;  Enter how the values of ‘TempF’ will be displayed in the table below: | |  |  |  |
|  | **SAS Data Set**   |  |  |  | | --- | --- | --- | |  | TempF | TempF Displayed As: | | 1 | 72 | Moderate | | 2 | -5 | Cold | | 3 | 103 | 103 | | 4 | . | . | | 5 | 80 | Hot | | 6 | 49.5 | 49.5 | | |  |  | / 3 pts |
|  |  | |  |  |  |
| 35. | Write a VALUE statement to be used in a PROC FORMAT step which will create a character format named ‘Grades’ which meets the following specifications:  Value(s) Label  A, B, C, D Pass  F Fail  Any other value \* CHECK \* | |  |  |  |
|  | VALUE $Grades  'A','B','C','D' = 'Pass'  'F' = 'Fail'  OTHER = '\* CHECK \*';  1 pt for correct format name (including $ sign)  1 pt for correct value-label pairs  1 pt for using keyword OTHER | |  |  | / 3 pts |
|  |  | |  |  |  |
| 36. | In order to use permanent formats, SAS must be directed to the catalog which contains the permanent formats. Suppose you’ve stored permanent formats in a catalog named ‘StudyFmts’ in a library with a libref of ‘Study18’.  Write the OPTIONS statement which will successfully direct SAS to search this catalog for formats. | |  |  |  |
|  | OPTIONS FMTSEARCH = (Study18.StudyFmts); | |  |  | / 1 pt |
|  |  | |  |  |  |
| 37. | Your task is to create a variable named ‘View’ using conditional logic based on the values of the ‘Row’ variable using the following requirements: | |  |  |  |
|  | Row View  A to D Excellent  E Good  Anything else Fair | **SAS Data Set**   |  |  |  |  | | --- | --- | --- | --- | |  | Name | Row | View | | 1 | Alice | C | Excellent | | 2 | Ted | H | Fair | | 3 | Salina | E | Good | | 4 | … | … | … | |  |  |  |
|  | IF Row IN('A', 'B', 'C', 'D') THEN View = 'Excellent';  ELSE IF Row = 'E' THEN View = 'Good';  ELSE View = 'Fair';  1 pt for correctly using IF/THEN or IF/THEN/ELSE statements  1 pt for correctly specifying the IF expressions  1 pt for correctly specifying the THEN statements | |  |  | / 3 pts |
|  |  | |  |  |  |
| 38. | Suppose a SAS data set named ‘WORK.Source’ contains the ‘Month’ and ‘SnowIn’ variables and the following SAS code is executed:  **DATA** WORK.Stats;  SET WORK.Source;  Inches + SnowIn;  MonCount + **1**;  Snowed5In = (SnowIn >= **5**);  **RUN**;  Fill in the values for the ‘Inches’, ‘MonCount’, and ‘Snowed5In’ variables below. | |  |  |  |
|  | **SAS Data Set ‘WORK.Stats’**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | Month | SnowIn | Inches | MonCount | Snowed5In | | 1 | JAN | 4 | 4 | 1 | 0 | | 2 | FEB | 10 | 14 | 2 | 1 | | 3 | MAR | 7 | 21 | 3 | 1 | | 4 | APR | 2 | 23 | 4 | 0 | | … | … | … | … | … | … | | |  |  | / 3 pts |
|  |  | |  |  |  |
| 39. | What value would be assigned to X if the following statement were submitted today?  X = DAY(TODAY()); | |  |  |  |
|  | 16 | |  |  | / 1 pt |
|  |  | |  |  |  |
| 40. | What value is returned by the following expression: ROUND(**84.259**, **0.1**)   1. 84.3 2. 84 3. 84.26 4. 80 | |  |  |  |
|  | A | |  |  | / 1 pt |
|  |  | |  |  |  |
| 41. | Below is a temporary SAS data set containing variables named ‘Player’ and ‘Pct’.   |  |  |  | | --- | --- | --- | |  | Player | Pct | | 1 | Ty Cobb | (98.2%) | | 2 | Honus Wagner | (95.1%) | | 3 | Babe Ruth | (95.1%) | | … | … | … |   Each of the following 4 assignment statements creates a new variable.  Vbl1 = SCAN(Player, **2**);  Vbl2 = FINDC(LOWCASE(Player), 'b');  Vbl3 = COMPRESS(Pct, '()');  Vbl4 = SUBSTR(Pct, **2**, **2**);  Based on the code above, enter values for the 4 new variables into the data set below: | |  |  |  |
|  | **SAS Data Set**   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | |  | Player | Pct | Vbl1 | Vbl2 | Vbl3 | Vbl4 | | 1 | Ty Cobb | (98.2%) | Cobb | 6 | 98.2% | 98 | | 2 | Honus Wagner | (95.1%) | Wagner | 0 | 95.1% | 95 | | 3 | Babe Ruth | (95.1%) | Ruth | 1 | 95.1% | 95 | | … | … | … | … | … | … | … | | |  |  | / 4 pts |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  | Examination Score: | |  |  | / 70 pts |