SAS (Statistical Analysis System), a statistical software package, is available on GEMINI. SAS is a tool for statistical analysis, data management, and report writing. A SAS job is divided into DATA and PROC (PROCedure) steps. In DATA steps, you describe your data to SAS and create any needed variables. In PROC steps, you tell SAS to perform specific analyses on your data.

You can execute a SAS program under VMS in interactive, non-interactive, or batch mode. Interactive mode is useful when you are learning a new feature or testing a short program. In non-interactive mode, you enter the command SAS followed by the name of a file containing SAS statements. When you use interactive or non-interactive mode, your terminal locks while SAS executes the program. If your SAS program requires a long time or large amounts of memory to execute, you may want to use batch mode. Some steps for running a SAS program in batch mode are shown below.

**BATCH MODE**

1. Creating a SAS program `EXAMPLE.SAS`

```sas
$ EVE EXAMPLE.SAS

/* This is an example of a SAS program;
   This is a comment statement ;
   Every SAS statement ends up with semi-colon;
   This is another kind of comment statement */;
DATA STUDY1; * DATA step always starts with DATA */;
   INFILE EXPER; /* Input file is EXPER.DAT */;
   INPUT TREAT $1 Y 4-10 .2; /* Specifying verse and format */;

PROC PRINT; /* PROC step usually starts with PROC statement */;
   TITLE 'The Original Data List'; /* A title goes with
      PRINT procedure */;

PROC ANOVA; /* Another PROC step */;
   CLASS TREAT;
   MODEL Y = TREAT;
   MEANS TREAT / DUNCAN; /* Do multiple comparison */;
   TITLE 'One-way Analysis of Variance and Means Comparisons';
      [CTRL/Z]
```

$
2. Creating a data file called EXPER.DAT

   $ EVE EXPER.DAT

   A 12.15
   B 20.00
   (TYPE IN THE REST OF DATA) [CTRL/Z]

   $

   It is very important that the length of a record should be equal or longer than those specified in the SAS program. A single period (.) represents a missing value. If missing values occur at the end of a record, SAS will go to the beginning of the next line to look for the missing data. This often causes problems; therefore, you might want to run a program to fill out each record with blanks so that SAS will read missing values as really missing.

   $ RUN [ACS.UTIL]PADBLANKS

The program will ask you to enter the input filename, output filename, and maximum columns specified in INPUT statement of your SAS program.

3. Creating a command file RUNSAS.COM

   $ EVE RUNSAS.COM

   $ SAS EXAMPLE [CTRL/Z]

   $

4. Submitting the command file RUNSAS.COM

   $ S30 RUNSAS ! CPU less than 30 minutes - BATCH30
   or
   $ SE RUNSAS ! CPU less than a minute - EXPRESS
   or
   $ S60/AFTER=23 RUNSAS ! Submit it after 11PM
   ! and CPU less than an hour - BATCH60

5. Checking the results and getting a hard copy

   Errors usually can be detected by checking the LOG files of the command file and the SAS program.

   $ TYPE RUNSAS.LOG ! Check the command procedure
   $ TYPE EXAMPLE.LOG ! Check the SAS program
   $ TYPE EXAMPLE.LIS ! Check the SAS output
   $ PRINT/NOTIFY EXAMPLE.LOG,EXAMPLE.LIS ! Let us get a hard copy