Homework 9-SOLUTIONS

Part I:

1. Using an Iterative DO Loop with a Conditional Clause

```
data expenses;
   Income= 50000000;
   Expenses = 38750000;
   do Year=1 to 30 until (Expenses > Income);
      income+(income * .01);
      expenses+(expenses * .02);
   end;
run;

proc print data=expenses;
   format income expenses dollar15.2;
run;
```

2. Using Arrays for Repetitive Computations

```
data special_offer;
   set orion.orders_midyear;
   array mon{*} month1-month3;
   keep Total_Sales Projected_Sales Difference;
   Total_Sales=sum(of month1-month6);
   do i=1 to 3;
      mon{i}=mon{i} *.90;
   end;
   Projected_Sales=sum(of month1-month6);
   Difference=Total_Sales-Projected_Sales;
run;

options nodate nonumber;
title 'Total Sales with 10% Discount in First Three Months';
proc print data=special_offer noobs;
   format total_sales projected_sales difference dollar10.2;
   sum difference;
run;
title;
```

3. Using a Character Array for Table Lookup

```
data passed failed;
   set orion.test_answers;
   drop i;
   array Response{10} Q1-Q10;
   array Answer{10} $ 1 _temporary_ ('A','C','C','B','E',
                                    'E','D','B','B','A');
   Score=0;
   do i=1 to 10;
      if Answer{i}=Response{i} then Score+1;
   end;
   if Score ge 7 then output passed;
   else output failed;
run;

title 'Passed';
proc print data=passed;
run;
title;

title 'Failed';
proc print data=failed;
run;
title;
```

Part II

1. Rotating a Data Set and Using a Lookup Table

```
data travel;
   set orion.travel_expense;
   keep employee_id trip_id Expense_Type amount;
   array exp{5} exp1-exp5;
   array descr{5} $ 14 _temporary_ ('Airfare', 'Hotel', 'Meals',
                                 'Transportation', 'Miscellaneous');
   do i=1 to 5;
      if exp{i} ne . then do;
         Expense_Type=descr{i};
         Amount=exp{i};
         output;
      end;
   end;
run;

proc print data=travel;
   format Amount dollar8.2;
run;
```

Part III

1. Using Other Loop Control Statements

```
data expenses;
   Income=50000000;
   Expenses =38750000;
   do Year=1 to 75;
      income +(income * .01);
      expenses+(expenses * .02);
      if expenses > income then leave;
   end;
run;

proc print data=expenses;
   format income expenses dollar14.2;
run;
```

2. Terminating a DATA Step

```
data fsp;
   set orion.orders_midyear;
   keep Customer_ID Months_Ordered Total_Order_Amount;
   array amt{*} month:;
   if dim(amt) < 3 then do;
      put 'Insufficient data for Frequent Shopper Program';
      stop;
   end;
   Total_Order_Amount=0;
   Months_Ordered=0;
   do i=1 to dim(amt);
      if amt{i} ne . then Months_Ordered+1;
      Total_Order_Amount+amt{i};
   end;
   if Total_Order_Amount>1000 and Months_Ordered >= (dim(amt))/2;
run;

title 'orion.orders_midyear: Frequent Shoppers ';
proc print data=fsp;
   format total_order_amount dollar10.2;
run;
title;

data fsp;
   set orion.orders_qtr1;
   keep Customer_ID Months_Ordered Total_Order_Amount;
   array amt{*} month:;
   if dim(amt) < 3 then do;
      put 'Insufficient data for Frequent Shopper Program';
      stop;
   end;
   Total_Order_Amount=0;
   Months_Ordered=0;
   do i=1 to dim(amt);
      if amt{i} ne . then Months_Ordered+1;
      Total_Order_Amount+amt{i};
   end;
   if Total_Order_Amount>1000 and Months_Ordered >= (dim(amt))/2;
run;
title 'orion.orders_qtr1: Frequent Shoppers ';
proc print data=fsp;
   format total_order_amount dollar10.2;
run;
title;

data fsp;
   set orion.orders_two_months;
   keep Customer_ID Months_Ordered Total_Order_Amount;
   array amt{*} month:;
   if dim(amt) < 3 then do;
      put 'Insufficient data for Frequent Shopper Program';
      stop;
   end;
   Total_Order_Amount=0;
   Months_Ordered=0;
   do i=1 to dim(amt);
      if amt{i} ne . then Months_Ordered+1;
      Total_Order_Amount+amt{i};
   end;
   if Total_Order_Amount>1000 and Months_Ordered >= (dim(amt))/2;
run;
title 'orion.orders_two_months: Frequent Shoppers ';

proc print data=fsp;
   format total_order_amount dollar10.2;
run;
title;
```

3. Rotating a Data Set

```
data customer_orders;
   set orion.order_summary;
   retain Month1-Month12;
   array Month{12};
   by Customer_ID;
   if first.Customer_ID then call missing(of Month{*});
   Month{Order_month}=Sale_Amt;
   if last.Customer_ID;
   drop Order_Month Sale_Amt;
run;

options ls=120;
proc print data=customer_orders noobs;
run;
```