CENTERS FOR MEDICARE & MEDICAID SERVICES

CY 2022 PART D

BASELINE OUT-OF-POCKET COST MODEL

USER GUIDE

JANUARY 2022

Table of Contents

Introduction

The Out-of-Pocket Cost (OOPC) Model is a set of programs used to calculate the estimated OOPCs using a beneficiary sample to determine the value of the benefits being offered by a Plan Benefit Package (PBP). The purpose of this User Guide is to provide Medicare Advantage Organizations (MAOs) and Prescription Drug Plan (PDP) Sponsors with the technical information required to generate OOPC values in order to evaluate the changes to the OOPC model. Please note, MAOs will need to calculate their Part C & Part D OOPCs separately and combine them for their total OOPC value.

This version of the Part D OOPC model, referred to as the **CY 2022 Part D Baseline OOPC Model,** uses a 0.1% sample of Part D beneficiary drug utilization rather than the MCBS sample cohort that had been used in older versions of the model. The OOPC Model is designed to enable plan organizations to review benefit structures using the software and data that will be used by CMS to evaluate annual bid submissions. The OOPC Model reports OOPC values at the plan level. The section, **Development of the Part D OOPC Data**, summarizes CMS's process to produce the OOPC values. Plan Sponsors are encouraged to review the more comprehensive Methodology document located in the most recently posted Plan version OOPC Model package at

https://www.cms.gov/Medicare/Prescription-Drug-Coverage/PrescriptionDrugCovGenIn/OOPCResources.html

Organizations use their own completed 2022 PBP data as input to the software (a given organization may have multiple plans for a given contract). After the user has successfully input their data for a particular contract/plan into the PBP, and exit/validated the PBP, then the data is ready for use in the model. Users download the OOPC Model and follow the directions for where to copy the SAS programs and SAS data that serve as the other inputs. The user edits several small SAS programs and then executes them.

The CY 2022 Part D Baseline OOPC Model package (**CY 2022 Part D Baseline OOPC Model.zip**) contains input datasets (SAS transport format) and a series of SAS programs. The programs import PBP, formulary, and utilization data; calculate person-plan-level costs for Part D benefits, summarize the costs to the plan level, and output the results at a plan-level to an Excel workbook.

Questions can be directed as follows:

> For technical questions about the OOPC model, please submit an email to OOPC@cms.hhs.gov
>
> For Part D policy related questions about meaningful difference, please submit an email to partdbenefits@cms.hhs.gov
>
> For policy related questions about Total Beneficiary Costs (TBC), please submit an email to https://mabenefitsmailbox.lmi.org/MABenefitsMailbox/.

For Bid Pricing Tool (BPT) questions, please submit questions to actuarial-bids@cms.hhs.gov

Resource Requirements

Operation of the Model requires that the user be familiar with PC file management and operating SAS software.

Model Requirements: The Models have been tested on a variety of PCs. The user will need WINZIP (or similar software) to unzip the OOPC model package and storage space to accommodate the downloadable files that total over 5 MB zipped. A version of PC SAS with SAS/ACCESS Interface to PC Files installed will be required. The Model was developed and tested using SAS Version 9.4 on 64-bit machines using Windows Office 16. Microsoft Excel is required for generating and using the model output. Testing has been done using 2016 and 2019 versions of Excel and Access.

Processing Time: The programs that import the various input files will run quickly. The other programs for the Part D calculations incorporate many different variables and combinations of covered/non-covered drugs, pricing structures, and formularies. Running a single or a few plans at a time will shorten the run time, especially when fewer drug formularies are involved.

Input Datasets Included in the Software Package

Drug Utilization Data Provided by CMS

The software includes four SAS transport datasets. The person-level (PERSON.XPT) file contains information on the cohort of beneficiaries drawn from the CMS Common Medicare Environment (CME) files. The BENE_SCRIPT.XPT file lists the drug names and scripts for each sample beneficiary extracted from the PDE file. The RXCUI_REFERENCE.XPT file lists all of the RxCUIs used in the OOPC model, along with their associated Brand/Generic status, and average prices. The FRF_XWALK.XPT lists all the RxCUIs and their potential generic substitution RxCUIs. The datasets are used after they are converted to SAS datasets with a SAS program included in the package (CIMPORT.SAS).

Input Datasets Provided by the User

Plan List

The user will provide a text file list of the plans to be included in an OOPC run. This file (**PLANFILE.TXT**) will consist of a combined Contract/Plan/Segment identifier. Part D plans do not have segment identifier, but one is required to be added as “000” for this file. For example, PDP plan S9999 001 will appear as S9999001000.

Planfile.txt Record Layout

Required File Format = ASCII File - Tab Delimited
Do not include a header record
Filename extension should be ".TXT"

Field Name	Field Type	Field Length	Field Description	Sample Field Value(s)
Contract_Plan_Segment	CHAR	11	Unique Contract/Plan/Segment identifier	S9999001000

Parts of an example file look like:

S9999001000
S9998001000
S9997002000

Note: Only plans in the plan list will be run in the OOPC calculation, even if more plans exist in a user's PBP database.

PBP Data

Each year, plan personnel and other users are required to enter their benefit data into the Plan Benefit Package (PBP) software in order to submit a bid. Plans are provided with instructions each year on how to enter data into the PBP software. See the PBP System Reference Guide for PBP instructions.

PBP Data Input to OOPC Tool: As part of this bid submission process, the PBP data is automatically stored in a database. The OOPC Model reads a plan's PBP data from the Access database as input. The PBP-created databases required as input to the Model are **PBP2022.MDB** and **PBPPLANS2022.MDB**. The location of these files is a user input in the OOPC software.

Note: The OOPC Model should point to the databases associated with the PBP Super User. If there are other PBP data entry users, the Super User should ensure that he or she has received the most up-to-date data entry before running the OOPC Model.

Drug Formulary Data

For producing the Part D OOPC values, plan organizations with Part D benefits (PDPs and MA-PDs) will produce five files that describe the plan's formulary.

The first file, **FORMULARY.TXT**, needs to contain a tab-delimited list of the drugs for each formulary of the plans to be included in an OOPC calculation. This file and other .txt files described below should not contain header, or label rows, and should keep any leading zeroes. Each row in the file will contain, in this order: a formulary identifier, an RxCUI, and a tier-level identifier (1-7). This information can be obtained from the plan organization’s formulary.

Formulary.txt Record Layout

Required File Format = ASCII File - Tab Delimited
Do not include a header record
Filename extension should be “.TXT”

Field Name	Field Type	Field Length	Field Description	Sample Field Value(s)
Formulary ID	CHAR	8	Unique Formulary Identifier	00022990
RxCUI	Number	Maximum of 8 digits	Rx Norm concept unique identifier from the active CY2022 Formulary Reference File	721775
Tier Level	Number	1	Defines the Cost Share Tier level associated with the drug	1 = Tier Level 1 2 = Tier Level 2 3 = Tier Level 3 4 = Tier Level 4 5 = Tier Level 5 6 = Tier Level 6 7 = Tier Level 7

Parts of an example file look like:

```
00022990   721775        1
00022991   721793        1
00022992   721795        2
00022993   721797        3
00022994   722113        2
```

Note: The Formulary ID needs to have an 8-digit field length. Any entries greater or less than 8 digits will not be read or used by the model.

The second file, **GAP_DRUGS.TXT** contains a tab-delimited list of all plans and drugs (RxCUIs) for each plan that has partial tier coverage. This information can be obtained from the plan organizations’ supplemental formulary file submissions. The file will be submitted with a contract identifier and a plan identifier (no segment identifier required).

The third file, **FFF.TXT**, contains a tab-delimited list of all plans and drugs (RxCUIs) for each plan that offers Free First Fill coverage. This information can be obtained from the plan organizations’ supplemental formulary file submissions. The file will be submitted with a Contract identifier and a Plan identifier (no segment identifier required).

Note: If a plan has no partial tier coverage or no Free First Fill drugs, a blank version (i.e., no rows) of the **GAP_DRUGS.TXT** or **the FFF.TXT** file needs to be created and saved.

Gap Drugs.txt/FFF.txt Record Layout

Required File Format = ASCII File - Tab Delimited
Do not include a header record
Filename extension should be ".TXT"

Field Name	Field Type	Field Length	Field Description	Sample Field Value(s)
Contract ID	CHAR	5	Contract Number	S9999
Plan ID	CHAR	3	Plan Identifier	001
RxCUI	Number	Maximum of 8 digits	Rx Norm concept unique identifier from the active CY2022 Formulary Reference File	721775

Parts of an example file look like:

```
S9999        001      721797
S9999        001      722113
```

The fourth file, **PLAN_FORMULARY.TXT**, contains a tab-delimited list of all contract, plan, and formulary identifiers that are to be run. The list of plans needs to correspond exactly with the list of plans in the **PLANFILE.TXT** file described above, although only the contract plan and plan identifiers are required.

Plan Formulary.txt Record Layout

Required File Format = ASCII File - Tab Delimited
Do not include a header record
Filename extension should be ".TXT"

Field Name	Field Type	Field Length	Field Description	Sample Field Value(s)
Contract ID	CHAR	5	Contract Number	S9999
Plan ID	CHAR	3	Plan Identifier	001
Formulary ID	CHAR	8	Unique Formulary	00022990

Parts of an example file look like:

```
S9999        001      00022994
S9998        001      00022990
S9997        002      00022991
```

The fifth file, **SENIOR_SAVINGS.TXT** contains a tab-delimited list of all plans drugs (RxCUIs), and the cohort for each plan that participates in the Senior Savings Demonstration. This information can be obtained from the plan organizations' supplemental formulary file submissions. The file will be submitted with a Contract identifier, a Plan identifier, and a Segment identifier.

Note: If a plan has no partial tier coverage, no Free First Fill drugs, or does not participate in the Senior Savings Demo, a blank version (i.e., no rows) of the **GAP_DRUGS.TXT, the FFF.TXT**, or the **SENIOR_SAVINGS.TXT** file needs to be created and saved.

Senior_Savings.txt Record Layout

Required File Format = ASCII File - Tab Delimited
Do not include a header record
Filename extension should be ".TXT"

Field Name	Field Type	Field Length	Field Description	Sample Field Value(s)
Contract ID	CHAR	5	Contract Number	S9999
Plan ID	CHAR	3	Plan Identifier	001
Segment ID	CHAR	3	Segment Identifier	000
RxCUI	Number	Maximum of 8 digits	Rx Norm Concept Unique Identifier from the active CY2022 Formulary Reference File	721775
Cohort	CHAR	1	Defines the Cohort number (1,2,3,4 or 5) associated with the drug	3

Excerpt of an example file:

```
S9999   001     000     721797  2
S9999   001     000     722113  5
```

Programs Included in the Software Package

The complete list of SAS Programs utilized by the model can be found in the Contents of the Zip File section below. The key programs that launch the computations are described below:

CIMPORT.SAS converts the SAS transport files supplied with this software into SAS datasets.

PARTD_FORM.SAS takes the Part D related formulary files described above and converts them into SAS format.

OOPCV1P.SAS supplies user-defined parameters needed to run the OOPC Model and calls the other SAS programs that carry out the calculations.

Instructions for Running the Model and Creating OOPC Values

Please read and follow the instructions carefully before running the software. Note that this assumes the PBP has already been completed for the plan(s) of interest.

Step 1: Create a text file (**PLANFILE.TXT**) that lists the plans to be used in the OOPC run.

Step 2: Complete the PBP data entry for plans of interest using the PBP software. The resulting files will be named **PBP2022.MDB** and **PBPPLANS2022.MDB**. Make a note of the location of these files: e.g., c:\program files\PBP2021.

Step 3: Create text files for the formulary information of the plans to be run: **FORMULARY.TXT**, **PLAN_FORMULARY.TXT**, **GAP_DRUGS.TXT**, **FFF.TXT, and SENIOR_SAVINGS.TXT**, and copy them to a created formulary directory: e.g., c:\oopc_d\formulary. Make a note of the location of these files.

Step 4: Set up directory locations for all files.

a. Copy the **CY 2022 Part D Baseline OOPC Model.zip** to a working directory (e.g., **c:\oopc_d**) and unzip its contents to that directory
b. In the working directory, extract the contents of **programs.zip** to create the **c:\oopc\programs** directory for the SAS programs modified by the user.
c. In the working directory, extract the contents of **input.zip** to create the **c:\oopc\input** directory for the input files and the programs that are not changed by the user.
d. Set up a directory for the output spreadsheet file (e.g., **c:\ oopc_d\output**).
e. Copy the **PLANFILE.TXT** file to the newly created programs file directory. (e.g., **c:\oopc_d\programs**).

Step 5: Edit the program **CIMPORT.SAS** as necessary so that the location **(in bold below)** of the input data is specified for all of the .XPT files. The programs provided in the model package contain, as defaults, the directory locations listed above. The user can change these locations, as desired.

```
* PROGRAM: CIMPORT.SAS;
* DESCRIPTION: IMPORT THE INPUT FILES TO THE OOPC PROCESS;

%LET DATALOC = %str(c:\oopc_d\input);
```

Then run **CIMPORT.SAS**.

For all SAS runs, check the SAS Log to make sure the text string **ERROR** does not appear anywhere. (In the **Troubleshooting** section below are noted several sources of problems when setting up and running the programs).

***Note*:** The SAS CIMPORT program only needs to be run once, prior to the first OOPC run.

Step 6: Import **FORMULARY.TXT**, **PLAN_FORMULARY.TXT, GAP_DRUGS.TXT, FFF.TXT,** and **SENIOR_SAVINGS.TXT**, files by editing the provided **PARTD_FORM.SAS** program, as necessary, for the correct directory locations and files.

```
*PROGRAM: PARTD_FORM.SAS;
*DESCRIPTION: CREATES SAS FILES FOR FIVE TAB DELIMITED FILES;
%LET DIR = C:\OOPC_D\formulary;
%LET FORMFILE = FORMULARY.TXT;
%LET PLANFORM = PLAN_FORMULARY.TXT;
%LET GAPDRUGS = GAP_DRUGS.TXT;
%LET FFF = FFF.TXT;
%LET SS = SENIOR_SAVINGS.TXT;
```

Then run **PARTD_FORM.SAS**.

Note: Once this step is finished, and if there is no change in the formulary data, the user does not need to redo this step for subsequent runs. As necessary, the user may create and use different formulary text files and rerun **PARTD_FORM.SAS**.

Step 7: Edit the program **OOPCV1P.SAS** as shown below to indicate the directories (**in bold**) where the SAS programs and input files are stored. The programs provided in the model package contain, as defaults, the directory locations listed above. Also, edit the program to indicate where the PBP data are stored. And finally, edit the program to identify the location and name of the output spreadsheet file. The output spreadsheet name can be changed as necessary.

For example, in the "**OOPC** =&OUTPUT.**OOPC_RUN**&file_date." line, to identify the first run for a given day, change the default label "**OOPC_RUN**" to "**OOPC_RUN1_.**" (The **"&file_date**" function automatically outputs the date of the run.)

```
* PROGRAM: OOPCV1P.SAS;
* DESCRIPTION: MAIN OOPC PROGRAM;

%LET INPUTDIR =         c:\oopc_d\input;
%LET PROGDIR  =         c:\oopc_d\programs;
%LET PBPDIR =           c:\ pbp2022;
%LET FORMDIR =          c:\oopc_d\formulary;
%LET PLANFILEDIR =      c:\oopc_d\programs;
%LET OUTPUT =           c:\oopc_d\output;

%OOPCV1M(RUNYEAR              =2022,
      INP                     =IN1.PERSON,
PBP                          =&PBPDIR,
FORMULARY                    =&FORMDIR,
```

Then run **OOPCV1P.SAS**.

When checking the SAS Log for the run, the run time is determined by looking at the last few lines of a successful run. For example,

```
NOTE: The SAS System used:
   real time         1:36.67
   cpu time          43.10 seconds
```

The resulting Excel spreadsheet file (.xlsx) will be written to the designated output file directory when the program finishes running successfully. The field **PartD** displays the Part D OOPC.

An example (truncated) of the resulting spreadsheet output is shown below (test data):

CONTRACT_NUMBER	PLAN_ID	SEGMENT_ID	PLAN_NAME	BENEFIT_YEAR	PARTD	PBP_VERSION_DATE
H0204	001	000	RFB MA-PD A/B Full Network EA (PFFS)	2022	129.008346	12FEB2021:09:02:14
H0809	001	000	RFB MA-PD A/B DS (HMO-POS)	2022	178.0785887	12FEB2021:09:08:10
H1360	001	000	RFB MA-PD A/B Partial Network AE (PFFS)	2022	279.331	12FEB2021:09:10:51
H1668	001	000	MA-PD A/B SNP Chroncic/Disabling EA (HMO-POS SNP)	2022	129.008346	12FEB2021:09:12:20
H1936	001	000	MA-PD A/B Full Network BA (PFFS)	2022	129.008346	12FEB2021:09:33:42
H4850	001	000	RFB MA-PD A/B DS (PPO)	2022	178.0785887	12FEB2021:09:36:11
H5596	015	000	MA-PD A/B EA (PPO)	2022	279.331	12FEB2021:09:42:49
H6332	002	000	RFB MA-PD A/B DS (HMO)	2022	129.008346	12FEB2021:10:37:34
R118	001	000	MA-PD A/B BA (Regional PPO)	2022	156.4093657	12FEB2021:10:02:44
S2447	008	000	PD-Only EA (PDP)	2022	178.0785887	12FEB2021:10:41:12

Note: The PBP_Version_Date is displayed for reference purposes.

Rerunning the Model

Change Plan Benefits for a Plan: To change the plan benefit assumptions for the same plan(s), first modify the appropriate PBP data entry.

Change Plans: To change plans, you must modify the PBP data entry, change the PLANFILE.TXT and if necessary, the formulary .txt files.

Change Formulary files/Same plan: To change formulary assumptions for the same plan(s), change the formulary.txt files.

For any of the above changes, after changing input files, and rerunning **PARTD_FORM.SAS**, as necessary, rerun **OOPCV1P.SAS**, while changing the Excel output file name.

Contents of the Output (Excel) File

The output from the OOPC Model is a single Excel file. The table below lists the labels as they appear in the output file and in the corresponding detailed heading.

Note: Labels used in the output file are restricted to no more than 32 characters by SAS.

Label Used in Output Files	Detailed Heading/Description
Contract_Number	Contract Number
Plan_ID	Plan ID
Segment_ID	Segment ID
Plan_Name	Plan Name
Benefit_Year	Benefit Year/PBP for Estimated OOPC Values
PartD	Part D OOPC Value
PBP_Version_Date	PBP Version Date

Contents of the ZIP File (**CY 2022 Part D Baseline OOPC Model.zip**)

1. **Input.zip**
   AE_CALCS.SAS
   BA_CALCS.SAS
   BASEID_DRUGS.SAS
   BENE_SCRIPT.XPT
   BENEFIT_OOPC_COST_CALCULATION.SAS
   BUILD_FFF_FORMAT.SAS
   BUILD_GAPDRUGS_FORMAT.SAS
   BUILD_LOOKUP_CMS_SUB.SAS
   BUILD_LOOKUP_SUB.SAS
   BUILD_LOOKUP_FORMALT.SAS
   BUILD_SS_FORMAT.SAS
   CLEANUP.SAS
   CONVERT.SAS
   DS_CALCS.SAS
   EA_CALCS.SAS
   EXCEPT_TA.SAS
   FFS_BENEFIT_OOPC_COST_CALCULATION.SAS
   FRF_XWALK.XPT
   OOPCV1M.SAS
   PBP_DRUG_VARIABLES.SAS

PBP_DRUG_VARIABLES_SSM.SAS
PBP_IMPORT.SAS
PBP_IMPORT_CMS.SAS
PBP_IMPORT_PARTD.SAS
PBP_IMPORT_PARTD_CMS.SAS
PERSON.XPT
PLAN_CATNAME_NEW.SAS
PLAN_LEVEL.SAS
RXCUI_REFERENCE.XPT

2. **Programs.zip**
CIMPORT.SAS
OOPCV1P.SAS
PARTD_FORM.SAS

Development of the Part D OOPC Data

The CY 2022 Part D Baseline OOPC Model was developed using the methodology summarized below.

Average monthly Part D OOPCs are calculated for each health plan. CMS applies the data entered into the Plan Benefit Packages (PBPs) to compute the OOPCs based on formulary and co-payments/coinsurance for each drug.

CMS made the following basic assumptions related to the OOPC estimates for prescription drugs:

- Prescription Drug Event (PDE) National Drug Codes (NDCs) are mapped into RxCUI codes to apply a particular plan's tier-formulary based cost sharing. The PDE drug events used by the CY 2022 Part D Baseline OOPC model is based on a random 0.1% sample of 2020 Part D enrolled Medicare beneficiaries.

- Prescription Drug Event (PDE) claims data (2020) were used to calculate average drug prices.

Part D OOPC

The estimated OOPC values are based upon the drug information found in the PDE file provided for the individual sample beneficiaries. The beneficiaries used to identify the drug utilization come from a random 0.1% sample of Part D enrolled Medicare beneficiaries in 2020. The data are used in conjunction with the Calendar Year (CY) 2022 Plan Benefit Packages submitted by plans that detail the drug benefit cost sharing and plan coverage as well as the CY 2022 plan-level formulary submissions. The NDC on each PDE record is mapped into an RxCUI using the appropriate CMS formulary reference file (FRF).

An average price for each RxCUI is calculated using the 2020 PDE claims data. The average price is calculated as the total gross expenditure (ingredient cost + dispensing fee + taxes + vaccination fee) divided by the number of 30-day equivalent prescriptions.

Using each plan's drug coverage status and PBP-based cost sharing information (deductible, initial coverage limit, co-copayments and/or coinsurance, gap coverage, etc.), the beneficiary's OOPCs are calculated. The calculations are performed according to the type of Part D plan (Defined Standard, Basic Alternative, Actuarially Equivalent, or Enhanced Alternative) and the associated cost share structure. The calculations are based on the assumption that each prescription is for a one-month (30-day) supply of drugs (rather than a 60- or 90-day supply) from an In-Network Pharmacy. In the event that both a preferred and a standard pharmacy exist, the calculations are based on the preferred pharmacy cost-sharing.

Generic substitution is assumed such that when a generic version of a brand drug exists and is covered on the plan's formulary, the generic version is the one included in the calculations, provided it has lower cost sharing. Food and Drug Administration (FDA) application type is utilized to determine the applicable/non-applicable status of drugs. The average price is calculated as total gross drug expenditure (ingredient cost, dispensing fee, taxes, and vaccination fee) divided by the number of PDE.

In the instances when both a brand drug and its AB rated generic are not covered on a plan's formulary, the OOPC model assigns the full cost, as reflected by the average price, for a non-formulary drug to a plan's OOPC value.

This data creation process results in a file that includes the total cost of the drug for each sample beneficiary and prescription, as well as each plan's associated cost sharing structure for that drug. The beneficiary level OOPC values are then averaged at the plan level (across all beneficiaries in the data sample) in order to yield nationally representative data.

Troubleshooting

Below are several areas where users may have problems running the model:

Wrong or Missing Directory Locations

If an "input" directory is empty, the following type of error can show up in the SAS log while attempting to run the **CIMPORT.SAS** or **Part D_FORM.SAS** programs.

```
NOTE: Library IN does not exist.
ERROR: Library IN does not exist.
NOTE: Library OUTPUT does not exist.
ERROR: Physical file does not exist, c:\oopc_d\input\person.xpt
```

If an incorrect directory name for input data is listed in the OOPCV1P.SAS program, the following type of error may be displayed in the SAS log.

```
%LET DIR  = c:\oopc_d\formulary (correct)
%LET DIR  = c:\oopc_d\form (incorrect)
```

```
ERROR: Library FORMULARY does not exist.
ERROR: Unable to open catalog FORMULARY.FORMATS.
```

Problems with Output Files

Each new SAS run should have a new unique output file name designated in the **OOPCV1P.SAS** program. If you do not change the name from a previously created Excel file, the new SAS run will overwrite the old file contents, or if the current Excel file is open, will not produce output at all. An example error message is shown below:

```
ERROR:  The MS  Excel  table  OOPCS_2022  has  been opened for
OUTPUT. This table already exists, or there is a name conflict with an
existing object. This table will not be replaced. This engine does not support
the REPLACE option.
ERROR: Export unsuccessful. See SAS Log for details.
```

Another message will be generated if you forget to create an output directory. For example,

```
ERROR: Connect: 'c:\oopc_d\output\OOPC_RUN20210415.xlsx' is not a valid
path. Make sure that the path name is spelled correctly and that you are
connected to the server on which the file resides.
ERROR: Error in the LIBNAME statement.
```

Also, you may submit a run, find no “Error” messages in the **OOPCV1P.SAS** program, and yet find no Excel output file. One way this can happen is if the plan identifiers in the PLANLIST.TXT file are filled out without the final 3 segment identifiers, e.g.:

```
S9999001
```

Problems with Insufficient Hard Drive Space

If you have been running the model repeatedly, you may encounter the following error message:

```
WARNING: File 'WORK.xxxxxx.DATA' is shorter than expected.
ERROR: The file WORK.xxxxxx.DATA is shorter than expected.
ERROR: The file WORK.xxxxxx.DATA is shorter than expected.
ERROR: The file WORK.xxxxxx.DATA is shorter than expected.
WARNING: Data set WORK.yyyyyy was not replaced because this step was
stopped.
ERROR: The open failed because library member WORK.xxxxxx.DATA
is damaged.
ERROR: The open failed because library member WORK.xxxxxx.DATA
is damaged.
```

ERROR: The open failed because library member WORK.xxxxxx.DATA is damaged.

This problem means that SAS does not have sufficient hard disk space for its temporary files. You can reboot your machine so that more memory is available to SAS. Also, check that you do not have 'leftover' SAS temporary directories. An example of SAS temporary directories that may remain from other sessions under 'My Computer' is:

> c:\Documents and Settings\yourname\Local Settings\Temp\SAS Temporary Files\
> with subdirectories such as:
> TD_xxxxx
> SAS_util000100000150_machinename

Part D Output Expected, but Blank

When you have completed your PBP data entry, make sure you have exit/validated from the program. Also, output may not be produced if the formulary IDs are not formatted correctly (i.e., tab-delimited) or if they are formatted differently in the two input files:

FORMULARY.TXT and **PLAN_FORMULARY.TXT**.

Running the Model Using an Older Version of SAS 9.3

The OOPC Model has been developed and tested using SAS 9.4. Prior versions of SAS may not support all the functions and procedure used in the software.