Fiscal Impacts Model Documentation

Version 1.0

**S. L. Beck
M. J. Scott**

May 1986

**Prepared for the U.S. Department of Energy
under Contract DE-AC06-76RLO 1830**

**Pacific Northwest Laboratory
Operated for the U.S. Department of Energy
by Battelle Memorial Institute**

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

PACIFIC NORTHWEST LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC06-76RLO 1830

Printed in the United States of America
Available from
National Technical Information Service
United States Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22161

NTIS Price Codes
Microfiche A01

Printed Copy

Pages	Price Codes
001-025	A02
026-050	A03
051-075	A04
076-100	A05
101-125	A06
126-150	A07
151-175	A08
176-200	A09
201-225	A010
226-250	A011
251-275	A012
276-300	A013

FISCAL IMPACTS MODEL
DOCUMENTATION

VERSION 1.0

S. L. Beck
M. J. Scott

May 1986

Prepared for
the U.S. Department of Energy
under Contract DE-AC06-76RLO 1830

Pacific Northwest Laboratory
Richland, Washington 99352

SUMMARY

The Fiscal Impacts (FI) Model, Version 1.0 was developed under Pacific Northwest Laboratory's Monitored Retrievable Storage (MRS) Program to aid in development of the MRS Reference Site Environmental Document (PNL 5476). It computes estimates of 182 fiscal items for state and local government jurisdictions, using input data from the U.S. Census Bureau's 1981 Survey of Governments and local population forecasts. The model can be adapted for any county or group of counties in the United States.

CONTENTS

1.0	PURPOSE AND OVERVIEW	1
2.0	DATA SOURCES	2
3.0	PROCEDURES USED FOR MODEL ESTIMATION AND DERIVATION OF PARAMETERS	3
4.0	PROCEDURES USED AND RESULTS OF EVALUATION TESTS FOR MODEL ACCURACY, SENSITIVITY AND VALIDATION	4
5.0	PROGRAMS USED	5
5.1	FINANCE PROGRAMS	5
5.1.1	PROGRAM -- STATE.FORT	5
5.1.2	PROGRAM -- FINSTATE.FORT	7
5.1.3	PROGRAM -- SURV81.FORT	16
5.1.4	PROGRAM -- FINANCE.CITY.FORT	17
5.1.5	PROGRAM -- FINANCE.COUNTY.FORT	18
5.1.6	PROGRAM -- FINANCE.SCHOOL.FORT	19
5.1.7	PROGRAM -- FINANCE.COMBINE.FORT	20
5.1.8	PROGRAM -- FINANCE.COMBINE.SCHOOL.FORT	21
5.2	PROGRAM -- FISCAL.FORT	22
5.3	MERGING AND EDITING SCENARIO FILES	24
5.4	PROGRAM -- REREPS.FORT	25
5.5	PROGRAM -- RECAT.SUMMARY.FORT	26

APPENDIX A: PROGRAM LISTINGS

FINANCE.CITY.FORT	A.1
FINANCE.COMBINE.FORT	A.5
FINANCE.COMBINE.SCHOOL.FORT	A.6
FINANCE.COUNTY.FORT	A.7
FINANCE.SCHOOL.FORT	A.11
FINSTATE.FORT	A.15
FISCAL.FORT	A.18

FISCAL.FORT	.	.	.	.	.	.	.	.	.	A.18
RECAT.SUMMARY.FORT	.	.	.	.	.	.	.	.	.	A.19
REREPS.FORT	.	.	.	.	.	.	.	.	.	A.29
STATE.FORT	.	.	.	.	.	.	.	.	.	A.30

1.0 PURPOSE AND OVERVIEW

The Fiscal Impacts (FI) Model, Version 1.0 was developed under Pacific Northwest Laboratory's Monitored Retrievable Storage (MRS) Program for development of MRS Reference Site Environmental Document (PNL 5476). The model is extremely simple in philosophy. It accepts a historical population estimate by age cohort for each city, county, state or special district government jurisdiction for which it forecasts, and estimates per capita fiscal items prevailing during the 1981 Survey of Governments performed by the U.S. Census Bureau. These per capita items are then used with a user-supplied forecast of population for these same local government jurisdictions to forecast future revenue, expenditure, and debt items as if the 1981 per capita rates continued to prevail. The model has the capability of estimating the discounted value of each fiscal item in 1985 dollars. It can be use to compare the difference in fiscal items between two future economic development scenarios for the purpose of estimating fiscal impacts.

2.0 DATA SOURCES

Data were gathered from the Survey of Governments, 1981: Finance Statistics tape. This tape contains data which includes a survey of all state governments, all school district governments in 30 states and sample coverage in 20 states, and a sample of approximately 14,000 other local government units included in the Census of Governments, 1977 Finance Summary Statistics. The tape name is GOVT81.FIN.SUR.DATA and is located on the Energy Information Administration (EIA) computer, which is located at the Forrestal Building, Washington D.C.. All data files are on account JSI6948. Documentation on the data tape can be found in the Survey of Governments, 1981: Finance Statistics, Tape Technical Documentation. It is assumed that the user is familiar with the EIA computer, e.g., logging onto the system, accessing files and using TSO and SUPERWYLBUR.

3.0 PROCEDURES USED FOR MODEL ESTIMATION AND DERIVATION OF PARAMETERS

No parameters are derived or estimated.

4.0 PROCEDURES USED AND RESULTS OF EVALUATION TESTS FOR MODEL ACCURACY, SENSITIVITY, AND VALIDATION

The purpose of this model is to forecast fiscal items for state and local government using the revenue, expenditure, and debt rates measured by the 1981 Survey of Governments. Therefore it is not reasonable to attempt to validate the model by forecasting some other years' fiscal items for these governments, since it is obvious that 1981 rates for fiscal items would change. Hand calculations were performed on the individual fiscal items and totals to confirm that the code was reproducing 1981 values. Since the model is linear, either an increase in population or in per capita fiscal items will proportionately increase the corresponding fiscal total.

5.0 PROGRAMS USED

This section contains the names of the programs used, a description of each program, the input and output files of each program and a glossary of variable names used within each program. All programs are located on account JSI6948. Documentation of Survey of Governments Finance data can be found in U.S. Department of Commerce, Bureau of the Census, Survey of Governments, 1981: Finance Statistics.

5.1 FINANCE PROGRAMS

The following programs are used to read the Survey of Government Finance data, aggregate the data into 182 finance categories and calculate the per capita figures:

```
STATE.FORT
FINSTATE.FORT
SURV81.FORT
FINANCE.CITY.FORT
FINANCE.COUNTY.FORT
FINANCE.SCHOOL.FORT
FINANCE.COMBINE.FORT
FINANCE.COMBINE.SCHOOL.FORT
```

5.1.1 PROGRAM -- STATE.FORT

This program reads data from the Survey of Governments, 1981: Finance Statistics tape and writes all state data to a sequential disk file.

INPUT: CN6948.JSI.GOV.T.FIN.SUR.DATA (tape)

OUTPUT: CN6948.JSI.SUR81.STATE.FINANCE.DATA

GLOSSARY OF VARIABLE NAMES:

```
ICODE --Type of Government Code
0 - state
1 - county
2 - municipality
3 - township
4 - special district
5 - school district
```

II --Government State Code

AL - 1	MT - 27
AK - 2	NE - 28
AZ - 3	NV - 29
AR - 4	NH - 30
CA - 5	NJ - 31
CO - 6	NM - 32
CT - 7	NY - 33
DE - 8	NC - 34
DC - 9	ND - 35
FL - 10	OH - 36
GA - 11	OK - 37
HI - 12	OR - 38
ID - 13	PA - 39
IL - 14	RI - 40
IN - 15	SC - 41
IA - 16	SD - 42
KS - 17	TN - 43
KY - 18	TX - 44
LA - 19	UT - 45
ME - 20	VT - 46
MD - 21	VA - 47
MA - 22	WA - 48
MI - 23	WV - 49
MN - 24	WI - 50
MS - 25	WY - 51
MO - 26	

JJ --Type of Government (same as ICODE above)

KK --Government County Code

LL --Unit Identification Code

XMM --Depending on type of Government:
 If JJ = 0,1,2, or 3 then XMM = Population
 If JJ = 5 then XMM = School Enrollment
 If JJ = 4 then XMM = Special School District
 Function

NN --Standard Metropolitan Statistical Area (SMSA) Code

NUMS1 --283 fields representing revenue, expenditure, indebtedness,
 and cash and security holdings

XNAME2--Name of state, county, municipality, township, school district
 or special district

5.1.2 PROGRAM -- FINSTATE.FORT

This program reads the output file created by the program, STATE.FORT, aggregates the state data on government revenues, expenditures, and debt (financial) items into 182 summary fiscal items and calculates the per capita figures for the states specified in lines 180 and 181. These lines need to be modified to select the states the user is interested in (see section 4.1.1 for a list of Government State Codes). The user may select more than one state and write the data for each of these states to one output file. The data for one particular state can then be copied to a separate output file using the naming convention as in Output #2 below. Population estimates must be supplied by the user.

INPUT: SUR81.STATE.FINANCE.DATA

OUTPUT: 1) STATES.FINANCE.DATA (use this naming convention if file contains more than 1 state)

2) XXXXXX.STATE.FINANCE.DATA (use this naming convention if file contains only 1 state;
XXXXXX = state)

GLOSSARY OF VARIABLE NAMES:

II --Government State Code (see section 4.1.1)

JJ --Type of Government Code
0 - state
1 - county
2 - municipality
3 - township
4 - special district
5 - school district

KK --Government County Code

LL --Unit Identification Code

POP --If value is:
0,1,2 or 3 - Population for state, county, municipality, or township
4 - Special district function
5 - School enrollment

NN --MSA code

ARY1 --An array of 283 elements representing Revenue, Expenditure, Indebtedness, and cash and security holdings

XNAME --State, county, municipality, township, school district or special district name

ARY2 --Revenue, expenditure, indebtedness, and cash and security holdings aggregated into 182 summary fiscal items. These items are:

- 1) Property Tax Revenue
- 2) General Sales Tax Revenue
- 3) Sum of:
 - Alcoholic Beverages
 - Motor Fuels Tax
 - Public Utilities Tax
 - Tobacco Products Tax
 - Other Selective Sales and Gross Receipts Tax
- 4) Sum of:
 - Motor Vehicle Licenses
 - Other License Tax
- 5) Sum of:
 - Income Taxes, NEC
 - Taxes, NEC
- 6) From Federal Government - Education
- 7) From Federal Government - Employment Security
- 8) From Federal Government - Health & Hospital
- 9) From Federal Government - Highways
- 10) Sum of:
 - From Federal Government - Housing & Urban Renewal
 - From Federal Government - Public Welfare
 - From Federal Government - All Other
- 11) Sum of:
 - From State Government - Local Schools, NEC
 - From State Government - Education, NEC
- 12) From State Government - General Local Government Support
- 13) From State Government - Health and Hospital
- 14) From State Government - Highways
- 15) Sum of:
 - From State Government - Housing & Urban Renewal
 - From State Government - Public Welfare
 - From State Government - All Other
- 16) Interschool System Revenue

- 17) Sum of:
 - From Local Government - Local Schools, NEC
 - From Local Government - Education, NEC
 - From Local Government - General Local Governmen
- Support
 - From Local Government - Health and Hospitals
 - From Local Government - Highways
 - From Local Government - Housing & Urban Renewal
 - From Local Government - Public Welfare
 - From Local Government - All Other
- 18) Sum of:
 - Airport charges
 - Natural Resource Charges
 - School Lunch Charges
 - School Tuition Charges
 - Local School Charges, NEC
 - Higher Education Charges, NEC
 - Hospital Charges
 - Highway Charges, NEC
 - Housing and Urban Renewal Charges
 - Parking Facilities Charges
 - Parks and Recreation Charges
 - Sewerage Charges
 - Sanitation Other Than Sewerage Charges
 - Water Transport and Terminal Charges
 - Other Charges
- 19) Sum of:
 - Special Assessments and Reimbursement for Capital Outlay
 - Sale of Property - Other
 - Interest Earnings
 - Miscellaneous General Revenue, NEC
- 20) Liquor Supply Revenue
- 21) Sum of:
 - Water Supply Revenue
 - Electric Power Revenue
 - Gas Supply Revenue
- 22) Transit Utility Revenue
- 23) Sum of:
 - Airports - Current Operations
 - Airports - Construction
 - Airports - Equipment and Land
 - Airports - Salaries and Wages
 - Airports - Intergovernmental Expend. - To state
 - Airports - Intergovernmental Expend. - To local gov.

- 24) Sum of:
 - Misc. Commercial Activities - Current Operations
 - Misc. Commercial Activities - Construction
 - Misc. Commercial Activities - Equipment & Land
 - Misc. Commercial Activities - Salaries and Wages
- 25) Sum of:
 - Correction, NEC - Current Operations
 - Correction, NEC - Equipment & Land
 - Correction, NEC - Construction
 - Correction, NEC - Salaries and Wages
 - Correction, NEC - Intergovernmental - To state
 - Correction, NEC - Intergovernmental - To local gov.
- 26) Education Expenditure From One School District to Another
- 27) Local Schools Current Operations
- 28) Sum of:
 - Local Schools - Construction
 - Local Schools - Equipment & Land
- 29) Local Schools - Salaries and Wages
- 30) Sum of:
 - Local Schools Intergovernmental - To state
 - Local Schools Intergovernmental - To local gov.
- 31) Higher Education - Current Operations
- 32) Sum of:
 - Higher Education - Construction
 - Higher Education - Equipment & Land
- 33) Higher Education - Salaries & Wages
- 34) Sum of:
 - Higher Education - Intergovernmental - To state
 - Higher Education - Intergovernmental - To local gov.
- 35) Education, NEC - Current Operations
- 36) Sum of:
 - Education, NEC - Construction
 - Education, NEC - Equipment & Land
- 37) Education, NEC - Salaries and Wages
- 38) Sum of:
 - Education, NEC - Intergovernmental - To state
 - Education, NEC - Intergovernmental - To local gov.
- 39) Education, NEC - Intergovernmental - To school dist.
- 40) Employment Security Admin. - Current Operations
- 41) Sum of:
 - Employment Security Admin. - Construction
 - Employment Security Admin. - Equipment & Land
- 42) Employment Security Admin. - Salaries and Wages
- 43) Financial Administration - Current Operations
- 44) Sum of:
 - Financial Administration - Construction
 - Financial Administration - Equipment & Land
- 45) Financial Administration - Salaries and Wages

- 46) Sum of:
 - Financial Admin. - Intergovernmental - To state
 - Financial Admin. - Intergovernmental - To local gov.
- 47) Fire Protection - Current Operations
- 48) Sum of:
 - Fire Protection - Construction
 - Fire Protection - Equipment & Land
- 49) Fire Protection - Salaries and Wages
- 50) Sum of:
 - Fire Protection - Intergovernmental - To state
 - Fire Protection - Intergovernmental - To local gov.
- 51) General Control, NEC - Current Operations
- 52) Sum of:
 - General Control, NEC - Construction
 - General Control, NEC - Equipment & Land
- 53) General Control, NEC - Salaries and Wages
- 54) Sum of:
 - General Control, NEC - Intergovern. - To state
 - General Control, NEC - Intergovern. - To local gov.
- 55) General Public Bldgs. - Current Operations
- 56) Sum of:
 - General Public Bldgs. - Construction
 - General Public Bldgs. - Equipment & Land
- 57) General Public Bldgs. - Salaries and Wages
- 58) Sum of:
 - General Public Bldgs. - Intergovernmental - To state
 - General Public Bldgs. - Intergovernmental - To local government
- 59) Health Other Than Hospitals - Current Operations
- 60) Sum of:
 - Health Other Than Hospitals - Construction
 - Health Other Than Hospitals - Equipment & Land
- 61) Health Other Than Hospitals - Salaries and Wages
- 62) Sum of:
 - Health Other Than Hospitals - Intergovernmental - To state
 - Health Other Than Hospitals - Intergovernmental - To local government
- 63) Own Hospitals, NEC - Current Operations
- 64) Sum of:
 - Own Hospitals, NEC - Construction
 - Own Hospitals, NEC - Equipment & Land
- 65) Own Hospitals, NEC - Salaries and Wages
- 66) Other Hospitals - Current Operations
- 67) Sum of:
 - Other Hospitals - Construction
 - Other Hospitals - Equipment & Land
- 68) Other Hospitals - Salaries and Wages

- 69) Sum of:
 - Other Hospitals - Intergovernmental - To state
 - Other Hospitals - Intergovernmental - To local gov.
- 70) Highways, NEC - Current Operations
- 71) Sum of:
 - Highways, NEC - Construction
 - Highways, NEC - Equipment & Land
- 72) Highways, NEC - Salaries and Wages
- 73) Sum of:
 - Highways, NEC - Intergovernmental - To state
 - Highways, NEC - Intergovernmental - To local gov.
- 74) Housing and Urban Renewal, NEC - Current Operations
- 75) Sum of:
 - Housing and Urban Renewal, NEC - Construction
 - Housing and Urban Renewal, NEC - Equipment & Land
- 76) Housing and Urban Renewal, NEC - Salaries and Wages
- 77) Sum of:
 - Housing and Urban Renewal, NEC - Intergovernmental - To state
 - Housing and Urban Renewal, NEC - Intergovernmental - To local government
- 78) Libraries - Current Operations
- 79) Sum of:
 - Libraries - Construction
 - Libraries - Equipment & Land
- 80) Libraries - Salaries and Wages
- 81) Sum of:
 - Libraries - Intergovernmental - To state
 - Libraries - Intergovernmental - To local gov.
- 82) Natural Resources, NEC - Current Operations
- 83) Sum of:
 - Natural Resources, NEC - Construction
 - Natural Resources, NEC - Equipment & Land
- 84) Natural Resources, NEC - Salaries and Wages
- 85) Sum of:
 - Natural Resources, NEC - Intergovernmental - To state
 - Natural Resources, NEC - Intergovernmental - To local government
- 86) Parking Facilities - Current Operations
- 87) Sum of:
 - Parking Facilities - Construction
 - Parking Facilities - Equipment & Land
- 88) Parking Facilities - Salaries and Wages
- 89) Sum of:
 - Parking Facilities - Intergovernmental - To state
 - Parking Facilities - Intergovernmental - To local gov.
- 90) Parks and Recreation - Current Operations

- 91) Sum of:
 - Parks and Recreation - Construction
 - Parks and Recreation - Equipment and Land
- 92) Parks and Recreation - Salaries and Wages
- 93) Sum of:
 - Parks and Recreation - Intergovernmental - To state
 - Parks and Recreation - Intergovernmental - To local government
- 94) Police Protection - Current Operations
- 95) Sum of:
 - Police Protection - Construction
 - Police Protection - Equipment & Land
- 96) Police Protection -Salaries and Wages
- 97) Sum of:
 - Police Protection - Intergovernmental - To state
 - Police Protection - Intergovernmental - To local gov.
- 98) Cash Assistance Payments, Categorical Programs
- 99) Cash Assistance Payments, NEC
- 100) Vendor Payments for Medical Care
- 101) Vendor Payments, NEC
- 102) Public Welfare, NEC - Current Operations
- 103) Sum of:
 - Public Welfare, NEC - Construction
 - Public Welfare, NEC - Equipment & Land
- 104) Public Welfare, NEC - Salaries and Wages
- 105) Sum of:
 - Public Welfare, NEC - Intergovernmental - To state
 - Public Welfare, NEC - Intergovernmental - To local gov.
- 106) Sewerage - Current Operations
- 107) Sum of:
 - Sewerage - Construction
 - Sewerage - Equipment and Land
- 108) Sewerage - Salaries and Wages
- 109) Sum of:
 - Sewerage - Intergovernmental - To state
 - Sewerage - Intergovernmental - To local government
- 110) Sanitation Other Than Sewerage - Current Operations
- 111) Sum of:
 - Sanitation Other Than Sewerage - Construction
 - Sanitation Other Than Sewerage - Equipment & Land
- 112) Sanitation Other Than Sewerage - Salaries and Wages
- 113) Sum of:
 - Sanitation Other Than Sewerage - Intergovernmental - To state
 - Sanitation Other Than Sewerage - Intergovernmental - To local government
- 114) Water Transport and Terminals - Current Operations

- 115) Sum of:
 - Water Transport and Terminals - Construction
 - Water Transport and Terminals - Equipment and Land
- 116) Water Transport and Terminals - Salaries and Wages
- 117) Sum of:
 - Water Transport and Terminals - Intergovernmental - To state
 - Water Transport and Terminals - Intergovernmental - To local government
- 118) General Government, NEC - Current Operations
- 119) Sum of:
 - General Government, NEC - Construction
 - General Government, NEC - Equipment & Land
- 120) General Government, NEC - Salaries and Wages
- 121) Interest on General Debt
- 122) Sum of:
 - General Government, NEC - Intergovernmental - To state
 - General Government, NEC - Intergovernmental - To local government
- 123) Liquor Stores - Current Operations
- 124) Sum of:
 - Liquor Stores - Construction
 - Liquor Stores - Equipment & Land
- 125) Liquor Stores - Salaries and Wages
- 126) Water Supply - Current Operations
- 127) Sum of:
 - Water Supply - Construction
 - Water Supply - Equipment & Land
- 128) Water Supply - Salaries and Wages
- 129) Water Supply - Interest
- 130) Electric Power - Current Operations
- 131) Sum of:
 - Electric Power - Construction
 - Electric Power - Equipment & Land
- 132) Electric Power - Salaries and Wages
- 133) Electric Power - Interest
- 134) Gas Supply - Current Operations
- 135) Sum of:
 - Gas Supply - Construction
 - Gas Supply - Equipment & Land
- 136) Gas Supply - Salary and Wages
- 137) Gas Supply - Interest
- 138) Transit Utility - Current Operation
- 139) Sum of:
 - Transit Utility - Construction
 - Transit Utility - Equipment & Land
- 140) Transit Utility - Salaries and Wages
- 141) Transit Utility - Interest
- 142) Personal Service Expenditure

- 143) Debt Issued, Not Classified by Character - Water Supply
- 144) Debt Issued, Not Classified by Character - Electric Power
- 145) Debt Issued, Not Classified by Character - Gas Supply
- 146) Debt Issued, Not Classified by Character - Transit System
- 147) Debt Issued, Not Classified by Character - General Debt
- 148) FFC Issued - Local Schools
- 149) FFC Issued - Higher Education
- 150) FFC Issued - Education, NEC
- 151) Non-Guaranteed Issued - Local Schools
- 152) Non-Guaranteed Issued - Higher Education
- 153) Non-Guaranteed Issued - Education, NEC
- 154) Long-Term Debt Retired, Not Classified by Character - Water Supply
- 155) Long-Term Debt Retired, Not Classified by Character - Electric Power
- 156) Long-Term Debt Retired, Not Classified by Character - Gas Supply System
- 157) Long-Term Debt Retired, Not Classified by Character - Transit System
- 158) Long-Term Debt Retired, Not Classified by Character - General Debt, NEC
- 159) FFC, Retired - Local Schools
- 160) FFC, Retired - Higher Education
- 161) FFC, Retired - Education, NEC
- 162) Non-Guaranteed - Retired - Local Schools
- 163) Non-Guaranteed - Retired - Higher Education
- 164) Non-Guaranteed - Retired - Education, NEC
- 165) FFC, At End of Year - Water Supply
- 166) FFC, At End of Year - Electric Power
- 167) FFC, At End of Year - Gas Supply
- 168) FFC, At End of Year - Transit Utility
- 169) FFC, At End of Year - Local Schools
- 170) FFC, At End of Year - Higher Education
- 171) FFC, At End of Year - Education, NEC
- 172) FFC, At End of Year - General Debt, NEC
- 173) Non-Guaranteed At End of Year, Water Supply
- 174) Non-Guaranteed At End of Year, Electric Power
- 175) Non-Guaranteed At End of Year, Gas Supply
- 176) Non-Guaranteed At End of Year, Transit System
- 177) Non-Guaranteed At End of Year, Local Schools
- 178) Non-Guaranteed At End of Year, Higher Education
- 179) Non-Guaranteed At End of Year, Education, NEC
- 180) Non-Guaranteed At End of Year, General Debt, NEC
- 181) Short-Term Debt At Beginning of Year
- 182) Short-Term Debt At End of Year

PERCAP --182 per capita values created by dividing the population
into each of the 182 fiscal items

5.1.3 PROGRAM -- SURV81.FORT

This program reads data from the tape called GOVT81.FIN.SUR.DATA located on account JSI6948 and writes the data to a sequential disk file for each state the program selects. Lines 210 and 290 will need to be modified to select the proper state (see Section 4.1.1 for state codes). This program is executed if the user is interested in calculating the per capita figures at the city, county, or school government levels.

INPUT: CN6948.JSI.GOV81.FIN.SUR.DATA

OUTPUT: CN6948.JSI.SUR81.XXXXXX.FINANCE.DATA (XXXXXX = state)

GLOSSARY OF VARIABLE NAMES:

IST1 --Government State Code

ICNTY --Government County number

II --Government State Code

JJ --Type of Government Code
values: 0 = state
1 = county
2 = municipality
3 = township
4 = special district
5 = school district

KK --Government County Code

LL --Unit Identification Code

MM --Depending on Type of Government,
If JJ = 0,1,2 or 3 then MM = Population
If JJ = 5 then MM = School Enrollment
If JJ = 4 then MM = Special School District
Function

NN --MSA CODE

NUMS1 --283 fields representing revenue, expenditure,
indebtedness, and cash and security holdings

XNAME2--Name of state, county, municipality, township, school district or special district

5.1.4 PROGRAM -- FINANCE.CITY.FORT

This program is the same as FINSTATE.FORT except the calculations are performed on the city level. The user will be prompted to 'ENTER THE 4 DIGIT COUNTY PSEUDONYM NUMBER'. The user must now enter the region number, e.g., 1601 = cold, 1761 = warm, 201 = arid. Next, the user will be prompted to 'ENTER THE 1 DIGIT FILE TYPE CODE'. Enter '2' for municipality. The user is then prompted to 'ENTER A 3 DIGIT COUNTY CODE OR 333 TO STOP'.

INPUT: CN6948.JSI.SUR81.XXXXXX.FINANCE.DATA (XXXXXX = state)

OUTPUT: CN6948.JSI.FXXXX.CITY.DATA (XXXX = region)

GLOSSARY OF VARIABLE NAMES:

IPSEUD	--4 digit county pseudonym number, e.g., 1601 = cold region, 1761 = warm region, 201 = arid region
ICODE	--1 digit file type code. The user must enter a '2', which is equal to the 'Type of Government' code, municipality (see section 5.1.3, Glossary of Variable Names)
IC	--3 digit county code
II	--Government State Code (see section 5.1.1)
JJ	--Type of Government Code 0 - state 1 - county 2 - municipality 3 - township 4 - special district 5 - school district
KK	--Government County Code
LL	--Unit Identification Code
POP	--If value for JJ is: 0, 1, 2, or 3 - Population for state, county, municipality, or township 4 - Population for special district 5 - School enrollment

NN --MSA code
 FIN & ARY1--An array of 283 elements representing Revenue,
 Expenditure, Indebtedness, and Cash and Security holdings
 SUMPOP --Total Population
 ARY2 --182 fiscal items (see section 5.1.2, Glossary of Variable
 Names)
 PERCAP --Each of the 182 fiscal items divided by the total
 population

5.1.5 PROGRAM -- FINANCE.COUNTY.FORT

This program is the same as FINANCE.CITY.FORT, except the calculations are performed at the county level rather than at the city level. The other difference is that when the user is prompted to 'ENTER THE 1 DIGIT FILE TYPE CODE', the user must enter '1' for county.

INPUT: CN6948.JSI.SUR81.XXXXXX.FINANCE.DATA (XXXXXX = state)

OUTPUT: CN6948.JSI.FXXXX.CNTY.FORT (XXXX = region number)

GLOSSARY OF VARIABLE NAMES:

IPSEUD --4 digit county pseudonym number, e.g. 1601 = cold region,
 1761 = warm region, 201 = arid region
 ICODE --1 digit file type code. The user will enter a '2', which
 is equal to the 'Type of Government' code, municipality
 (see section 5.1.3, Glossary of Variable Names)
 IC --3 digit county code
 II --Government State Code (see section 5.1.1)
 JJ --Type of Government Code
 0 - state
 1 - county
 2 - municipality
 3 - township
 4 - special district
 5 - school district
 KK --Government County Code
 LL --Unit Identification Code

POP --If value for JJ is:
 0, 1, 2, or 3 - Population for state, county,
 municipality, or township
 4 - Population for special district
 5 - School enrollment

NN --MSA code

FIN & ARY1--An array of 283 elements representing Revenue,
 Expenditure, Indebtedness, and Cash and Security holdings

SUMPOP --Total Population

ARY2 --182 fiscal items (see section 5.1.2, Glossary of Variable
 Names)

PERCAP --Each of the 182 fiscal items divided by the total
 population

5.1.6 PROGRAM -- FINANCE.SCHOOL.FORT

This program is the same as FINANCE.CITY.FORT, except the calculations are performed at the school government level. The other difference is that when the user is prompted to 'ENTER THE 1 DIGIT FILE TYPE CODE', the user must enter '5' for school district.

INPUT: CN6948.JSI.SUR81.XXXXXX.FINANCE.DATA (XXXXXX = state)

OUTPUT: CN6948.JSI.FXXXX.SCHOOL.DATA (XXXX = region number)

GLOSSARY OF VARIABLE NAMES:

IPSEUD --4 digit county pseudonym number, e.g., 1601 = cold region,
 1761 = warm region, 201 = arid region

ICODE --1 digit file type code. The user will enter a '2', which
 is equal to the 'Type of Government' code, municipality
 (see section 5.1.3, Glossary of Variable Names)

IC --3 digit county code

II --Government State Code (see section 5.1.1)

```

JJ      --Type of Government Code
        0 - state
        1 - county
        2 - municipality
        3 - township
        4 - special district
        5 - school district

KK      --Government County Code

LL      --Unit Identification Code

POP     --If value for JJ is:
        0, 1, 2, or 3 - Population for state, county,
                        municipality, or township, respectively
        4 - Population for special district
        5 - School enrollment

NN      --MSA code

FIN & ARY1--An array of 283 elements representing Revenue,
            Expenditure, Indebtedness, and Cash and Security holdings

SUMPOP  --Total Population

ARY2    --182 fiscal items (see section 5.1.2, Glossary of
            Variable Names)

PERCAP  --Each of the 182 fiscal items divided by the total
            population

```

5.1.7 PROGRAM -- FINANCE.COMBINE.FORT

This program would generally be used to combine data from two different states at the county level or the city level. This program will combine two county or two city output files created after executing FINANCE.COUNTY.FORT or FINANCE.CITY.FORT. The two files being combined must be of the same government sector and same scenario, e.g., both county or both city. The population figures from each file are added together as are the 182 fiscal items, e.g., fiscal item #1 from file a is added to fiscal item #1 of file b, etc. The total population of the two files combined is then divided into each of the 182 fiscal items after they have been combined in order to arrive at the per capita amounts for the two areas combined.

INPUTS: 1) File A--
 WWWXXXX.YYYY.ZZZZZZZZ.DATA

2) File B--
 WWWXXXX.YYYY.ZZZZZZZZ.DATA

FOR FILES A & B:

WWW = government sector (CNTY or CITY)
XXXX = region number, e.g., 201, 202, 1601, 1602,
 1761 or 1762
YYYY = region climate type
ZZZZZZZ = scenario, e.g., TREND, CONCASK or DRYWELL

OUTPUT: FXXXX.YYYYYY.DATA (Combination of files A & B)
 XXXX = region number
 YYYYYY = government sector

GLOSSARY OF VARIABLE NAMES

XLINE	--Dummy variable, used to read a line from the first file and write a line to the output file
YLINE	--Dummy variable, used to read a line from the second file
IPSEU1	--Pseudonym number for region, e.g., 201, 202, 1601, 1602, 1761 or 1762
L	--Integer values 1 - 182
ARY1	--Amounts for 182 fiscal items, first file
ARY2	--Amounts for 182 fiscal items, second file
PERCAP1	--Per capita amount, first file
PERCAP2	--Per capita amount, second file
ARY3	--Totals for each of 182 fiscal items, ARY1 + ARY2
PERCAP3	--For each of the 182 fiscal items, total of fiscal items (first file + second file)

5.1.8 PROGRAM -- FINANCE.COMBINE.SCHOOL.FORT

This program is the same as FINANCE.COMBINE.FORT except the two input files being combined are at the school government level.

INPUT: 1) File A
SCHLXXXX.YYYY.ZZZZZZZZ.DATA

2) File B
SCHLXXXX.YYYY.ZZZZZZZZ.DATA

For Files A & B:

XXXX = region number
YYYY = region climate type
ZZZZZZZ = scenario

OUTPUT: FSCHL.XXXX.FORT (Combination of file A & B)
XXXX = region number

GLOSSARY OR VARIABLE NAME:

XLINE	--Dummy variable, used to read a line from the first file and write a line to the output file
YLINE	--Dummy variable, used to read a line from the second file
IPSEU1	--Pseudonym number for region, e.g., 201, 202, 1601, 1602, 1761 or 1762
L	--Integer values 1 - 182
ARY1	--Amounts for 182 fiscal items, first file
ARY2	--Amounts for 182 fiscal items, second file
PERCAP1	--Per capita amount, first file
PERCAP2	--Per capita amount, second file
ARY3	--Totals for each of 182 fiscal items, ARY1 + ARY2
PERCAP3	--For each of the 182 fiscal items, total of fiscal items (first file + second file)

5.2 PROGRAM -- FISCAL.FORT

This program reads in population data and per capita data from two separate input files. The population data consists of the cross region, the year, total population, and total population for ages 5-17 (representing school age children). The years are 1975-2030, a total of 56 years. The per capita

data consists of various aggregated income and expense data --a total of 182 separate aggregations. The population is multiplied by each per capita amount for each year and then by 1.1821 to bring it up to 1985 dollars. This result is written to an output file if it is not equal to 0, along with the total revenue and total expenses for each year. Per capita data numbered 1-22 were used to calculate total revenue and those numbered 23-141 were used to calculate total expenses.

Lines 52, 70 and 110 need to be edited. Line 52 is the Heading that appears at the top of the output file and should be edited to correspond with the output data. In line 70, 'Z' should only be present when executing school data. Line 110 should have a '3' when executing for State and School data and should have a '4' when executing for County and City data.

INPUT: 1) POPXXXX.YYYY.ZZZZZZZZ

XXXX = region number, e.g., 201, 202, 1601, 1602, 1761 or 1762

YYYY = region climate type, e.g., ARID, COLD or WARM

ZZZZZZZ = scenario, e.g., TREND, CONCASK or DRYWELL

2) XXXXXX.STATE.FINANCE.DATA

XXXXXX = state

- or -

FXXXX.YYYYYY.DATA

XXXX = region number

YYYY = government sector, e.g., COUNTY, CITY or SCHOOL

OUTPUT: WWWXXXXX.YYYY.ZZZZZZZZ.DATA

WWW = government sector, e.g., STAT, CNTY, CITY or SCHL

XXXX = region number, e.g., 201, 202, 1601, 1602, 1761 or 1762

YYYY = region climate type, e.g., ARID, COLD or WARM

ZZZZZZZ = scenario, e.g., TREND, CONCASK or DRYWELL

GLOSSARY OF VARIABLE NAMES:

X --region number

Y --year

Z --total population (if present in line 70)

POP --if 'Z' is present in line 70, POP = population on school age children, ages 5 - 17, otherwise, POP = total population

REV --total revenue

EXP	--total expenditures
ARRAY	--dummy variable, used for reading a line of data
I	--integers, 1 - 182
JJ	--year
PERCAP	--per capita amount
AMT	--per capita * population * 1.1821

5.3 MERGING AND EDITING SCENARIO FILES

After the output files are created when executing FISCAL.FORT, the user must merge the scenario files into one file. This is then edited so that only the total revenue and expenses remain in the file. Follow the steps below to merge and edit using TSO. The output file used as an example in the steps 1 through 10 shown below is STAT201.ARID.TREND.DATA, using two impact scenarios (CONCASK and DRYWELL) and a baseline scenario (TREND).

- 1) COPY STAT201.ARID.TREND.DATA TEMP.DATA (TEMP.DATA is used as a holding file so the user can delete all lines except the revenue and expense totals)
- 2) EDIT TEMP.DATA NONUM
- 3) COUNT (displays total number of lines in file)
- 4) DEL * 5100 (subtract 56 from total number lines, e.g. 5156 (total number of lines) - 56 (number of years) = 5100)
- 5) L * (should list first year of revenue & expense totals, which is 1975, if not, start over with step 1)
- 6) END S
- 7) MERGE TEMP.DATA TCD201.XXXX.STATE.DATA
(XXXX = region climate type)
- 8) Repeat steps 1 through 7 above for STAT201.XXXX.CONCASKS.DATA and STAT201.XXXX.DRYWELL.DATA (XXXX = region climate type).
- 9) COUNT TCD201.ARID.STATE.DATA (total number of lines should be 168)
- 10) Using SUPERWYLBUR, edit TCD201.STATE.DATA as follows:
 1. USE FR TCD201.ARID.STATE.DATA

2. CH ' TOTAL REVENUE = ' TO ' ' IN 1/L NOLIST (use 2 single quotes)
3. CH ` TOTAL EXPENSE = ' TO ' ' IN 1/L NOLIST (use 2 single quotes)
4. CH '0000000' TO ' ' IN 1/L NOLIST (use 2 single quotes)
5. RESAVE CARD

The remaining scenario files now will need to be merged and edited for each government sector within each region.

5.4 PROGRAM -- REREPS.FORT

This program produces two revenue and expense reports. One report simply reads the data from the input file and rearranges it into a new format. The other report calculates differences between impact and baseline scenarios for the total revenue and total expenditures for the years 1975 - 2030.

Lines 40 and 100 of the program must be edited to reflect the government sector and the region of the input file.

INPUT: TCDXXXX.YYYY.ZZZZZZ.DATA

XXXX = region number, e.g., 201, 202, 1601, 1602, 1761 or 1762

YYYY = region climate type, e.g., ARID, COLD or WARM

ZZZZZZ = government sector (STATE, COUNTY, CITY OR SCHOOL)

OUTPUT: 1) REVEXP.XXXXXX.TCDYYYY.DATA

XXXXXX = government sector (STATE, COUNTY, CITY or SCHOOL)

YYYY = region number, e.g., 201, 202, 1601, 1602, 1761
or 1762

2) REVEXP.XXXXXX.TCDYYYY.DIF.DATA

XXXXXX = government sector (STATE, COUNTY, CITY or SCHOOL)

YYYY = region number, e.g., 201, 202, 1601, 1602, 1761
or 1762

GLOSSARY OF VARIABLE NAMES:

KK --year

TREV --revenue - Trend

TEXP --expend. - Trend

CREV --revenue - Concasks

CEXP --expend. - Concasks

DREV --revenue - Drywell

DEXP --expend. - Drywell

DCASKR --revenue - Concasks minus Trend

DDRYR --revenue - Drywell minus Trend

DCASKE --expend. - Concasks minus Trend

DDRYE --expend. - Drywell minus Trend

5.5 PROGRAM -- RECAT.SUMMARY.FORT

This program creates a summary report. The 182 fiscal items are aggregated further into new revenue, expense and debt items. For each new item, the discount dollars and the present values are calculated.

Lines 2450 and 2460 must be edited to reflect the government sector (state, count, city or school), the climate type (arid, cold, warm) and the scenario (trend, concask and drywell) of the input file. The date must also be edited on line 2460 to reflect the date that the program is being executed on.

The output file created by executing FISCAL.FORT must now be copied to a temporary file, e.g., TEMP.DATA. The temporary file must then be edited by deleting the revenue and expense totals at the end of the file. The steps that follow illustrate the use of TSO to create and edit the temporary file and to execute a clist that executes RECAT.SUMMARY.FORT.

- 1) Type: COPY WWWXXXX.YYYY.ZZZZZZZZ.DATA TEMP.DATA
 WWW = government sector (STAT, CNTY, CITY, or SCHL)
 XXXX = region number, e.g., 201, 202, 1601, 1602,
 1761 or 1762
 YYYY = region climate type, e.g., ARID, COLD OR WARM
 ZZZZZZZ = scenario, e.g., TREND, CONCASKS or DRYWELL
- 2) Type: EDIT TEMP.DATA NONUM
- 3) Type: V
- 4) Type : F 'TOTAL'
- 5) Type: DEL * 56
 (56 = number of years)
- 6) Type: END S
 (saves file)

- 7) The clist used to execute the program RECAT.SUMMARY.FORT should look like the following:

```
00010 FREEALL
00020 ATTR AA RECFM(F B) LRECL(180) BLKSIZE(1800) DSORG(PS)
00030 ALLOC F(FT07F001) DS(TEMP.DATA)
00040 ALLOC F(FT08F001) DS(RECAT.SUM.WWWXXXXX.ZZZZZZZZ.DATA)
00050 RUN RECAT.SUMMARY FORT
```

- 8) Type: EDIT RECAT.SUMMARY FORTG

Edit lines 2450 and 2460 as described above and then save file

- 9) Type: EX RECAT.SUM

RECAT.SUM is the clist

- 10) Type: %CONTROL C

This is a clist used to check for control characters so paging will occur. Enter file name when prompted to do so.

INPUT: Created by copying the output file created by executing FISCAL.FORT to a temporary file and deleting the revenue and expense totals at the end of the temporary file

OUTPUT: RECAT.SUM.XXXXXYYY.ZZZZZZZZ.DATA
XXXX = government sector (STAT, CNTY, CITY or SCHL)
YYYY = region number, e.g., 201, 202, 1601, 1602, 1761 or 1762
ZZZZZZZ = scenario, e.g., TREND, CONCASKS or DRYWELL

GLOSSARY OF VARIABLE NAMES

ICAT	--fiscal item number (see section 5.1.2)
IYEAR	--year
AMOUNT	--total cost
AMT	--two-dimensional array containing AMOUNT, using YEAR and fiscal item number as subscripts
REV	--new aggregated revenue categories
	1) Property Tax (fiscal item 1)
	2) Sales Tax (fiscal item 2)
	3) Misc. Tax (Sum of fiscal items, 3 - 5)
	4) Transfer Payments - Federal (Sum of fiscal items, 6-10)

5) Transfer Payments - State (Sum of fiscal items,
 11 - 15)
 6) Transfer Payments - Local (Sum of fiscal items,
 16 - 17) 7)
 Misc. Revenue (Sum of MRS fiscal categories,
 18 - 22)
 EXPC --Capital expense
 EXPNC --Non-Capital expense

Capital and Non-Capital expense items are calculated as follows:

- 1) Education -
 - Capital - Sum of fiscal items:
28, 32, 36
 - Non-Capital - Sum of fiscal items:
26, 27, 29 - 31, 33 - 35, 37 - 39
- 2) Public Safety -
 - Capital - Sum of fiscal items:
48, 95
 - Non-Capital - Sum of fiscal items:
26, 27, 29 - 31, 33 - 35, 37 - 39
- 3) Public Assistance -
 - Capital - Sum of fiscal items:
75, 103
 - Non-Capital - Sum of fiscal items:
74, 76, 77, 98 - 102, 104, 105
- 4) Parks & Recreation -
 - Capital - fiscal item: 91
 - Non-Capital - Sum of fiscal items:
90, 92, 93
- 5) Administration -
 - Capital - Sum of fiscal items:
41, 44, 52, 56, 119
 - Non-Capital - Sum of fiscal items:
40, 42, 43, 45, 46, 51, 53, 54,
55, 57, 58, 118, 120, 121
- 6) Health -
 - Capital - Sum of fiscal items:
60, 64, 67
 - Non-Capital - Sum of fiscal items:
59, 61 - 63, 65, 66, 68, 69

- 7) Sanitation -
 - Capital - Sum of fiscal items:
107, 111
 - Non-Capital - Sum of fiscal items:
106, 108 - 110, 112, 113
- 8) Utilities -
 - Capital - Sum of fiscal items:
127, 131, 135
 - Non-Capital - Sum of fiscal items:
126, 128 - 130, 132 - 134, 136, 137
- 9) Transportation -
 - Capital - Sum of fiscal items:
71, 87, 115, 139
 - Non-Capital - Sum of fiscal items:
23, 70, 72, 73, 86, 88, 89, 114,
116, 117, 138, 140, 141
- 10) Libraries -
 - Capital - fiscal item: 79
 - Non-Capital - Sum of fiscal items:
78, 80, 81
- 11) Natural Resources -
 - Capital - fiscal item: 83
 - Non-Capital - Sum of fiscal items:
82, 84, 85
- 12) Misc. Expenditures -
 - Capital - fiscal item: 124
 - Non-Capital - Sum of fiscal items:
24, 123, 125, 126

EXPTOT --Total expenses, one total for each of the 12 categories

DEBT --Debt
Categories are calculated as follows:

- 1) Issued - Sum of fiscal items: 143-153
- 2) Retired - Sum of fiscal items: 154-164
- 3) Full Faith and Credit - Sum of fiscal items:
165-172
- 4) Non-Guaranteed - Sum of fiscal items:
173-180
- 5) Short Term-Beg. of Year - fiscal item: 181
- 6) Short Term-End of Year - fiscal item: 182

II --Time index to make 1975 = 0

T	--Forecast year (T = 0 in 1985)
R	--Real Discount Rate (adjusted for inflation)
Q	--Discount factor
DREV	--Discounted value, revenue
DEXPC	--Discounted value, capital expenses
DEXPNC	--Discounted value, non-capital expenses
DEXTOT	--Discounted value, total expenses
DDEBT	--Discounted value, debt
PVR	--Present value, revenue
PVEC	--Present value, capital expenses
PVENC	--Present value, non-capital expenses
PVETOT	--Present value, total expenses
PVD	--Present value, debt

APPENDIX A

PROGRAM LISTINGS

L FR FINANCE.CITY.FORT

```

1. C*****
2. C PROGRAMER: JANICE SCHWAB AND KEN BORK 5/21/84 *
3. C PROGRAM: FINANCE.FORT *
4. C THIS PROGRAM WILL READ SURVEY OF GOVT.FINANCE DATA AND *
5. C AGGREGATE IT AND CALCULATE THE PER CAPITA NUMBERS. *
6. C*****
7. C
8. DOUBLE PRECISION ARY1(283),ARY2(182),PERCAP(182),FIN(283)
9. DIMENSION XNAME(8)
10. DO 500 NN = 1,182
11. ARY2(NN) = 0.0
12. ARY1(NN) = 0.0
13. PERCAP(NN) = 0.0
14. 300 CONTINUE
15. WRITE(6,320)
16. 320 FORMAT(1X,'ENTER THE 4 DIGIT COUNTY PSEUDONYM NUMBER')
17. READ(5,330) IPSEUD
18. 330 FORMAT(I4)
19. C
20. WRITE(6,325)
21. 325 FORMAT(1X,'ENTER THE 1 DIGIT FILE TYPE CODE')
22. READ(5,335) ICODE
23. 335 FORMAT(I1)
24. C
25. 10 WRITE(6,340)
26. 340 FORMAT(1X,'ENTER A 3 DIGIT COUNTY CODE OR 333 TO STOP')
27. READ(5,310) IC
28. 310 FORMAT(I3)
29. IF(IC.EQ.333) GOTO 333
30. C
31. 11 READ(2,100,END=333) II,JJ,KK,LL,POP,NN,(FIN(I),I=1,283),
32. & (XNAME(J),J=1,8)
33. 100 FORMAT(I2,1X,I1,1X,I3,1X,I3,1X,F12.0,1X,I6,/,
34. & 35(8(F12.0,1X),/),3(F12.0,1X),8A4)
35. IF(ICODE.NE.JJ) GOTO 11
36. IF(KK.GT.IC) GOTO 12
37. IF(IC.NE.KK) GOTO 11
38. SUMPOP = SUMPOP + POP
39. DO 155 M = 1,283
40. ARY1(M) = ARY1(M) + FIN(M)
41. 155 CONTINUE
42. GOTO 10
43. 12 DO 615 M = 1,37
44. BACKSPACE 2
45. 615 CONTINUE
46. GOTO 10
47. C AGGREGATE THE NUMBERS
48. C
49. 333 ARY2(1) = ARY1(1)
50. ARY2(2) = ARY1(2)
51. DO 40 I = 3,7
52. ARY2(3) = ARY2(3) + ARY1(I)
53. 40 CONTINUE
54. ARY2(4) = ARY1(8) + ARY1(9)
55. ARY2(5) = ARY1(10) + ARY1(11)
56. ARY2(6) = ARY1(12)

```

37.	ARY2(7) = ARY1(13)	00000560
38.	ARY2(8) = ARY1(14)	00000570
39.	ARY2(9) = ARY1(15)	00000580
40.	ARY2(10) = ARY1(16) + ARY1(17) + ARY1(18)	00000590
41.	ARY2(11) = ARY1(19) + ARY1(20)	00000600
42.	ARY2(12) = ARY1(21)	00000610
43.	ARY2(13) = ARY1(22)	00000620
44.	ARY2(14) = ARY1(23)	00000630
45.	ARY2(15) = ARY1(24) + ARY1(25) + ARY1(26)	00000640
46.	ARY2(16) = ARY1(27)	00000650
47.	DO 45 I = 28,36	00000660
48.	ARY2(17) = ARY2(17) + ARY1(I)	00000670
49.	45 CONTINUE	00000680
50.	DO 50 I = 37,51	00000690
51.	ARY2(18) = ARY2(18) + ARY1(I)	00000700
52.	50 CONTINUE	00000710
53.	DO 60 I = 52,55	00000720
54.	ARY2(19) = ARY2(19) + ARY1(I)	00000730
55.	60 CONTINUE	00000740
56.	ARY2(20) = ARY1(56)	00000750
57.	ARY2(21) = ARY1(57) + ARY1(58) + ARY1(59)	00000760
58.	ARY2(22) = ARY1(60)	00000770
59.	DO 70 I = 61,66	00000780
60.	ARY2(23) = ARY2(23) + ARY1(I)	00000790
61.	70 CONTINUE	00000800
62.	ARY2(24) = ARY1(67) + ARY1(68) + ARY1(69) + ARY1(70)	00000810
63.	DO 80 I = 71,76	00000820
64.	ARY2(25) = ARY2(25) + ARY1(I)	00000830
65.	80 CONTINUE	00000840
66.	ARY2(26) = ARY1(77)	00000850
67.	ARY2(27) = ARY1(78)	00000860
68.	N = 28	00000870
69.	NN = 79	00000880
70.	DO 90 I = 1,6	00000890
71.	NNN = NN + 1	00000900
72.	NN = N + 1	00000910
73.	NNN = NN + 2	00000920
74.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00000930
75.	ARY2(NN) = ARY1(NNN)	00000940
76.	N = N + 2	00000950
77.	NN = NN + 3	00000960
78.	90 CONTINUE	00000970
79.	ARY2(40) = ARY1(97)	00000980
80.	ARY2(41) = ARY1(98) + ARY1(99)	00000990
81.	ARY2(42) = ARY1(100)	00001000
82.	ARY2(43) = ARY1(101)	00001010
83.	N = 44	00001020
84.	NN = 102	00001030
85.	DO 101 I = 1,11	00001040
86.	NN = N + 1	00001050
87.	NNN = NN + 1	00001060
88.	NNN = NN + 2	00001070
89.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00001080
90.	ARY2(NN) = ARY1(NNN)	00001090
91.	N = N + 2	00001100
92.	NN = NN + 3	00001110
93.	101 CONTINUE	00001120

114.	ARY2(66) = ARY1(135)	00001130
115.	N = 67	00001140
116.	NN = 136	00001150
117.	DO 110 I = 1,16	00001160
118.	NN = N + 1	00001170
119.	NNN = NN + 1	00001180
120.	NNN = NN + 2	00001190
121.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00001200
122.	ARY2(NN) = ARY1(NNN)	00001210
123.	N = N + 2	00001220
124.	NN = NN + 3	00001230
125.	110 CONTINUE	00001240
126.	ARY2(99) = ARY1(184)	00001250
127.	ARY2(100) = ARY1(185)	00001260
128.	ARY2(101) = ARY1(186)	00001270
129.	ARY2(102) = ARY1(187)	00001280
130.	ARY2(103) = ARY1(188) + ARY1(189)	00001290
131.	ARY2(104) = ARY1(191)	00001300
132.	N = 105	00001310
133.	NN = 192	00001320
134.	DO 120 I = 1,8	00001330
135.	NN = N + 1	00001340
136.	NNN = NN + 1	00001350
137.	NNN = NN + 2	00001360
138.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00001370
139.	ARY2(NN) = ARY1(NNN)	00001380
140.	N = N + 2	00001390
141.	NN = NN + 3	00001400
142.	120 CONTINUE	00001410
143.	ARY2(121) = ARY1(216)	00001420
144.	ARY2(122) = ARY1(217) + ARY1(218)	00001430
145.	ARY2(123) = ARY1(219)	00001440
146.	ARY2(124) = ARY1(220) + ARY1(221)	00001450
147.	ARY2(125) = ARY1(222)	00001460
148.	ARY2(126) = ARY1(223)	00001470
149.	N = 127	00001480
150.	NN = 224	00001490
151.	DO 130 I = 1,4	00001500
152.	NN = N + 1	00001510
153.	NNN = N + 2	00001520
154.	NNNN = N + 3	00001530
155.	NNN = NN + 1	00001540
156.	NNNN = NN + 2	00001550
157.	NNNNN = NN + 3	00001560
158.	NNNNN = NN + 4	00001570
159.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00001580
160.	ARY2(NNN) = ARY1(NNNN)	00001590
161.	ARY2(NNN) = ARY1(NNNNN)	00001600
162.	ARY2(NNNN) = ARY1(NNNNNN)	00001610
163.	N = N + 4	00001620
164.	NN = NN + 5	00001630
165.	130 CONTINUE	00001640
166.	N = 143	00001650
167.	DO 140 I = 244,283	00001660
168.	ARY2(N) = ARY1(I)	00001670
169.	N = N + 1	00001680
170.	140 CONTINUE	00001690

171.	C		00001700
172.	C	PRINT REPORT	00001710
173.	C		00001720
174.		IF(JJ.EQ.5) GOTO 266	00001730
175.		WRITE(3,205)	00001731
176.		205 FORMAT(1X,'CITY')	00001732
177.		WRITE(3,200) IPSEUD,SUMPOP	00001740
178.		200 FORMAT('1',//,1X,15,5X,'POPULATION',3X,F12.0)	00001750
179.		GOTO 202	00001760
180.		266 WRITE(3,201)IPSEUD,SUMPOP	00001770
181.		201 FORMAT('1',//,15,'SCHOOL',5X,'NUMBER OF STUDENTS',3X,F12.0)	00001780
182.		202 WRITE(3,210)	00001790
183.		210 FORMAT('0',17X,'TOTAL',10X,'PER CAPITA')	00001800
184.		DO 150 I = 1,182	00001810
185.		PERCAP(I) = ARY2(I)/SUMPOP	00001820
186.		WRITE(3,220) I,ARY2(I),PERCAP(I)	00001830
187.		220 FORMAT(11X,13,2X,F12.0,5X,F9.2)	00001840
188.		150 CONTINUE	00001850
189.	C	GOTO 10	00001860
190.		999 STOP	00001870
191.		DEBUG INIT(IPSEUD,IC,ICODE,II,JJ,XX,LL,SUMPOP)	00001880
192.		END	00001890

27

L FR FINANCE.COMBINE.FORT

```

1. C PROGRAMER: JANICE SCHWAB 11/26/84 * 00000020
2. C PROGRAM: FINANCE.COMBINE.FORT * 00000030
3. C THIS PROGRAM WILL COMBINE TWO AGGREGATED FINANCE FILES. * 00000040
4. C***** 00000050
5. C 00000060
6. DOUBLE PRECISION ARY1(182),ARY2(182),ARY3(182),PERCA1(182), 00000070
7. & PERCA2(182),PERCA3(182),XLINE(20),YLINE(20) 00000080
8. C 00000090
9. DO 500 II = 1,3 00000140
10. READ(2,200)(XLINE(I),I=1,20) 00000150
11. READ(3,200)(YLINE(J),J=1,20) 00000160
12. WRITE(4,200)(XLINE(K),K=1,20) 00000170
13. WRITE(6,200)(XLINE(K),K=1,20) 00000175
14. 200 FORMAT(20A4) 00000180
15. 500 CONTINUE 00000190
16. C 00000200
17. READ(2,210) IPSEU1,SUMPO1 00000210
18. READ(3,210) IPSEU2,SUMPO2 00000220
19. 210 FORMAT(1X,15,18X,F12.0) 00000230
20. SUMPO3 = SUMPO1+SUMPO2 00000240
21. WRITE(4,220) IPSEU1,SUMPO3 00000250
22. WRITE(6,220) IPSEU1,SUMPO3 00000255
23. 220 FORMAT(1X,15,5X,'POPULATION',3X,F12.0) 00000260
24. C 00000270
25. READ(2,200)(XLINE(I),I=1,20) 00000280
26. READ(3,200)(YLINE(J),J=1,20) 00000290
27. WRITE(4,200)(XLINE(K),K=1,20) 00000300
28. WRITE(6,200)(XLINE(K),K=1,20) 00000305
29. C 00000310
30. DO 150 I = 1,182 00000320
31. READ(2,230) L,ARY1(I),PERCA1(I) 00000330
32. READ(3,230) L,ARY2(I),PERCA2(I) 00000340
33. 230 FORMAT(11X,13,2X,F12.0,5X,F12.2) 00000350
34. ARY3(I) = ARY1(I) + ARY2(I) 00000360
35. PERCA3(I) = ARY3(I) / SUMPO3 00000370
36. WRITE(4,230) L,ARY3(I),PERCA3(I) 00000380
37. 150 CONTINUE 00000390
38. C DEBUG INIT(IPSEU1,IPSEU2,SUMPO1,SUMPO2,SUMPO3,ARY1,ARY2,ARY3, 00000400
39. C & PERCA1,PERCA2,PERCA3) 00000401
40. END 00000410

```


L FR FINANCE.COMBINE.SCHOOL.FORT

```

1. C***** 00000010
2. C PROGRAMER: JANICE SCHWAB 11/26/84 * 00000020
3. C PROGRAM: FINANCE.COMBINE.FORT * 00000030
4. C THIS PROGRAM WILL COMBINE TWO AGGREGATED FINANCE FILES. * 00000040
5. C***** 00000050
6. C 00000060
7. DOUBLE PRECISION ARY1(182),ARY2(182),ARY3(182),PERCA1(182), 00000070
8. & PERCA2(182),PERCA3(182),XLINE(20),YLINE(20) 00000080
9. C 00000090
10. DO 500 II = 1,3 00000140
11. READ(2,200)(XLINE(I),I=1,20) 00000150
12. READ(3,200)(YLINE(J),J=1,20) 00000160
13. WRITE(4,200)(XLINE(K),K=1,20) 00000170
14. WRITE(6,200)(XLINE(K),K=1,20) 00000175
15. 200 FORMAT(20A4) 00000180
16. 500 CONTINUE 00000190
17. - C 00000200
18. READ(2,210) IPSEU1,SUMF01 00000210
19. READ(3,210) IPSEU2,SUMF02 00000220
20. 210 FORMAT(1X,I4,32X,F12.0) 00000230
21. SUMF03 = SUMF01+SUMF02 00000240
22. WRITE(4,220) IPSEU1,SUMF03 00000250
23. WRITE(6,220) IPSEU1,SUMF03 00000255
24. 220 FORMAT(1X,I5,2X,'SCHOOL',3X,'NUMBER OF STUDENTS',F12.0) 00000260
25. C 00000270
26. READ(2,200)(XLINE(I),I=1,20) 00000280
27. READ(3,200)(YLINE(J),J=1,20) 00000290
28. WRITE(4,200)(XLINE(K),K=1,20) 00000300
29. WRITE(6,200)(XLINE(K),K=1,20) 00000305
30. C 00000310
31. DO 150 I = 1,182 00000320
32. READ(2,230) L,ARY1(I),PERCA1(I) 00000330
33. READ(3,230) L,ARY2(I),PERCA2(I) 00000340
34. 230 FORMAT(11X,I3,2X,F12.0,5X,F9.2) 00000350
35. ARY3(I) = ARY1(I) + ARY2(I) 00000360
36. PERCA3(I) = ARY3(I) / SUMF03 00000370
37. WRITE(4,230) L,ARY3(I),PERCA3(I) 00000380
38. 150 CONTINUE 00000390
39. C DEBUG INIT(IPSEU1,IPSEU2,SUMF01,SUMF02,SUMF03,ARY1,ARY2,ARY3, 00000400
40. C & PERCA1,PERCA2,PERCA3) 00000401
41. END 00000410
2?

```

1. FR FINANCE.COUNTY.FORT

```

1. C***** 00000010
2. C PROGRAM: JANICE SCHWAB AND KEN BURK 6/21/84 * 00000020
3. C PROGRAM: FINANCE.FORT * 00000030
4. C THIS PROGRAM WILL READ SURVEY OF GOVT.FINANCE DATA AND * 00000040
5. C AGGREGATE IT AND CALCULATE THE PER CAPITA NUMBERS. * 00000050
6. C***** 00000060
7. C 00000070
8. DOUBLE PRECISION ARY1(283),ARY2(182),PERCAP(182),FIN(283) 00000080
9. DIMENSION XNAME(8) 00000090
10. DO 500 NN = 1,182 00000100
11. ARY2(NN) = 0.0 00000110
12. ARY1(NN) = 0.0 00000115
13. PERCAP(NN) = 0.0 00000120
14. 300 CONTINUE 00000130
15. WRITE(6,320) 00000140
16. 320 FORMAT(1X,'ENTER THE 4 DIGIT COUNTY PSEUDONYM NUMBER') 00000150
17. READ(5,330) IPSELD 00000160
18. 330 FORMAT(I4) 00000170
19. C 00000180
20. WRITE(6,325) 00000190
21. 325 FORMAT(1X,'ENTER THE 1 DIGIT FILE TYPE CODE') 00000200
22. READ(5,335) ICODE 00000210
23. 335 FORMAT(I1) 00000220
24. C 00000230
25. 10 WRITE(6,340) 00000240
26. 340 FORMAT(1X,'ENTER A 3 DIGIT COUNTY CODE OR 333 TO STOP') 00000250
27. READ(5,310) IC 00000260
28. 310 FORMAT(I3) 00000270
29. IF(IC.EQ.333) GOTO 333 00000280
30. C 00000290
31. 11 READ(2,100,END=333) II,JJ,KK,LL,POP,IN,(FIN(I),I=1,283), 00000300
32. & (XNAME(J),J=1,8) 00000310
33. 100 FORMAT(I2,1X,I1,1X,I3,1X,I3,1X,F12.0,1X,I6,/, 00000320
34. & 35(8(F12.0,1X),/),3(F12.0,1X),8A4) 00000330
35. IF(ICODE.NE.JJ) GOTO 11 00000335
36. IF(KK.GT.IC) GOTO 12 00000340
37. IF(IC.NE.KK) GOTO 11 00000350
38. SUMPOP = SUMPOP + POP 00000370
39. DO 155 M = 1,283 00000380
40. ARY1(M) = ARY1(M) + FIN(M) 00000390
41. 155 CONTINUE 00000400
42. GOTO 10 00000410
43. 12 DO 615 M = 1,37 00000420
44. BACKSPACE 2 00000430
45. 615 CONTINUE 00000440
46. GOTO 10 00000450
47. C AGGREGATE THE NUMBERS 00000460
48. C 00000470
49. 333 ARY2(1) = ARY1(1) 00000480
50. ARY2(2) = ARY1(2) 00000490
51. DO 40 I = 3,7 00000500
52. ARY2(3) = ARY2(3) + ARY1(I) 00000510
53. 40 CONTINUE 00000520
54. ARY2(4) = ARY1(8) + ARY1(9) 00000530
55. ARY2(5) = ARY1(10) + ARY1(11) 00000540
56. ARY2(6) = ARY1(12) 00000550

```

57.	ARY2(7) = ARY1(13)	00000560
58.	ARY2(8) = ARY1(14)	00000570
59.	ARY2(9) = ARY1(15)	00000580
60.	ARY2(10) = ARY1(16) + ARY1(17) + ARY1(18)	00000590
61.	ARY2(11) = ARY1(19) + ARY1(20)	00000600
62.	ARY2(12) = ARY1(21)	00000610
63.	ARY2(13) = ARY1(22)	00000620
64.	ARY2(14) = ARY1(23)	00000630
65.	ARY2(15) = ARY1(24) + ARY1(25) + ARY1(26)	00000640
66.	ARY2(16) = ARY1(27)	00000650
67.	DO 45 I = 28,36	00000660
68.	ARY2(17) = ARY2(17) + ARY1(I)	00000670
69.	45 CONTINUE	00000680
70.	DO 50 I = 37,51	00000690
71.	ARY2(18) = ARY2(18) + ARY1(I)	00000700
72.	50 CONTINUE	00000710
73.	DO 60 I = 52,55	00000720
74.	ARY2(19) = ARY2(19) + ARY1(I)	00000730
75.	60 CONTINUE	00000740
76.	ARY2(20) = ARY1(56)	00000750
77.	ARY2(21) = ARY1(57) + ARY1(58) + ARY1(59)	00000760
78.	ARY2(22) = ARY1(60)	00000770
79.	DO 70 I = 61,66	00000780
80.	ARY2(23) = ARY2(23) + ARY1(I)	00000790
81.	70 CONTINUE	00000800
82.	ARY2(24) = ARY1(67) + ARY1(68) + ARY1(69) + ARY1(70)	00000810
83.	DO 80 I = 71,76	00000820
84.	ARY2(25) = ARY2(25) + ARY1(I)	00000830
85.	80 CONTINUE	00000840
86.	ARY2(26) = ARY1(77)	00000850
87.	ARY2(27) = ARY1(78)	00000860
88.	N = 28	00000870
89.	NN = 79	00000880
90.	DO 90 I = 1,6	00000890
91.	NNN = NN + 1	00000900
92.	NN = N + 1	00000910
93.	NNN = NN + 2	00000920
94.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00000930
95.	ARY2(NN) = ARY1(NNN)	00000940
96.	N = N + 2	00000950
97.	NN = NN + 3	00000960
98.	90 CONTINUE	00000970
99.	ARY2(40) = ARY1(97)	00000980
100.	ARY2(41) = ARY1(98) + ARY1(99)	00000990
101.	ARY2(42) = ARY1(100)	00001000
102.	ARY2(43) = ARY1(101)	00001010
103.	N = 44	00001020
104.	NN = 102	00001030
105.	DO 101 I = 1,11	00001040
106.	NN = N + 1	00001050
107.	NNN = NN + 1	00001060
108.	NNN = NN + 2	00001070
109.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00001080
110.	ARY2(NN) = ARY1(NNN)	00001090
111.	N = N + 2	00001100
112.	NN = NN + 3	00001110
113.	101 CONTINUE	00001120

114.	ARY2(66) = ARY1(135)	00001130
115.	N = 67	00001140
116.	NN = 136	00001150
117.	DO 110 I = 1,16	00001160
118.	NN = N + 1	00001170
119.	NNN = NN + 1	00001180
120.	NNM = NN + 2	00001190
121.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00001200
122.	ARY2(NM) = ARY1(NNM)	00001210
123.	N = N + 2	00001220
124.	NN = NN + 3	00001230
125.	110 CONTINUE	00001240
126.	ARY2(99) = ARY1(184)	00001250
127.	ARY2(100) = ARY1(185)	00001260
128.	ARY2(101) = ARY1(186)	00001270
129.	ARY2(102) = ARY1(187)	00001280
130.	ARY2(103) = ARY1(188) + ARY1(189)	00001290
131.	ARY2(104) = ARY1(191)	00001300
132.	N = 105	00001310
133.	NN = 192	00001320
134.	DO 120 I = 1,8	00001330
135.	NN = N + 1	00001340
136.	NNN = NN + 1	00001350
137.	NNM = NN + 2	00001360
138.	ARY2(N) = ARY1(NN) + ARY1(NNM)	00001370
139.	ARY2(NM) = ARY1(NNM)	00001380
140.	N = N + 2	00001390
141.	NN = NN + 3	00001400
142.	120 CONTINUE	00001410
143.	ARY2(121) = ARY1(216)	00001420
144.	ARY2(122) = ARY1(217) + ARY1(218)	00001430
145.	ARY2(123) = ARY1(219)	00001440
146.	ARY2(124) = ARY1(220) + ARY1(221)	00001450
147.	ARY2(125) = ARY1(222)	00001460
148.	ARY2(126) = ARY1(223)	00001470
149.	N = 127	00001480
150.	NN = 224	00001490
151.	DO 130 I = 1,4	00001500
152.	NN = N + 1	00001510
153.	NNN = N + 2	00001520
154.	NNNN = N + 3	00001530
155.	NNN = NN + 1	00001540
156.	NNNN = NN + 2	00001550
157.	NNNNN = NN + 3	00001560
158.	NNNNNN = NN + 4	00001570
159.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00001580
160.	ARY2(NNM) = ARY1(NNNN)	00001590
161.	ARY2(NNM) = ARY1(NNNNN)	00001600
162.	ARY2(NNNM) = ARY1(NNNNNN)	00001610
163.	N = N + 4	00001620
164.	NN = NN + 5	00001630
165.	130 CONTINUE	00001640
166.	N = 143	00001650
167.	DO 140 I = 244,283	00001660
168.	ARY2(N) = ARY1(I)	00001670
169.	N = N + 1	00001680
170.	140 CONTINUE	00001690

171.	C		00001700
172.	C	PRINT REPORT	00001710
173.	C		00001720
174.		IF(JJ.EQ.5) GOTO 266	00001730
175.		WRITE(3,205)	00001731
176.		205 FORMAT(1X,'COUNTY')	00001732
177.		WRITE(3,200) IPSEUD,SUMPOP	00001740
178.		200 FORMAT('1',//,1X,I5,5X,'POPULATION',3X,F12.0)	00001750
179.		GOTO 202	00001760
180.		266 WRITE(3,201)IPSEUD,SUMPOP	00001770
181.		201 FORMAT('1',//,I5,'SCHOOL',5X,'NUMBER OF STUDENTS',3X,F12.0)	00001780
182.		202 WRITE(3,210)	00001790
183.		210 FORMAT('0',17X,'TOTAL',10X,'PER CAPITA')	00001800
184.		DO 150 I = 1,182	00001810
185.		PERCAP(I) = ARY2(I)/SUMPOP	00001820
186.		WRITE(3,220) I,ARY2(I),PERCAP(I)	00001830
187.		220 FORMAT(11X,I3,2X,F12.0,5X,F9.2)	00001840
188.		150 CONTINUE	00001850
189.	C	GOTO 10	00001860
190.		999 STOP	00001870
191.		DEBUG INIT(IPSEUD,IC,ICODE,II,JJ,KK,SUMPOP)	00001880
192.		END	00001890

12?

L FR FINANCE.SCHOOL.FORT

```

1. C***** 00000010
2. C PROGRAMER: JANICE SCHWAB AND KEN BURK 5/21/84 * 00000020
3. C PROGRAM: FINANCE.FORT * 00000030
4. C THIS PROGRAM WILL READ SURVEY OF GOVT.FINANCE DATA AND * 00000040
5. C AGGREGATE IT AND CALCULATE THE PER CAPITA NUMBERS. * 00000050
6. C***** 00000060
7. C 00000070
8. DOUBLE PRECISION ARY1(283),ARY2(182),PERCAP(182),FIN(283) 00000080
9. DIMENSION XNAME(8) 00000090
10. DO 300 NN = 1,182 00000100
11. ARY2(NN) = 0.0 00000110
12. ARY1(NN) = 0.0 00000115
13. PERCAP(NN) = 0.0 00000120
14. 300 CONTINUE 00000130
15. WRITE(6,320) 00000140
16. 320 FORMAT(1X,'ENTER THE 4 DIGIT COUNTY PSEUDONYM NUMBER') 00000150
17. READ(5,330) IPSEUD 00000160
18. 330 FORMAT(I4) 00000170
19. C 00000180
20. WRITE(6,325) 00000190
21. 325 FORMAT(1X,'ENTER THE 1 DIGIT FILE TYPE CODE') 00000200
22. READ(5,335) ICODE 00000210
23. 335 FORMAT(I1) 00000220
24. C 00000230
25. 10 WRITE(6,340) 00000240
26. 340 FORMAT(1X,'ENTER A 3 DIGIT COUNTY CODE OR 333 TO STOP') 00000250
27. READ(5,310) IC 00000260
28. 310 FORMAT(I3) 00000270
29. IF(IC.EQ.333) GOTO 333 00000280
30. C 00000290
31. 11 READ(2,100,END=333) II,JJ,KK,LL,FOP,NN,(FIN(I),I=1,283), 00000300
32. & (XNAME(J),J=1,8) 00000310
33. 100 FORMAT(I2,1X,I1,1X,I3,1X,I3,1X,F12.0,1X,I6,/, 00000320
34. & 35(8(F12.0,1X),/),3(F12.0,1X),8A4) 00000330
35. IF(ICODE.NE.JJ) GOTO 11 00000335
36. IF(KK.GT.IC) GOTO 12 00000340
37. IF(IC.NE.KK) GOTO 11 00000350
38. SUMFOP = SUMPOF + FOP 00000370
39. DO 155 M = 1,283 00000380
40. ARY1(M) = ARY1(M) + FIN(M) 00000390
41. 155 CONTINUE 00000400
42. p V GOTO 10 00000410
43. 12 DO 615 M = 1,37 00000420
44. BACKSPACE 2 00000430
45. 615 CONTINUE 00000440
46. GOTO 10 00000450
47. C AGGREGATE THE NUMBERS 00000460
48. C 00000470
49. 333 ARY2(1) = ARY1(1) 00000480
50. ARY2(2) = ARY1(2) 00000490
51. DO 40 I = 3,7 00000500
52. ARY2(3) = ARY2(3) + ARY1(I) 00000510
53. 40 CONTINUE 00000520
54. ARY2(4) = ARY1(8) + ARY1(9) 00000530
55. ARY2(5) = ARY1(10) + ARY1(11) 00000540
56. ARY2(6) = ARY1(12) 00000550

```

57.	ARY2(7) = ARY1(13)	00000560
58.	ARY2(8) = ARY1(14)	00000570
59.	ARY2(9) = ARY1(15)	00000580
60.	ARY2(10) = ARY1(16) + ARY1(17) + ARY1(18)	00000590
61.	ARY2(11) = ARY1(19) + ARY1(20)	00000600
62.	ARY2(12) = ARY1(21)	00000610
63.	ARY2(13) = ARY1(22)	00000620
64.	ARY2(14) = ARY1(23)	00000630
65.	ARY2(15) = ARY1(24) + ARY1(25) + ARY1(26)	00000640
66.	ARY2(16) = ARY1(27)	00000650
67.	DO 45 I = 28,36	00000660
68.	ARY2(17) = ARY2(17) + ARY1(I)	00000670
69.	45 CONTINUE	00000680
70.	DO 50 I = 37,51	00000690
71.	ARY2(18) = ARY2(18) + ARY1(I)	00000700
72.	50 CONTINUE	00000710
73.	DO 60 I = 52,55	00000720
74.	ARY2(19) = ARY2(19) + ARY1(I)	00000730
75.	60 CONTINUE	00000740
76.	ARY2(20) = ARY1(56)	00000750
77.	ARY2(21) = ARY1(57) + ARY1(58) + ARY1(59)	00000760
78.	ARY2(22) = ARY1(60)	00000770
79.	DO 70 I = 61,66	00000780
80.	ARY2(23) = ARY2(23) + ARY1(I)	00000790
81.	70 CONTINUE	00000800
82.	ARY2(24) = ARY1(67) + ARY1(68) + ARY1(69) + ARY1(70)	00000810
83.	DO 80 I = 71,76	00000820
84.	ARY2(25) = ARY2(25) + ARY1(I)	00000830
85.	80 CONTINUE	00000840
86.	ARY2(26) = ARY1(77)	00000850
87.	ARY2(27) = ARY1(78)	00000860
88.	N = 28	00000870
89.	NN = 79	00000880
90.	DO 90 I = 1,6	00000890
91.	NNN = NN + 1	00000900
92.	NN = N + 1	00000910
93.	NNN = NN + 2	00000920
94.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00000930
95.	ARY2(NN) = ARY1(NNN)	00000940
96.	N = N + 2	00000950
97.	NN = NN + 3	00000960
98.	90 CONTINUE	00000970
99.	ARY2(40) = ARY1(97)	00000980
100.	ARY2(41) = ARY1(98) + ARY1(99)	00000990
101.	ARY2(42) = ARY1(100)	00001000
102.	ARY2(43) = ARY1(101)	00001010
103.	N = 44	00001020
104.	NN = 102	00001030
105.	DO 101 I = 1,11	00001040
106.	NN = N + 1	00001050
107.	NNN = NN + 1	00001060
108.	NNN = NN + 2	00001070
109.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00001080
110.	ARY2(NN) = ARY1(NNN)	00001090
111.	N = N + 2	00001100
112.	NN = NN + 3	00001110
113.	101 CONTINUE	00001120

114.	ARY2(66) = ARY1(135)	00001130
115.	N = 67	00001140
116.	NN = 136	00001150
117.	DO 110 I = 1,16	00001160
118.	NN = N + 1	00001170
119.	NNN = NN + 1	00001180
120.	NNM = NN + 2	00001190
121.	ARY2(N) = ARY1(NN) + ARY1(NNM)	00001200
122.	ARY2(NM) = ARY1(NNM)	00001210
123.	N = N + 2	00001220
124.	NN = NN + 3	00001230
125.	110 CONTINUE	00001240
126.	ARY2(99) = ARY1(184)	00001250
127.	ARY2(100) = ARY1(185)	00001260
128.	ARY2(101) = ARY1(186)	00001270
129.	ARY2(102) = ARY1(187)	00001280
130.	ARY2(103) = ARY1(188) + ARY1(189)	00001290
131.	ARY2(104) = ARY1(191)	00001300
132.	N = 105	00001310
133.	NN = 192	00001320
134.	DO 120 I = 1,8	00001330
135.	NN = N + 1	00001340
136.	NNN = NN + 1	00001350
137.	NNM = NN + 2	00001360
138.	ARY2(N) = ARY1(NN) + ARY1(NNM)	00001370
139.	ARY2(NM) = ARY1(NNM)	00001380
140.	N = N + 2	00001390
141.	NN = NN + 3	00001400
142.	120 CONTINUE	00001410
143.	ARY2(121) = ARY1(216)	00001420
144.	ARY2(122) = ARY1(217) + ARY1(218)	00001430
145.	ARY2(123) = ARY1(219)	00001440
146.	ARY2(124) = ARY1(220) + ARY1(221)	00001450
147.	ARY2(125) = ARY1(222)	00001460
148.	ARY2(126) = ARY1(223)	00001470
149.	N = 127	00001480
150.	NN = 224	00001490
151.	DO 130 I = 1,4	00001500
152.	NN = N + 1	00001510
153.	NNN = N + 2	00001520
154.	NNM = N + 3	00001530
155.	NNN = NN + 1	00001540
156.	NNNN = NN + 2	00001550
157.	NNNNN = NN + 3	00001560
158.	NNNNNN = NN + 4	00001570
159.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00001580
160.	ARY2(NNM) = ARY1(NNNN)	00001590
161.	ARY2(NNMN) = ARY1(NNNNN)	00001600
162.	ARY2(NNNNN) = ARY1(NNNNNN)	00001610
163.	N = N + 4	00001620
164.	NN = NN + 5	00001630
165.	130 CONTINUE	00001640
166.	N = 143	00001650
167.	DO 140 I = 244,283	00001660
168.	ARY2(N) = ARY1(I)	00001670
169.	N = N + 1	00001680
170.	140 CONTINUE	00001690

171.	C		00001700
172.	C	PRINT REPORT	00001710
173.	C		00001720
174.		WRITE(3,205)	00001730
175.		205 FORMAT(1X,'SCHOOL')	00001740
176.		266 WRITE(3,201)IPSEUD,SUMPOP	00001770
177.		201 FORMAT('1',/,15,5X,'NUMBER OF STUDENTS',3X,F12.0)	00001780
178.		202 WRITE(3,210)	00001790
179.		210 FORMAT('0',17X,'TOTAL',10X,'PER CAPITA')	00001800
180.		DO 1500ID= 1,182	00001810
181.		PERCAP(I) = ARY2(I)/SUMPOP	00001820
182.		WRITE(3,220) I,ARY2(I),PERCAP(I)	00001830
183.		220 FORMAT(11X,I3,2X,F12.0,5X,F9.2)	00001840
184.		150 CONTINUE	00001850
185.	C	GOTO 10	00001860
186.		999 STOP	00001870
187.		DEBUG INIT(IPSEUD,IC,ICODE,II,JI,XK,LL,SUMPOP)	00001880
188.		END	00001890
189.			

L FR FINSTATE.FORT

```

1. C***** 00000010
2. C PROGRAMER: JANICE SCHWAB AND KEN BURK 8/21/84 * 00000020
3. C PROGRAM: FINANCE.FORT * 00000030
4. C THIS PROGRAM WILL READ SURVEY OF GOVT.FINANCE DATA AND * 00000040
5. C AGGREGATE IT AND CALCULATE THE PER CAPITA NUMBERS. * 00000050
6. C***** 00000060
7. C 00000070
8. DOUBLE PRECISION ARY1(283),ARY2(182),PERCAP(182) 00000080
9. DIMENSION XNAME(8) 00000090
10. C 00000100
11. DO 300 NN = 1,182 00000110
12. ARY2(NN) = 0.0 00000120
13. PERCAP(NN) = 0.0 00000130
14. 300 CONTINUE 00000140
15. C 00000150
16. 10 READ(2,100,END=999) II,JJ,KK,LL,POP,NN,(ARY1(I),I=1,283), 00000160
17. & (XNAME(J),J=1,8) 00000170
18. IF((II.NE.48).AND.(II.NE.32).AND.(II.NE.14).AND. 00000180
19. &(II.NE.41).AND.(II.NE.11)) GOTO 10 00000181
20. 100 FORKAT(I2,1X,I1,1X,I3,1X,I3,1X,F12.0,1X,I6, 00000210
21. & 35(8(F12.0,1X),/),3(F12.0,1X),JA4) 00000220
22. C AGGREGATE THE NUMBERS 00000230
23. C 00000240
24. ARY2(1) = ARY1(1) 00000250
25. ARY2(2) = ARY1(2) 00000260
26. DO 40 I = 3,7 00000270
27. ARY2(3) = ARY2(3) + ARY1(I) 00000280
28. 40 CONTINUE 00000290
29. ARY2(4) = ARY1(8) + ARY1(9) 00000300
30. ARY2(5) = ARY1(10) + ARY1(11) 00000310
31. ARY2(6) = ARY1(12) 00000320
32. ARY2(7) = ARY1(13) 00000330
33. ARY2(8) = ARY1(14) 00000340
34. ARY2(9) = ARY1(15) 00000350
35. ARY2(10) = ARY1(16) + ARY1(17) +ARY1(18) 00000360
36. ARY2(11) = ARY1(19) + ARY1(20) 00000370
37. ARY2(12) = ARY1(21) 00000380
38. ARY2(13) = ARY1(22) 00000390
39. ARY2(14) = ARY1(23) 00000400
40. ARY2(15) = ARY1(24) + ARY1(25) +ARY1(26) 00000410
41. ARY2(16) = ARY1(27) 00000420
42. DO 45 I = 28,36 00000430
43. ARY2(17) = ARY2(17) + ARY1(I) 00000440
44. 45 CONTINUE 00000450
45. DO 50 I = 37,51 00000460
46. ARY2(18) = ARY2(18) + ARY1(I) 00000470
47. 50 CONTINUE 00000480
48. DO 60 I = 52,55 00000490
49. ARY2(19) = ARY2(19) + ARY1(I) 00000500
50. 60 CONTINUE 00000510
51. ARY2(20) = ARY1(56) 00000520
52. ARY2(21) = ARY1(57) + ARY1(58) +ARY1(59) 00000530
53. ARY2(22) = ARY1(60) 00000540
54. DO 70 I = 61,66 00000550
55. ARY2(23) = ARY2(23) + ARY1(I) 00000560
56. 70 CONTINUE 00000570

```

57.	ARY2(24) = ARY1(67) + ARY1(68) + ARY1(69) + ARY1(70)	00000580
58.	DO 80 I = 71,76	00000590
59.	ARY2(25) = ARY2(25) + ARY1(I)	00000600
60.	80 CONTINUE	00000610
61.	ARY2(26) = ARY1(77)	00000620
62.	ARY2(27) = ARY1(78)	00000630
63.	N = 28	00000640
64.	NN = 79	00000650
65.	DO 90 I = 1,6	00000660
66.	NNN = NN + 1	00000670
67.	NN = N + 1	00000680
68.	NNN = NN + 2	00000690
69.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00000700
70.	ARY2(NN) = ARY1(NNN)	00000710
71.	N = N + 2	00000720
72.	NN = NN + 3	00000730
73.	90 CONTINUE	00000740
74.	ARY2(40) = ARY1(97)	00000750
75.	ARY2(41) = ARY1(98) + ARY1(99)	00000760
76.	ARY2(42) = ARY1(100)	00000770
77.	ARY2(43) = ARY1(101)	00000780
78.	N = 44	00000790
79.	NN = 102	00000800
80.	DO 101 I = 1,11	00000810
81.	NN = N + 1	00000820
82.	NNN = NN + 1	00000830
83.	NNN = NN + 2	00000840
84.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00000850
85.	ARY2(NN) = ARY1(NNN)	00000860
86.	N = N + 2	00000870
87.	NN = NN + 3	00000880
88.	101 CONTINUE	00000890
89.	ARY2(66) = ARY1(135)	00000900
90.	N = 67	00000910
91.	NN = 136	00000920
92.	DO 110 I = 1,16	00000930
93.	NN = N + 1	00000940
94.	NNN = NN + 1	00000950
95.	NNN = NN + 2	00000960
96.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00000970
97.	ARY2(NN) = ARY1(NNN)	00000980
98.	N = N + 2	00000990
99.	NN = NN + 3	00001000
100.	110 CONTINUE	00001010
101.	ARY2(99) = ARY1(184)	00001020
102.	ARY2(100) = ARY1(185)	00001030
103.	ARY2(101) = ARY1(186)	00001040
104.	ARY2(102) = ARY1(187)	00001050
105.	ARY2(103) = ARY1(188) + ARY1(189)	00001060
106.	ARY2(104) = ARY1(191)	00001070
107.	N = 105	00001080
108.	NN = 192	00001090
109.	DO 120 I = 1,8	00001100
110.	NN = N + 1	00001110
111.	NNN = NN + 1	00001120
112.	NNN = NN + 2	00001130
113.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00001140

114.	ARY2(NN) = ARY1(NNN)	00001150
115.	N = N + 2	00001160
116.	NN = NN + 3	00001170
118.	120 CONTINUE	00001180
119.	ARY2(122) = ARY1(217) + ARY1(218)	00001200
120.	ARY2(123) = ARY1(219)	00001210
121.	ARY2(124) = ARY1(220) + ARY1(221)	00001220
122.	ARY2(125) = ARY1(222)	00001230
123.	ARY2(126) = ARY1(223)	00001240
124.	N = 127	00001250
125.	NN = 224	00001260
126.	DO 130 I = 1,4	00001270
127.	NN = N + 1	00001280
128.	NNN = N + 2	00001290
129.	NNNN = N + 3	00001300
130.	NNN = NN + 1	00001310
131.	NNNN = NN + 2	00001320
132.	NNNNN = NN + 3	00001330
133.	NNNNNN = NN + 4	00001340
134.	ARY2(N) = ARY1(NN) + ARY1(NNN)	00001350
135.	ARY2(NNN) = ARY1(NNNN)	00001360
136.	ARY2(NNNN) = ARY1(NNNNN)	00001370
137.	ARY2(NNNNN) = ARY1(NNNNNN)	00001380
138.	N = N + 4	00001390
139.	NN = NN + 5	00001400
140.	130 CONTINUE	00001410
141.	N = 143	00001420
142.	DO 140 I = 244,283	00001430
143.	ARY2(N) = ARY1(I)	00001440
144.	N = N + 1	00001450
145.	140 CONTINUE	00001460
146.	C	00001470
147.	C PRINT REPORT	00001480
148.	C	00001490
149.	IF(JJ.EQ.5) GOTO 266	00001500
150.	WRITE(3,200) (XNAME(I),I=1,8),POP	00001510
151.	200 FORMAT('1',/,1X,8A4,5X,'POPULATION',3X,F12.0)	00001520
152.	GOTO 202	00001530
153.	266 WRITE(3,201)(XNAME(I),I=1,6),POP	00001540
154.	201 FORMAT('1',/,1X,6A4,'SCHOOL',5X,'NUMBER OF STUDENTS',3X,F12.0)	00001550
155.	202 WRITE(3,210)	00001560
156.	210 FORMAT('0',17X,'TOTAL',10X,'PER CAPITA')	00001570
157.	DO 150 I = 1,182	00001580
158.	IF(POP.EQ.0) PERCAP(I)=0.0	00001581
159.	IF(POP.NE.0) PERCAP(I) = ARY2(I)/POP	00001582
160.	WRITE(3,220) I,ARY2(I),PERCAP(I)	00001600
161.	220 FORMAT(11X,I3,2X,F12.0,5X,F9.2)	00001610
162.	150 CONTINUE	00001620
163.	GOTO 10	00001630
164.	999 STOP	00001640
165.	END	00000150
170.		

L FR FISCAL.FORT

1.	DIMENSION POP(22),PERCAP(182),ARRAY(20),REV(22),EXP(22)	00000010
2.	DOUBLE PRECISION AMT(22,182),A	00000011
3.	C	00000030
4.	C READ 56 YEARS OF POPULATION DATA	00000040
5.	C	00000050
6.	WRITE (2,210) Ph	00000051
7.	210 FORMAT (5X,'REGION 7',5X,'LOW CASE - SCHOOL')	00000052
8.	WRITE(2,211)	00000053
9.	211 FORMAT(18X,'(1984 DOLLARS)')	00000054
10.	DO 200 I=1,22	00000060
11.	20 READ(7,*,END=999) X,Y,Z,POP(I)	00000070
12.	REV(I) = 0.0	00000080
13.	EXP(I) = 0.0	00000090
14.	200 CONTINUE	00000100
15.	DO 100 M = 1,4	00000110
16.	READ(8,188) (ARRAY(J),J=1,20)	00000120
17.	100 CONTINUE	00000130
18.	188 FORMAT(20A1) U18	00000140
19.	WRITE(2,188)	00000150
20.	READ(8,188) (ARRAY(J),J=1,20)	00000160
21.	DO 5 I = 1,182	00000170
22.	READ(8,108,END=999) I,PERCAP(I)	00000180
23.	108 FORMAT(11X,I3,19X,F9.2)	00000190
24.	DO 5 J = 1,22	00000200
25.	JJ = J + 1979	00000210
26.	AMT(J,I) = PERCAP(I) * POP(J) * 1.20	00000220
27.	A = AMT(J,I)	00000230
28.	IF(A.NE.0.0) WRITE(2,106)I,JJ,A	00000240
29.	106 FORMAT(15,1X,I5,4X,F15.4)	00000250
30.	5 CONTINUE	00000260
31.	DO 15 J = 1,22	00000270
32.	JJ = J + 1979	00000280
33.	DO 16 I = 1,22	00000290
34.	REV(J) = REV(J) + AMT(J,I)	00000300
35.	16 CONTINUE	00000310
36.	DO 17 I = 23,141	00000320
37.	EXP(J) = EXP(J) + AMT(J,I)	00000330
38.	17 CONTINUE	00000340
39.	WRITE(2,110) JJ,REV(J),EXP(J)	00000350
40.	110 FORMAT(1X,I5,2X,'TOTAL REVENUE = ',F15.0,2X,'TOTAL EXPENSE = ',	00000360
41.	& F15.0)	00000361
42.	15 CONTINUE	00000370
43.	999 STOP	00000380
44.	END	00000390

L?

L FR RECAT.SUMMARY.FORT

```

1.      REAL AMT(56,182),T(56),DREV(56,8),DEXPC(56,13),D,      00000010
2.      & DEXPNC(56,13),DDEBT(56,7),DXTOT(56,13),ARRAY(20),      00000020
3.      & REV(56,8),EXPC(56,13),EXPNC(56,13),EXPTOT(56,13),DEBT(56,7),      00000030
4.      & PUR(5,8),PVEC(5,13),PVENC(5,13),PXTOT(5,13),PVD(5,7)      00000040
5.      INTEGER YEAR      00000041
6.      DO 15 I = 1,3      00000060
7.      READ(7,25) (ARRAY(J),J=1,20)      00000070
8.      25 FORMAT(20A1)      00000080
9.      15 CONTINUE      00000090
10.     10 CONTINUE      00000100
11.     DO 20 I = 1,56      00000110
12.     READ(7,*,END=999)ICAT,IYEAR,AMOUNT      00000120
13.     AMT(I,ICAT) = AMOUNT      00000130
14.     20 CONTINUE      00000140
15.     GOTO 10      00000150
16. 999 CONTINUE      00000160
17. C      00000170
18. C *** AGGREGATE REVENUE CATEGORIES ***      00000180
19. C      00000190
20.     DO 80 I = 1,56      00000200
21.     REV(I,1) = AMT(I,1)      00000210
22.     REV(I,2) = AMT(I,2)      00000220
23.     REV(I,3) = AMT(I,3) + AMT(I,4) + AMT(I,5)      00000230
24.     REV(I,4) = AMT(I,6) + AMT(I,7) + AMT(I,8) + AMT(I,9) +      00000240
25.     & AMT(I,10)      00000250
26.     REV(I,5) = AMT(I,11) + AMT(I,12) + AMT(I,13) + AMT(I,14) +      00000260
27.     & AMT(I,15)      00000270
28.     REV(I,6) = AMT(I,16) + AMT(I,17)      00000280
29.     REV(I,7) = AMT(I,18) + AMT(I,19) + AMT(I,20) + AMT(I,21) +      00000290
30.     & AMT(I,22)      00000300
31.     REV(I,8) = REV(I,1) + REV(I,2) + REV(I,3) + REV(I,4) + REV(I,5) +      00000310
32.     & REV(I,6) + REV(I,7)      00000320
33. C      00000330
34. C *** AGGREGATE EXPENSE CATEGORIES ***      00000340
35. C      00000350
36.     EXPC(I,1) = AMT(I,28) + AMT(I,32) + AMT(I,36)      00000360
37.     EXPNC(I,1) = AMT(I,26) + AMT(I,27) + AMT(I,29) +      00000370
38.     & AMT(I,30) + AMT(I,31) + AMT(I,33) +      00000380
39.     & AMT(I,34) + AMT(I,35) + AMT(I,37) +      00000390
40.     & AMT(I,38) + AMT(I,39)      00000400
41.     EXPTOT(I,1) = EXPC(I,1) + EXPNC(I,1)      00000410
42.     EXPC(I,2) = AMT(I,48) + AMT(I,95)      00000420
43.     EXPNC(I,2) = AMT(I,25) + AMT(I,47) + AMT(I,49) +      00000430
44.     & AMT(I,50) + AMT(I,94) + AMT(I,96) +      00000440
45.     & AMT(I,97)      00000450
46.     EXPTOT(I,2) = EXPC(I,2) + EXPNC(I,2)      00000460
47.     EXPC(I,3) = AMT(I,75) + AMT(I,103)      00000470
48.     EXPNC(I,3) = AMT(I,74) + AMT(I,76) + AMT(I,77) +      00000480
49.     & AMT(I,98) + AMT(I,99) + AMT(I,100) +      00000490
50.     & AMT(I,101) + AMT(I,102) + AMT(I,104)      00000500
51.     & AMT(I,105)      00000510
52.     EXPTOT(I,3) = EXPC(I,3) + EXPNC(I,3)      00000520
53.     EXPC(I,4) = AMT(I,91)      00000530
54.     EXPNC(I,4) = AMT(I,90) + AMT(I,92) + AMT(I,93)      00000540
55.     EXPTOT(I,4) = EXPC(I,4) + EXPNC(I,4)      00000550
56.     EXPC(I,5) = AMT(I,41) + AMT(I,44) + AMT(I,52) +      00000560

```

57.	&	AMT(I,54) + AMT(I,119)	00000570
58.		EXPNC(I,5) = AMT(I,40) + AMT(I,42) + AMT(I,43) +	00000580
59.	&	AMT(I,45) + AMT(I,46) + AMT(I,51) +	00000590
60.	&	AMT(I,53) + AMT(I,54) + AMT(I,55) +	00000600
61.	&	AMT(I,57) + AMT(I,58) + AMT(I,118) +	00000610
62.	&	AMT(I,120) + AMT(I,121) + AMT(I,122)	00000620
63.		EXPTOT(I,5) = EXPC(I,5) + EXPNC(I,5)	00000630
64.		EXPC(I,6) = AMT(I,60) + AMT(I,64) + AMT(I,67)	00000640
65.		EXPNC(I,6) = AMT(I,59) + AMT(I,61) +	00000650
66.	&	AMT(I,62) + AMT(I,63) + AMT(I,65) +	00000660
67.	&	AMT(I,66) + AMT(I,68) + AMT(I,69)	00000670
68.		EXPTOT(I,6) = EXPC(I,6) + EXPNC(I,6)	00000680
69.		EXPC(I,7) = AMT(I,107) + AMT(I,111)	00000690
70.		EXPNC(I,7) = AMT(I,106) + AMT(I,108) + AMT(I,109) +	00000700
71.	&	AMT(I,110) + AMT(I,112) + AMT(I,113)	00000710
72.		EXPTOT(I,7) = EXPC(I,7) + EXPNC(I,7)	00000720
73.		EXPC(I,8) = AMT(I,127) + AMT(I,131) + AMT(I,135)	00000730
74.		EXPNC(I,8) = AMT(I,126) + AMT(I,128) + AMT(I,129) +	00000740
75.	&	AMT(I,130) + AMT(I,132) + AMT(I,133) +	00000750
76.	&	AMT(I,134) + AMT(I,136) + AMT(I,137)	00000760
77.		EXPTOT(I,8) = EXPC(I,8) + EXPNC(I,8)	00000770
78.		EXPC(I,9) = AMT(I,71) + AMT(I,87) +	00000780
79.	&	AMT(I,115) + AMT(I,139)	00000790
80.		EXPNC(I,9) = AMT(I,23) + AMT(I,70) + AMT(I,72) +	00000800
81.	&	AMT(I,73) + AMT(I,86) + AMT(I,88) +	00000810
82.	&	AMT(I,89) + AMT(I,114) + AMT(I,116) +	00000820
83.	&	AMT(I,117) + AMT(I,138) + AMT(I,140) +	00000830
84.	&	AMT(I,141)	00000840
85.		EXPTOT(I,9) = EXPC(I,9) + EXPNC(I,9)	00000850
86.		EXPC(I,10) = AMT(I,79)	00000860
87.		EXPNC(I,10) = AMT(I,78) + AMT(I,80) + AMT(I,81)	00000870
88.		EXPTOT(I,10) = EXPC(I,10) + EXPNC(I,10)	00000880
89.		EXPC(I,11) = AMT(I,83)	00000890
90.		EXPNC(I,11) = AMT(I,82) + AMT(I,84) + AMT(I,85)	00000900
91.		EXPTOT(I,11) = EXPC(I,11) + EXPNC(I,11)	00000910
92.		EXPC(I,12) = AMT(I,124)	00000920
93.		EXPNC(I,12) = AMT(I,24) + AMT(I,123) + AMT(I,125)	00000930
94.		EXPTOT(I,12) = EXPC(I,12) + EXPNC(I,12)	00000950
95.	C		00000960
96.	C *** AGGREGATE DEBT CATEGORIES ***		00000970
97.	C		00000980
98.		DEBT(I,1) = AMT(I,143) + AMT(I,144) + AMT(I,145) +	00000990
99.	&	AMT(I,146) + AMT(I,147) + AMT(I,148) +	00001000
100.	&	AMT(I,149) + AMT(I,150) + AMT(I,151) +	00001010
101.	&	AMT(I,152) + AMT(I,153)	00001020
102.		DEBT(I,2) = AMT(I,154) + AMT(I,155) + AMT(I,156) +	00001030
103.	&	AMT(I,157) + AMT(I,158) + AMT(I,160) +	00001040
104.	&	AMT(I,161) + AMT(I,162) + AMT(I,163) +	00001050
105.	&	AMT(I,164)	00001060
106.		DEBT(I,3) = AMT(I,165) + AMT(I,166) + AMT(I,167) +	00001070
107.	&	AMT(I,168) + AMT(I,169) + AMT(I,170) +	00001080
108.	&	AMT(I,171) + AMT(I,172)	00001090
109.		DEBT(I,4) = AMT(I,173) + AMT(I,174) + AMT(I,175) +	00001100
110.	&	AMT(I,176) + AMT(I,177) + AMT(I,178) +	00001110
111.	&	AMT(I,179) + AMT(I,180)	00001120
112.		DEBT(I,5) = AMT(I,181)	00001130
113.		DEBT(I,6) = AMT(I,182)	00001140

2 114.	DEBT(I,7) = DEBT(I,1) + DEBT(I,2) + DEBT(I,3) + DEBT(I,4) +	00001150
2 115.	DEBT(I,5) + DEBT(I,6)	00001160
2 116.	80 CONTINUE	00001170
2 117.	C	00001180
2 118.	C	00001190
2 119.	DO 150 I=1,56	00001200
2 120.	DO 140 J=1,12	00001210
2 121.	EXPC(I,13) = EXPC(I,13) + EXPC(I,J)	00001220
2 122.	EXPNC(I,13) = EXPNC(I,13) + EXPNC(I,J)	00001230
2 123.	EXPTOT(I,13) = EXPTOT(I,13) + EXPTOT(I,J)	00001240
2 124.	140 CONTINUE	00001250
2 125.	150 CONTINUE	00001260
2 126.	C	00001270
2 127.	II=-11	00001280
2 128.	DO 160 I=1,56	00001290
2 129.	T(I) = II + I	00001300
2 130.	160 CONTINUE	00001310
2 131.	C	00001320
2 132.	R = .02	00001330
2 133.	C	00001340
2 134.	DO 170 J = 1,56	00001350
2 135.	C	00001360
2 136.	C *** CALCULATE REVENUE DISCOUNT ***	00001370
2 137.	C	00001380
2 138.	DO 180 I = 1,8	00001390
2 139.	Q = 1/(1 + R) ** T(J)	00001400
2 140.	DREV(J,I) = REV(J,I) * Q	00001410
2 141.	180 CONTINUE	00001420
2 142.	C	00001430
2 143.	C *** CALCULATE EXPENDITURE DISCOUNTS ***	00001440
2 144.	C	00001450
2 145.	DO 190 II = 1,13	00001460
2 146.	DEXPC(J,II) = EXPC(J,II) * 1/(1 + R) ** T(J)	00001470
2 147.	DEXPNC(J,II) = EXPNC(J,II) * 1/(1 + R) ** T(J)	00001480
2 148.	DEXTOT(J,II) = EXPTOT(J,II) * 1/(1 + R) ** T(J)	00001490
2 149.	190 CONTINUE	00001500
2 150.	C	00001510
2 151.	C *** CALCULATE DEBT DISCOUNTS ***	00001520
2 152.	C	00001530
2 153.	DO 200 III = 1,7	00001540
2 154.	DDEBT(J,III) = DEBT(J,III) * 1/(1 + R) ** T(J)	00001550
2 155.	200 CONTINUE	00001560
2 156.	170 CONTINUE	00001570
2 157.	C	00001580
2 158.	C *** CALCULATE REVENUE PRESENT VALUES ***	00001590
2 159.	C	00001600
2 160.	DO 501 I = 1,8	00001610
2 161.	DO 505 J = 11,17	00001620
2 162.	FVR(1,I) = FVR(1,I) + DREV(J,I)	00001630
2 163.	505 CONTINUE	00001640
2 164.	DO 502 J = 18,23	00001650
2 165.	PVR(2,I) = PVR(2,I) + DREV(J,I)	00001660
2 166.	502 CONTINUE	00001670
2 167.	DO 503 J = 24,44	00001680
2 168.	FVR(3,I) = FVR(3,I) + DREV(J,I)	00001690
2 169.	503 CONTINUE	00001700
2 170.	DO 504 J = 45,47	00001710

171.	PUR(4,I) = PUR(4,I) + DREV(J,I)	00001720
172.	504 CONTINUE	00001730
173.	501 CONTINUE	00001740
174.	C	00001750
175.	DO 506 J = 1,8	00001760
176.	PUR(5,J) = PUR(1,J) + PUR(2,J) + PUR(3,J) + PUR(4,J)	00001770
177.	506 CONTINUE	00001780
178.	C	00001790
179.	C *** CALCULATE EXPENDITURE PRESENT VALUES ***	00001800
180.	C	00001810
181.	DO 510 I = 1,13	00001820
182.	DO 514 J = 11,17	00001830
183.	PVEC(1,I) = PVEC(1,I) + DEXPC(J,I)	00001840
184.	PVENC(1,I) = PVENC(1,I) + DEXPNC(J,I)	00001850
185.	PVETOT(1,I) = PVETOT(1,I) + DEXTOT(J,I)	00001860
186.	514 CONTINUE	00001870
187.	C	00001880
188.	DO 511 J = 18,23	00001890
189.	PVEC(2,I) = PVEC(2,I) + DEXPC(J,I)	00001900
190.	PVENC(2,I) = PVENC(2,I) + DEXPNC(J,I)	00001910
191.	PVETOT(2,I) = PVETOT(2,I) + DEXTOT(J,I)	00001920
192.	511 CONTINUE	00001930
193.	C	00001940
194.	DO 512 J = 24,44	00001950
195.	PVEC(3,I) = PVEC(3,I) + DEXPC(J,I)	00001960
196.	PVENC(3,I) = PVENC(3,I) + DEXPNC(J,I)	00001970
197.	PVETOT(3,I) = PVETOT(3,I) + DEXTOT(J,I)	00001980
198.	512 CONTINUE	00001990
199.	C	00002000
200.	DO 513 J = 45,47	00002010
201.	PVEC(4,I) = PVEC(4,I) + DEXPC(J,I)	00002020
202.	PVENC(4,I) = PVENC(4,I) + DEXPNC(J,I)	00002030
203.	PVETOT(4,I) = PVETOT(4,I) + DEXTOT(J,I)	00002040
204.	513 CONTINUE	00002050
205.	C	00002060
206.	510 CONTINUE	00002070
207.	C	00002080
208.	DO 515 J = 1,13	00002090
209.	PVEC(5,J) = PVEC(1,J) + PVEC(2,J) + PVEC(3,J) + PVEC(4,J)	00002100
210.	PVENC(5,J) = PVENC(1,J) + PVENC(2,J) + PVENC(3,J) +	00002110
211.	& PVENC(4,J)	00002120
212.	PVETOT(5,J) = PVETOT(1,J) + PVETOT(2,J) + PVETOT(3,J) +	00002130
213.	& PVETOT(4,J)	00002140
214.	515 CONTINUE	00002150
215.	C	00002160
216.	C *** CALCULATE DEBT PRESENT VALUES ***	00002170
217.	C	00002180
218.	DO 520 I = 1,7	00002190
219.	DO 524 J = 11,17	00002200
220.	PVD(1,I) = PVD(1,I) + DDEBT(J,I)	00002210
221.	524 CONTINUE	00002220
222.	C	00002230
223.	DO 521 J = 18,23	00002240
224.	PVD(2,I) = PVD(2,I) + DDEBT(J,I)	00002250
225.	521 CONTINUE	00002260
226.	C	00002270
227.	DO 522 J = 24,44	00002280

228.	PVD(3,I) = PVD(3,I) + DDEBT(J,I)	00002290
229.	522 CONTINUE	00002300
230.	C	00002560
231.	DO 523 J = 45,47	00002320
232.	PVD(4,I) = PVD(4,I) + DDEBT(J,I)	00002330
233.	523 CONTINUE	00002340
234.	C	00002350
235.	520 CONTINUE	00002360
236.	C	00002370
237.	DO 525 J = 1,7	00002380
238.	PVD(5,J) = PVD(1,J) + PVD(2,J) + PVD(3,J) + PVD(4,J)	00002390
239.	525 CONTINUE	00002400
240.	C	00002410
241.	WRITE(8,800)	00002420
242.	C	00002430
243.	WRITE(8,205)	00002440
244.	205 FORMAT(44X,'STATE AND LOCAL FISCAL ITEMS, SCHOOL, 1985 \$',	00002450
245.	& /,1X,'REVENUE',52X,'ARID-202, CONCASKS',39X,'03/25/85')	00002460
246.	C	00002470
247.	WRITE(8,210)	00002480
248.	210 FORMAT(1X,7(' '),16X,'PROPERTY TAX',18X,	00002490
249.	& 'SALES TAX',19X,'MISC. TAX',12X,	00002500
250.	& 'TRANSFER PAYMENTS - FED.',//,1X,'YEAR',	00002510
251.	& 7X,'1985 \$',5X,'DISCOUNT \$',7X,'1985 \$',5X,	00002520
252.	& 'DISCOUNT \$',7X,'1985 \$',5X,'DISCOUNT \$',	00002530
253.	& 7X,'1985 \$',5X,'DISCOUNT \$',/)	00002540
254.	C	00002550
255.	DO 220 I = 1,56	00002560
256.	YEAR = 1974 + I	00002570
257.	WRITE(8,230) YEAR,REV(I,1),DREV(I,1),REV(I,2),DREV(I,2),	00002580
258.	& REV(I,3),DREV(I,3),REV(I,4),DREV(I,4)	00002590
259.	220 CONTINUE	00002600
260.	C	00002610
261.	230 FORMAT(1X,I4,4X,F11.0,2X,F11.0,4X,F11.0,2X,F11.0,4X,F11.0,2X,	00002620
262.	& F11.0,4X,F11.0,2X,F11.0)	00002630
263.	C	00002640
264.	WRITE(8,800)	00002650
265.	C	00002660
266.	WRITE(8,240)	00002670
267.	240 FORMAT(10X,'TRANSFER PAYMENTS - STATE',3X,	00002680
268.	& 'TRANSFER PAYMENTS - LOCAL',9X,	00002690
269.	& 'MISC. REVENUE',15X,'TOTAL REVENUE',//,	00002700
270.	& 1X,'YEAR',7X,'1985 \$',5X,'DISCOUNT \$',7X,	00002710
271.	& '1985 \$',5X,'DISCOUNT \$',7X,'1985 \$',5X,	00002720
272.	& 'DISCOUNT \$',7X,'1985 \$',5X,'DISCOUNT \$',/)	00002730
273.	C	00002740
274.	DO 250 I = 1,56	00002750
275.	YEAR = 1974 + I	00002760
276.	WRITE(8,260) YEAR,REV(I,5),DREV(I,5),REV(I,6),DREV(I,6),	00002770
277.	& REV(I,7),DREV(I,7),REV(I,8),DREV(I,8)	00002780
278.	250 CONTINUE	00002790
279.	C	00002800
280.	260 FORMAT(1X,I4,4X,F11.0,2X,F11.0,4X,F11.0,2X,F11.0,	00002810
281.	& 4X,F11.0,2X,F11.0,4X,F11.0,2X,F11.0)	00002820
282.	C	00002830
283.	WRITE(8,800)	00002840
284.	800 FORMAT('1')	00002850

```

285.      C                                          00002860
286.      WRITE(8,526) PUR(1,1),PUR(1,2),PUR(1,3),PUR(1,4),PUR(1,5), 00002870
287.      & PUR(1,6),PUR(1,7),PUR(1,8),PUR(2,1),PUR(2,2),PUR(2,3),PUR(2,4),00002880
288.      & PUR(2,5),PUR(2,6),PUR(2,7),PUR(2,8),PUR(3,1),PUR(3,2), 00002890
289.      & PUR(3,3),PUR(3,4),PUR(3,5),PUR(3,6),PUR(3,7),PUR(3,8), 00002900
290.      & PUR(4,1),PUR(4,2),PUR(4,3),PUR(4,4),PUR(4,5),PUR(4,6),PUR(4,7),00002910
291.      & PUR(4,8),PUR(5,1),PUR(5,2),PUR(5,3),PUR(5,4),PUR(5,5),PUR(5,6),00002920
292.      & PUR(5,7),PUR(5,8) 00002930
293.      C                                          00002940
294.      526 FORMAT(54X,'PRESENT VALUES (REVENUES)',//, 00002950
295.      & 65X,'TRANSFER',6X,'TRANSFER',6X,'TRANSFER',/, 00002960
296.      & 23X,'PROPERTY',8X,'SALES',9X,'MISC.',7X, 00002970
297.      & 'PAYMENTS',6X,'PAYMENTS',6X,'PAYMENTS',8X,'MISC.',9X, 00002980
298.      & 'TOTAL',/,25X,'TAX',12X,'TAX',11X, 00002990
299.      & 'TAX',8X,'(FEDERAL)',5X,'(STATE)',7X,'(LOCAL)',8X, 00003000
300.      & 'REVENUE',7X,'REVENUE',/,1X,'PRE-CONSTRUCTION',5X,8(F12.0,2X),00003010
301.      & /,1X,'CONSTRUCTION',9X,8(F12.0,2X),/,1X,'OPERATIONS', 00003020
302.      & 11X,8(F12.0,2X),/,1X,'DECOMMISSIONING',6X,8(F12.0,2X),/,1X, 00003030
303.      & 'TOTAL',15X,8(F13.0,1X)) 00003040
304.      C                                          00003050
305.      WRITE(8,800) 00003060
306.      C                                          00003070
307.      WRITE(8,301) 00003080
308.      301 FORMAT(1X,'EXPENDITURES',/,1X,12('-')) 00003090
309.      C                                          00003100
310.      DO 300 I = 1,13 00003110
311.      IF(I.GT.1) WRITE(8,800) 00003120
312.      IF(I.EQ.1) WRITE(8,302) 00003130
313.      IF(I.EQ.2) WRITE(8,303) 00003140
314.      IF(I.EQ.3) WRITE(8,304) 00003150
315.      IF(I.EQ.4) WRITE(8,305) 00003160
316.      IF(I.EQ.5) WRITE(8,306) 00003170
317.      IF(I.EQ.6) WRITE(8,307) 00003180
318.      IF(I.EQ.7) WRITE(8,308) 00003190
319.      IF(I.EQ.8) WRITE(8,309) 00003200
320.      IF(I.EQ.9) WRITE(8,310) 00003210
321.      IF(I.EQ.10) WRITE(8,311) 00003220
322.      IF(I.EQ.11) WRITE(8,312) 00003230
323.      IF(I.EQ.12) WRITE(8,313) 00003240
324.      IF(I.EQ.13) WRITE(8,314) 00003250
325.      WRITE(8,315) 00003260
326.      C                                          00003270
327.      DO 320 J = 1,56 00003280
328.      YEAR = 1974 + J 00003290
329.      WRITE(8,321) YEAR,EXPC(J,I),DEXPC(J,I),EXPMC(J,I), 00003300
330.      & DEXPNC(J,I),EXPTOT(J,I),DEXTOT(J,I) 00003310
331.      321 FORMAT(1X,I4,14X,F11.0,2X,F11.0,14X,F11.0,2X,F11.0,14X,F11.0, 00003320
332.      & 2X,F11.0) 00003330
333.      320 CONTINUE 00003340
334.      300 CONTINUE 00003350
335.      C                                          00003360
336.      302 FORMAT(24X,'EDUCATION (CAP.)',20X,'EDUCATION (NON-CAP.)', 00003370
337.      & 20X,'EDUCATION (TOTAL)',/) 00003380
338.      303 FORMAT(22X,'PUBLIC SAFETY (CAP.)',17X,'PUBLIC SAFETY (NON-CAP.)', 00003390
339.      & 15X,'PUBLIC SAFETY (TOTAL)',/) 00003400
340.      304 FORMAT(21X,'PUBLIC ASSISTANCE (CAP.)',12X,'PUBLIC ASSISTANCE ', 00003410
341.      & '(NON-CAP.)',11X,'PUBLIC ASSISTANCE (TOTAL)',/) 00003420

```

```

342. 305 FORMAT(20X,'PARKS & RECREATION (CAP.)',11X,'PARKS & RECREATION ', 00003430
343. & '(NON-CAP.)',11X,'PARKS & RECREATION (TOTAL)') 00003440
344. 306 FORMAT(22X,'ADMINISTRATION (CAP.)',16X,'ADMINISTRATION (NON-CAP.)' 00003450
345. & ',14X,'ADMINISTRATION (TOTAL)',/) 00003460
346. 307 FORMAT(26X,'HEALTH (CAP.)',22X,'HEALTH (NON-CAP.)',22X, 00003470
347. & 'HEALTH (TOTAL)',/) 00003480
348. 308 FORMAT(24X,'SANITATION (CAP.)',19X,'SANITATION (NON-CAP.)',18X, 00003490
349. & 'SANITATION (TOTAL)',/) 00003500
350. 309 FORMAT(24X,'UTILITIES (CAP.)',20X,'UTILITIES (NON-CAP.)',20X, 00003510
351. & 'UTILITIES (TOTAL)',/) 00003520
352. 310 FORMAT(22X,'TRANSPORTATION (CAP.)',16X,'TRANSPORTATION (NON-CAP.)', 00003530
353. & '15X,'TRANSPORTATION (TOTAL)',/) 00003540
354. 311 FORMAT(24X,'LIBRARIES (CAP.)',20X,'LIBRARIES (NON-CAP.)',20X, 00003550
355. & 'LIBRARIES (TOTAL)',/) 00003560
356. 312 FORMAT(21X,'NATURAL RESOURCES (CAP.)',12X,'NATURAL RESOURCES ', 00003561
357. & '(NON-CAP.)',11X,'NATURAL RESOURCES (TOTAL)',/) 00003562
358. 313 FORMAT(20X,'MISC. EXPENDITURES (CAP.)',11X,'MISC. EXPENDITURES ', 00003590
359. & '(NON-CAP.)',11X,'MISC. EXPENDITURES (TOTAL)',/) 00003600
360. 314 FORMAT(26X,'TOTALS (CAP.)',23X,'TOTALS (NON-CAP.)',22X,'TOTALS ', 00003610
361. & '(TOTAL)',/) 00003620
362. 315 FORMAT(1X,'YEAR',17X,'1985 $',5X,'DISCOUNT $',17X,'1985 $',5X, 00003630
363. & 'DISCOUNT $',17X,'1985 $',5X,'DISCOUNT $',/) 00003640
364. C 00003650
365. WRITE(8,800) 00003660
366. C 00003670
367. WRITE(8,530) 00003680
368. 530 FORMAT(53X,'PRESENT VALUES (EXPENDITURES)',/,22X,3('EDUCATION',9X)00003690
369. & ',2X,3('PUBLIC',12X),/,23X,'(CAP.)',10X,'(NON-CAP.)',10X, 00003700
370. & '(TOTAL)',8X,'SAFETY (CAP.)',3X,'SAFETY (NON-CAP.)',2X, 00003710
371. & 'SAFETY (TOTAL)',/) 00003720
372. WRITE(8,531) PVEC(1,1),PVENC(1,1),PVETOT(1,1),PVEC(1,2), 00003730
373. & PVENC(1,2),PVETOT(1,2),PVEC(2,1),PVENC(2,1),PVETOT(2,1), 00003740
374. & PVEC(2,2),PVENC(2,2),PVETOT(2,2),PVEC(3,1),PVENC(3,1), 00003750
375. & PVETOT(3,1),PVEC(3,2),PVENC(3,2),PVETOT(3,2),PVEC(4,1), 00003760
376. & PVENC(4,1),PVETOT(4,1),PVEC(4,2),PVENC(4,2),PVETOT(4,2), 00003770
377. & PVEC(5,1),PVENC(5,1),PVETOT(5,1),PVEC(5,2),PVENC(5,2), 00003780
378. & PVETOT(5,2) 00003790
379. C 00003800
380. 531 FORMAT(1X,'PRE-CONSTRUCTION',4X,6(F12.0,6X),/,1X,'CONSTRUCTION', 00003810
381. & 8X,6(F12.0,6X),/,1X,'OPERATIONS',10X,6(F12.0,6X),/,1X, 00003820
382. & 'DECOMMISSIONING',5X,6(F12.0,6X),/,1X,'TOTAL',15X,6(F12.0,6X), 00003830
383. & //) 00003840
384. C 00003850
385. 532 FORMAT(23X,'(CAP.)',10X,'(NON-CAP.)',10X,'(TOTAL)',11X,'(CAP.)', 00003860
386. & 10X,'(NON-CAP.)',10X,'(TOTAL)',/) 00003870
387. C 00003880
388. WRITE(8,533) 00003890
389. C 00003900
390. 533 FORMAT(23X,3('PUBLIC',12X),3('PARKS &',11X),/,21X, 00003910
391. & 3('ASSISTANCE',8X),3('RECREATION',8X)) 00003920
392. C 00003930
393. WRITE(8,532) 00003940
394. C 00003950
395. WRITE(8,531) PVEC(1,3),PVENC(1,3),PVETOT(1,3),PVEC(1,4), 00003960
396. & PVENC(1,4),PVETOT(1,4),PVEC(2,3),PVENC(2,3),PVETOT(2,3), 00003970
397. & PVEC(2,4),PVENC(2,4),PVETOT(2,4),PVEC(3,3),PVENC(3,3), 00003980
398. & PVETOT(3,3),PVEC(3,4),PVENC(3,4),PVETOT(3,4),PVEC(4,3), 00003990

```

399.	&	PVENC(4,3),PVETOT(4,3),FVEC(4,4),PVENC(4,4),PVETOT(4,4),	00004000
400.	&	FVEC(5,3),PVENC(5,3),PVETOT(5,3),FVEC(5,4),PVENC(5,4),	00004010
401.	&	PVETOT(5,4)	00004020
402.	C		00004030
403.		WRITE(8,534)	00004040
404.		534 FORMAT(20X,3('ADMINISTRATION',4X),2X,3('HEALTH',12X))	00004050
405.	C		00004060
406.		WRITE(8,532)	00004070
407.	C		00004080
408.		WRITE(8,531) FVEC(1,5),PVENC(1,5),PVETOT(1,5),FVEC(1,6),	00004090
409.	&	PVENC(1,6),PVETOT(1,6),FVEC(2,5),PVENC(2,5),PVETOT(2,5),	00004100
410.	&	FVEC(2,6),PVENC(2,6),PVETOT(2,6),FVEC(3,5),PVENC(3,5),	00004110
411.	&	PVETOT(3,5),FVEC(3,6),PVENC(3,6),PVETOT(3,6),FVEC(4,5),	00004120
412.	&	PVENC(4,5),PVETOT(4,5),FVEC(4,6),PVENC(4,6),PVETOT(4,6),	00004130
413.	&	FVEC(5,5),PVENC(5,5),PVETOT(5,5),FVEC(5,6),PVENC(5,6),	00004140
414.	&	PVETOT(5,6)	00004150
415.	C		00004160
416.		WRITE(8,535)	00004170
417.		535 FORMAT(22X,3('SANITATION',7X),2X,3('UTILITIES',9X))	00004180
418.	C		00004190
419.		WRITE(8,532)	00004200
420.	C		00004210
421.		WRITE(8,531) FVEC(1,7),PVENC(1,7),PVETOT(1,7),FVEC(1,8),	00004220
422.	&	PVENC(1,8),PVETOT(1,8),FVEC(2,7),PVENC(2,7),PVETOT(2,7),	00004230
423.	&	FVEC(2,8),PVENC(2,8),PVETOT(2,8),FVEC(3,7),PVENC(3,7),	00004240
424.	&	PVETOT(3,7),FVEC(3,8),PVENC(3,8),PVETOT(3,8),FVEC(4,7),	00004250
425.	&	PVENC(4,7),PVETOT(4,7),FVEC(4,8),PVENC(4,8),PVETOT(4,8),	00004260
426.	&	FVEC(5,7),PVENC(5,7),PVETOT(5,7),FVEC(5,8),PVENC(5,8),	00004270
427.	&	PVETOT(5,8)	00004280
428.	C		00004290
429.		WRITE(8,536)	00004300
430.		536 FORMAT(20X,3('TRANSFORMATION',4X),4X,3('LIBRARIES',9X))	00004310
431.	C		00004320
432.		WRITE(8,532)	00004330
433.	C		00004340
434.		WRITE(8,531) FVEC(1,9),PVENC(1,9),PVETOT(1,9),FVEC(1,10),	00004350
435.	&	PVENC(1,10),PVETOT(1,10),FVEC(2,9),PVENC(2,9),PVETOT(2,9),	00004360
436.	&	FVEC(2,10),PVENC(2,10),PVETOT(2,10),FVEC(3,9),PVENC(3,9),	00004370
437.	&	PVETOT(3,9),FVEC(3,10),PVENC(3,10),PVETOT(3,10),FVEC(4,9),	00004380
438.	&	PVENC(4,9),PVETOT(4,9),FVEC(4,10),PVENC(4,10),PVETOT(4,10),	00004390
439.	&	FVEC(5,9),PVENC(5,9),PVETOT(5,9),FVEC(5,10),PVENC(5,10),	00004400
440.	&	PVETOT(5,10)	00004410
441.	C		00004420
442.		WRITE(8,800)	00004430
443.	C		00004440
444.		WRITE(8,537)	00004450
445.		537 FORMAT(22X,3('NATURAL',11X),2X,2('MISC.',12X),	00004460
446.	&	2X,'MISC.',/,	00004461
447.	&	21X,3('RESOURCES',9X),3('EXPENDITURES',6X))	00004470
448.	C		00004480
449.		WRITE(8,532)	00004490
450.	C		00004500
451.		WRITE(8,531) FVEC(1,11),PVENC(1,11),PVETOT(1,11),FVEC(1,12),	00004510
452.	&	PVENC(1,12),PVETOT(1,12),FVEC(2,11),PVENC(2,11),PVETOT(2,11),	00004520
453.	&	FVEC(2,12),PVENC(2,12),PVETOT(2,12),FVEC(3,11),PVENC(3,11),	00004530
454.	&	PVETOT(3,11),FVEC(3,12),PVENC(3,12),PVETOT(3,12),FVEC(4,11),	00004540
455.	&	PVENC(4,11),PVETOT(4,11),FVEC(4,12),PVENC(4,12),PVETOT(4,12),	00004550

```

456.      & PVEC(5,11),PVENC(5,11),PVETOT(5,11),PVEC(5,12),PVENC(5,12), 00004560
457.      & PVETOT(5,12) 00004570
458. C 00004580
459.      WRITE(8,538) 00004590
460. C 00004600
461.      538 FORMAT(23X,3('TOTALS',12X)) 00004610
462. C 00004620
463.      WRITE(8,550) 00004621
464.      550 FORMAT(23X, '(CAP.)',10X, '(NON-CAP.)',10X, '(TOTAL)',/) 00004622
465. C 00004623
466.      WRITE(8,539) PVEC(1,13),PVENC(1,13),PVETOT(1,13),PVEC(2,13), 00004650
467.      & PVENC(2,13),PVETOT(2,13),PVEC(3,13),PVENC(3,13),PVETOT(3,13), 00004660
468.      & PVEC(4,13),PVENC(4,13),PVETOT(4,13),PVEC(5,13),PVENC(5,13), 00004670
469.      & PVETOT(5,13) 00004671
470. C 00004672
471.      539 FORMAT(1X, 'PRE-CONSTRUCTION',3X,3(F13.0,5X),/,1X, 00004673
472.      & 'CONSTRUCTION',7X,3(F13.0,5X),/,1X, 'OPERATIONS',9X, 00004674
473.      & 3(F13.0,5X),/,1X, 'DECOMMISSIONING',4X,3(F13.0,5X),/, 00004675
474.      & 1X, 'TOTAL',14X,3(F13.0,5X)) 00004676
475. C 00004677
476.      WRITE(8,800) 00004680
477. C 00004690
478.      WRITE(8,401) 00004700
479.      401 FORMAT(1X, 'DEBT',/,1X, '----',/,19X, 'ISSUED',22X, 'RETIRED',15X, 00004710
480.      & 'FULL FAITH & CREDIT',11X, 'NON-GUARANTEED',/) 00004720
481. C 00004730
482.      WRITE(8,402) 00004740
483.      402 FORMAT(1X, 'YEAR',7X, '1985 $',5X, 'DISCOUNT $',7X, '1985 $',5X, 00004750
484.      & 'DISCOUNT $',7X, '1985 $',5X, 'DISCOUNT $',7X, '1985 $',5X, 00004760
485.      & 'DISCOUNT $',/) 00004770
486. C 00004780
487.      DO 410 I = 1,56 00004790
488.      YEAR = 1974 + I 00004800
489.      WRITE(8,411) YEAR,DEBT(I,1),DDEBT(I,1),DEBT(I,2),DDEBT(I,2), 00004810
490.      & DEBT(I,3),DDEBT(I,3),DEBT(I,4),DDEBT(I,4) 00004820
491.      410 CONTINUE 00004830
492. C 00004840
493.      411 FORMAT(1X,14,4X,F11.0,2X,F11.0,4X,F11.0,2X,F11.0,4X,F11.0,2X,F11.0, 00004850
494.      & 4X,F11.0,2X,F11.0) 00004860
495. C 00004870
496.      WRITE(8,800) 00004880
497. C 00004890
498.      WRITE(8,412) 00004900
499.      412 FORMAT(12X, 'SHORT TERM-BEGIN. OF YR.',6X, 'SHORT TERM-END OF YR.', 00004910
500.      & 12X, 'TOTAL DEBT',/) 00004920
501. C 00004930
502.      WRITE(8,413) 00004940
503.      413 FORMAT(1X, 'YEAR',7X, '1985 $',5X, 'DISCOUNT $',7X, '1985 $',5X, 00004950
504.      & 'DISCOUNT $',7X, '1985 $',5X, 'DISCOUNT $',/) 00004960
505. C 00004970
506.      DO 420 I = 1,56 00004980
507.      YEAR = 1974 + I 00004990
508.      WRITE(8,421) YEAR,DEBT(I,5),DDEBT(I,5),DEBT(I,6),DDEBT(I,6), 00005000
509.      & DEBT(I,7),DDEBT(I,7) 00005010
510.      421 FORMAT(1X,14,4X,F11.0,2X,F11.0,4X,F11.0,2X,F11.0,4X,F11.0,2X, 00005020
511.      & F11.0) 00005021
512. C 00005030

```

513.	420 CONTINUE	00005040
514.	WRITE(8,800)	00005050
515.	C	00005060
516.	WRITE(8,540)	00005070
517.	540 FORMAT(57X,'PRESENT VALUES (DEBT)',/,50X,'FULL FAITH',7X,	00005080
518.	& 'NON-',7X,'SHORT TERM',4X,'SHORT TERM',6X,'TOTAL',/,	00005090
519.	& 24X,'ISSUED',8X,'RETIRED',6X,'& CREDIT',5X,'GUARANTEED',4X,	00005100
520.	& 'BEG. OF YR.',3X,'END OF YR.',7X,'DEBT',/,)	00005110
521.	C	00005120
522.	WRITE(8,541) PVD(1,1),PVD(1,2),PVD(1,3),PVD(1,4),PVD(1,5),PVD(1,6)	00005130
523.	& ,PVD(1,7),PVD(2,1),PVD(2,2),PVD(2,3),PVD(2,4),PVD(2,5),PVD(2,6),	00005140
524.	& PVD(2,7),PVD(3,1),PVD(3,2),PVD(3,3),PVD(3,4),PVD(3,5),PVD(3,6),	00005150
525.	& PVD(3,7),PVD(4,1),PVD(4,2),PVD(4,3),PVD(4,4),PVD(4,5),PVD(4,6),	00005160
526.	& PVD(4,7),PVD(5,1),PVD(5,2),PVD(5,3),PVD(5,4),PVD(5,5),PVD(5,6),	00005170
527.	& PVD(5,7)	00005180
528.	C	00005190
529.	541 FORMAT(1X,'PRE-CONSTRUCTION',3X,7(F12.0,2X),/,1X,'CONSTRUCTION',	00005200
530.	& 7X,7(F12.0,2X),/,1X,'OPERATIONS',9X,7(F12.0,2X),/,1X,	00005210
531.	& 'DECOMMISSIONING',4X,7(F12.0,2X),/,1X,'TOTAL',13X,7(F13.0,1X))	00005220
532.	C	00005230
533.	WRITE(8,800)	00005240
534.	C	00005250
535.	STOP	00005260
536.	END	00005270

2?

L FR KEREFS.FORT

```

1.      DIMENSION TREV(56),TEXP(56),CREV(56),CEXP(56),DREV(56),      00000010
2.      &      DEXP(56),DCASKR(56),DDRYR(56),DCASKE(56),DDRYE(56)      00000020
3.      WRITE(2,10)      00000030
4.      10 FORMAT(1X,'SUBAREA 201',39X,'STATE GOVERNMENT, ARID' ,//,      00000040
5.      &      31X,'REVENUES',48X,'EXPENDITURES' ,//,      00000050
6.      &      15X,'TREND',13X,'CASK',11X,'DRYWELL',19X,'TREND' ,      00000060
7.      &      13X,'CASK',11X,'DRYWELL' ,/)      00000070
8.      C      00000080
9.      WRITE(4,60)      00000090
10.     60 FORMAT(1X,'SUBAREA 201',30X,'STATE GOVERNMENT, ARID' ,//,      00000100
11.     &      25X,'REVENUES',33X,'EXPENDITURES' ,/,      00000110
12.     &      14X,'CASK - TREND',5X,'DRYWELL - TREND' ,10X,      00000120
13.     &      'CASK - TREND',5X,'DRYWELL - TREND' ,/)      00000121
14.     C      00000130
15.     DO 100 I = 1,56      00000140
16.     READ(3,*,END=999) KK,TREV(I),TEXP(I)      00000150
17.     40 FORMAT(1X,I5,4X,F11.0,4X,F11.0)      00000160
18.     100 CONTINUE      00000210
19.     C      00000220
20.     DO 200 I = 1,56      00000230
21.     READ(3,*,END=999) KK,CREV(I),CEXP(I)      00000240
22.     200 CONTINUE      00000270
23.     C      00000280
24.     DO 300 I = 1,56      00000290
25.     READ(3,*,END=999) KK,DREV(I),DEXP(I)      00000300
26.     300 CONTINUE      00000330
27.     C      00000340
28.     999 DO 400 I = 1,56      00000350
29.     JJ = I + 1974      00000360
30.     WRITE(2,50) JJ,TREV(I),CREV(I),DREV(I),TEXP(I),CEXP(I),DEXP(I)      00000370
31.     50 FORMAT(1X,I5,2X,F15.0,2X,F15.0,2X,F15.0,10X,F15.0,2X,F15.0,2X,      00000380
32.     &      F15.0)      00000390
33.     DCASKR(I) = CREV(I) - TREV(I)      00000400
34.     DDRYR(I) = DREV(I) - TREV(I)      00000410
35.     DCASKE(I) = CEXP(I) - TEXP(I)      00000420
36.     DDRYE(I) = DEXP(I) - TEXP(I)      00000430
37.     WRITE(4,70) JJ,DCASKR(I),DDRYR(I),DCASKE(I),DDRYE(I)      00000440
38.     70 FORMAT(1X,I5,2X,F15.0,2X,F15.0,10X,F15.0,2X,F15.0)      00000450
39.     400 CONTINUE      00000460
40.     STOP      00000470
41.     END      00000480

```

67


```

C FR STATE.FORT
C 1. C***** 00000010
C 2. C PROGRAM: STATE.FORT * 00000020
C 3. C PROGRAMMER: JANICE D. SCHWAB * 00000030
C 4. C THIS PROGRAM WILL READ THE COUNTY POPULATION FROM THE STF3C * 00000040
C 5. C TAPE FOR SELECTED STATES. * 00000050
C 6. C***** 00000060
C 7. C 00000070
C 8. DIMENSION ISUM(2),IST(2),ICN(2),TAB55(2,40),ISTATE(5), 00000080
C 9. & IST35(4),IST48(3),IST17(20),IST13(10),IST45(11), 00000090
C 10. & IFLAG(2,5) 00000100
C 11. DATA ISTATE /13,17,35,45,48/ 00000110
C 12. DATA IST35 /5,15,25,35/ 00000120
C 13. DATA IST48 /109,301,389/ 00000130
C 14. DATA IST17 /99,11,37,63,73,89,91, 00000140
C 15. & 103,105,123,113,141,143,155, 00000150
C 16. & 175,179,195,197,203,93/ 00000160
C 17. DATA IST45 /3,11,5,9,29,37,49, 00000170
C 18. & 63,65,75,81/ 00000180
C 19. DATA IST13 /33,73,107,125,163,165,131, 00000190
C 20. & 189,245,251/ 00000200
C 21. 350 READ(2,12,END=999) ISUM(1),IST(1),ICN(1),(IFLAG(1,J),J=1,5), 00000210
C 22. & (TAB55(1,JJ),JJ=1,40), 00000220
C 23. & ISUM(2),IST(2),ICN(2),(IFLAG(2,J),J=1,5), 00000230
C 24. & (TAB55(2,JJ),JJ=1,40) 00000240
C 25. 12 FORMAT(9X,I2,22X,I2,4X,I3,162X,5I1,18(100X),7X, 00000250
C 26. & 20(100X),16X,5(100X),79X,40F9.0,10(100X),77X, 00000260
C 27. & 60(100X),48X, 00000270
C 28. & 9X,I2,22X,I2,4X,I3,162X,5I1,18(100X),7X, 00000280
C 29. & 20(100X),16X,5(100X),79X,40F9.0) 00000290
C 30. DO 100 I = 1,2 00000300
C 31. C***** 00000310
C 32. C CHECK THE SUMMARY LEVEL * 00000320
C 33. C***** 00000330
C 34. N = ISUM(I) 00000340
C 35. IF(N.NE.11) GO TO 100 00000350
C 36. C***** 00000360
C 37. C CHECK FOR THE RIGHT STATE * 00000370
C 38. C***** 00000380
C 39. K = IST(I) 00000390
C 40. DO 110 L = 1,5 00000400
C 41. J = ISTATE(L) 00000410
C 42. IF(K.EQ.J) GO TO 360 00000420
C 43. 110 CONTINUE 00000430
C 44. GO TO 100 00000440
C 45. C***** 00000450
C 46. C CHECK FOR THE RIGHT COUNTY * 00000460
C 47. C***** 00000470
C 48. 360 ICC=ICN(I) 00000480
C 49. GOTO (200,210,220,230,240),L 00000490
C 50. 200 DO 201 N = 1,10 00000500
C 51. IC=IST13(N) 00000510
C 52. IF(IC.EQ.ICC) GO TO 600 00000520
C 53. 201 CONTINUE 00000530
C 54. GO TO 100 00000540
C 55. 210 DO 211 N = 1,20 00000550
C 56. IC=IST17(N) 00000560

```

57.		IF(IC.EQ.ICC) GO TO 600	00000570
58.	211	CONTINUE	00000580
59.		GO TO 100	00000590
60.	220	DO 221 N = 1,4	00000600
61.		IC=IST35(N)	00000610
62.		IF(IC.EQ.ICC) GO TO 600	00000620
63.	221	CONTINUE	00000630
64.		GO TO 100	00000640
65.	230	DO 231 N = 1,11	00000650
66.		IC=IST45(N)	00000660
67.		IF(IC.EQ.ICC) GO TO 600	00000670
68.	231	CONTINUE	00000680
69.		GO TO 100	00000690
70.	240	DO 241 N = 1,3	00000700
71.		IC=IST48(N)	00000710
72.		IF(IC.EQ.ICC) GO TO 600	00000720
73.	241	CONTINUE	00000730
74.		GO TO 100	00000740
75.		C*****	00000750
76.	C	WRITE THE COUNTY DATA *	00000760
77.		C*****	00000770
78.	600	WRITE(3,15) IST(I),ICN(I),(IFLAG(I,K),K=1,5),	00000780
79.		& (TAB55(I,J),J=1,40)	00000790
80.	15	FORMAT(1X,I2,I3,5(1X,I1),/9(1X,4F9.0,/)1X,4F9.0)	00000800
81.	100	CONTINUE	00000810
82.		GO TO 350	00000811
83.	999	STOP	00000820
84.		END	00000830
8?			

DISTRIBUTION

<u>No. of</u> <u>Copies</u>		<u>No of</u> <u>Copies</u>	
<u>OFFSITE</u>		<u>ONSITE</u>	
30	DOE Technical Information Center	3	<u>DOE Richland Operations Office</u>
			M. Dayani
4	James A. Carlson Office of Civilian Radioactive Waste Management Department of Energy 1000 Independence Avenue Washington, D.C. 20585		R. D. Izatt
			J. J. Sutey
		32	<u>Pacific Northwest Laboratory</u>
2	Peter Gross Oak Ridge Operations Office P.O. Box E Oak Ridge, Tennessee 37831		M. J. Scott (10)
			R. J. Moe
			R. E. Heineman, Sr.
			D. J. Silveira
			G. H. Sewart
			S. L. Beck
			Economics Library (10)
			Technical Information (5)
			Publishing Coordination (2)