

SYSTEM OPERATION MANUAL
to the
BALL STATE UNIVERSITY POLICE DEPARTMENT'S
RECORDS OFFICE SYSTEM

An Honors Thesis (CS 499)

by

Rhonda L. Graft

Rahul Dhesi

Dr. Rahul Dhesi

Ball State University

Muncie, Indiana

May 1986

Spring 1986

TABLE OF CONTENTS

	Page
HARDWARE INFORMATION	1
SOFTWARE INSTALLATION.	2
SYSTEM RECOVERY.	5
SYSTEM SHUTDOWN.	5
SYSTEM BACKUP.	6
TESTING.	7
SOURCE CODE.	9
FILE STRUCTURES.	100
STRUCTURE CHARTS	110
INDEX.	121

SYSTEM OPERATION MANUAL

This manual allows the user to recognize and respond properly to system errors and failures, both in the hardware and in the software. The software system for the Ball State University Police Department represents the functions of the radio room and the records office. The programs in the software system are written in dBASE III PLUS. These programs are designed to run on an IBM PC-XT, or other compatible machine, using dBASE III PLUS.

Specifically, the hardware referenced above consists of the following: an IBM PC-XT which has 640K core memory, 10M hard drive (referenced as drive C), and one floppy disk drive (referenced as drive A), the IBM PC keyboard, an IBM color display screen, and an IBM compatible printer.

The user should note some of the special keys on the keyboard.

1. Function keys <F1> through <F10>: located down the left side of the keyboard.
2. Caps Lock <Caps Lock>: located below the right hand shift key.
3. Arrow keys: located on the number pad on right side of the keyboard. Number lock <Num Lock> must be in the off position for the arrow keys to function.
4. Control key <Ctrl>: located above the left hand shift key and below the tab key.
5. Escape key <Esc>: located on the left of the top row.

Instructions for these keys can be found in the User Manual.

Switches for each component are located in various places. The on/off switch for the XT drive is located on the right side towards the back. Printer switches are usually found on the side or back. Since a color monitor is being used, this will need to be manually turned on also. The top knob on the front is for this purpose. The other two knobs are for the intensity and the brightness of the screen. These can be adjusted to suit the user.

SOFTWARE INSTALLATION

All of the following steps should be followed when the system is installed on a hard drive for the first time or in the event that all files on the disk are lost.

1. Copy DOS to the hard drive.
2. Create the necessary subdirectory. From the DOS prompt 'drive>' ensure access to drive C by typing 'C:' <return>. The DOS prompt should now be 'C>'. To create a new subdirectory type 'MD POLICE' <return>; enter the new subdirectory by 'cd police' <return>.
3. Install dBASE III PLUS to the subdirectory according to the directions specified in the dBASE III PLUS manuals.
4. Copy all program and database files to the subdirectory. This is accomplished by placing the floppy disk containing all program files into the floppy drive (A). Return the DOS prompt to 'A>' by typing 'A:' <return>. To copy, type 'COPY *.* C:' <return>. The files shown in figure 1 should now be in the subdirectory police. To check, type 'C:' <return>, 'DIR' <return>. Return to drive A, 'A:' <return>. Now, place the floppy containing the database files into the floppy drive (A). To copy, type 'COPY *.* C:' <return>. The files from figure 2 should now be in the subdirectory also. To check, go back to the C drive and type the directory again.

```
>>dir c:\police\*.prg
```

V_LIC_ST.PRG	CHECKCOM.PRG	V_LIC_YR.PRG	INCIDENT.PRG
CASE3.PRG	V_LIC_NO.PRG	FIND_STA.PRG	PRINTMEN.PRG
PRINTSUP.PRG	N_VICTIM.PRG	CRNO.PRG	ONE_INC.PRG
CASE4.PRG	CHECKCAS.PRG	CASE2.PRG	CASERPT.PRG
EDITCASE.PRG	EDITINC.PRG	PRINTCOM.PRG	N_WIT.PRG
PRINTINC.PRG	PCASE3.PRG	PRINTCAS.PRG	PCASE2.PRG
SEARCH.PRG	CHECKSUP.PRG	MANIPULA.PRG	INC2.PRG
TALLIES.PRG	N_OFFEND.PRG	V_BODY.PRG	N_RUN.PRG
N_SUSPEC.PRG	N_MP.PRG	N_APPLIC.PRG	MAIN.PRG
ENTRY.PRG	COMPLAIN.PRG	CHECKINC.PRG	SUPPLEMN.PRG
EDITMENU.PRG	EDITREC.PRG	EDITCOM.PRG	EDITSUP.PRG
PRINTREC.PRG	SINGLE_T.PRG	CASEDOC.PRG	VIEW_T.PRG
DAILY_SU.PRG	N_COMP.PRG	NAME.PRG	CRIME.PRG
STATUS.PRG	ALL_CASE.PRG	ALL_INC.PRG	ONE_CASE.PRG
VEHICLE.PRG	V_MAKE.PRG	V_COLOR.PRG	V_YEAR.PRG
V_MODEL.PRG	TYPE_ID.PRG	INV_SEAR.PRG	RANGE.PRG
PCASE4.PRG			

```
130315 bytes in    65 files.
11517952 bytes remaining on drive.
```

```
>>dir c:\police\*.fmt
```

SUMMARIZ.FMT	DETAILS.FMT	NARRATIV.FMT
--------------	-------------	--------------

```
811 bytes in    3 files.
11517952 bytes remaining on drive.
```

```
>>dir c:\police\*.dbf
MASTER.DBF          INCIDENT.DBF        DAILY_RE.DBF      CASESTAT.DBF
CASE.DBF            T_MONTH.DBF        COMPLAIN.DBF      SUPPLEME.DBF
```

```
    31273 bytes in      8 files.
11612160 bytes remaining on drive.
```

```
>>dir c:\police\*.dbt
COMPLAIN.DBT        DAILY_RE.DBT      INCIDENT.DBT       CASE.DBT
SUPPLEME.DBT
```

```
    24462 bytes in      5 files.
11612160 bytes remaining on drive.
```

```
>>dir c:\police\*.ndx
T_INDEX.NDX
```

```
    3072 bytes in      1 files.
11612160 bytes remaining on drive.
```

```
>>dir c:\police\*.tbk
COMPLAIN.TBK        SUPPLEME.TBK      INCIDENT.TBK       CASE.TBK
```

```
    12666 bytes in      4 files.
11612160 bytes remaining on drive.
```

```
>>set print off
```

Figure 2

If any of these necessary files are not copied, check the diskette by returning to drive A and typing 'DIR'. If the file is listed try 'COPY filename.ext C:'. If this still doesn't work or the file is not listed, check the appropriate backup diskette and follow the same directions. If once again the file does not exist the user will have to retype the source code, found later in this manual, using the dBASE III PLUS manuals for directions.

SYSTEM RECOVERY

If the hardware system fails or crashes, for any reason, in a way that violates file integrity, the appropriate steps will have to be taken to recover data current as of the time of the most recent backup. Compare the directory listings of the last section to the directory. Use the backup diskettes to reinstate any missing programs or database files. The user may need to reenter any unbacked entries.

DOS and dBASE III PLUS file existence should be checked also. Any missing files will need to be reinstated. As a final note, if any problems still exist, the user may need to consult University Computing Services for advice or assistance.

SYSTEM SHUTDOWN

The records office system will be shut down daily. To ensure proper saving of data, wait until the current

program has completed running. Exit the system by selecting the correct menu response. The user will want to exit completely to DOS. From DOS, initiate the backup procedures discussed in the next section. After backup is complete, turn off all portions of the machine-- screen, printer, and disk drive.

SYSTEM BACKUP

The term backup refers to the process of copying the contents of an entity. This entity can be defined as a data file, a program file, floppy disk (diskette), or hard disk. As a safety precaution, the system designers recommend that the Police Department BACKUP ALL DATA FILES ON A DAILY BASIS. This is done after exiting the system and obtaining the DOS prompt.

Two copies of all program (PGM), data structure, format (FMT), and index (NDX) files should be kept on floppy disks at all times. Backup of data files (DBF) will be done differently. Two sets of floppy disks should be used, referenced as day1 and day2. Rotate the backup onto the floppies so that two consecutive days are always backed.

Before the floppy disks can be used for backup, they must be formatted. Consult the DOS manual for the correct procedure. Be sure that you are formatting the floppy disk, not the hard drive.

Exit both the software system and dBASE III PLUS to DOS. The DOS prompt should be 'C>'. If not, type

'C:' <return>. Insert the diskette containing the oldest date into the drive (A). Type 'COPY *.DBF A:' <return>. If the files are too large and a file creation error occurs, copy the individual data files to separate diskettes. The necessary command, again at the DOS prompt 'C>', is 'COPY name.DBF a:'.

TESTING

Testing a new software system is an important stage of implementation; therefore, it will be necessary for the Police Department to continue the manual system while simultaneously running the new software system for the same data. This should be done for a minimum of six months. During this time, stored data and search results should be compared to the manual system equivalent. If after this period of time everything is corresponding between the software and manual systems, it can be assumed the software system is functioning correctly and reliably, and the software can safely take over all operations.

If discrepancies arise between data thought to be stored and data software says is stored, one possible solution may be to enter edit and check for any lower case data entries. Change the lower case data to upper case. Thus, the edit mode can be used to alter any entered data that does not coincide with correct data.

The rest of this manual contains source code, format files, file structures, and the structure chart. For each

program the name, number in relation to the structure chart, and a description of the function of the program are given. In the event source code needs to be reentered into the system, the respective code in this section should be reproduced exactly.

The file structures will be beneficial if searches other than those provided by the system are desired. From these listings, the user can determine the proper field names to be used in special searches. These searches are discussed briefly in the User Manual.

SOURCE CODE

```

*MAIN.PRG
*~~~~~*
* PROGRAM NAME:  MAIN.PRG                NUMBER:  1.0  *
*
* CALLING PROGRAM(S):  NONE                *
*   PARAMETER(S):  NONE                    *
* PROGRAM(S) CALLED:  ENTRY, MANIPULA    *
*   PARAMETER(S):  NONE                    *
*
* PURPOSE:  This module is the main menu for the *
*   police department's record office system.  *
*-----*

```

```

set color to /br, /bg,w
clear
set talk off
set intensity off

```

```

store 0 to choice
do while choice < 3
  clear
  @5,15 say "BALL STATE POLICE RECORDS OFFICE"
  @7,15 say "Choose a selection from below."
  @10,15 say "1.  Enter new records"
  @12,15 say "2.  Use current records"
  @14,15 say "3.  Exit the system"
  @17,15 say "Enter your selection:" get choice picture "@z"
  read
  @17,15 clear
  if choice >3 .or. choice <1
    @17,15 say "Please review the options shown above,"
    @18,15 say "and this time select one of them!"
    store 0 to choice
    wait
    @17,0 clear
  else
    @17,15 say "Thank you."
    do case
      case choice = 1
        do entry
      case choice = 2
        do manipulate
      case choice = 3
        exit
    endcase choice
    store 0 to choice
  endif
enddo

```

```

* ENTRY.PRG
*-----*
* PROGRAM NAME:  ENTRY                      NUMBER:  2.0  *
*
* CALLING PROGRAM(S):  MAIN                  *
*   PARAMETER(S):  NONE                      *
* PROGRAM(S) CALLED:  CHECKCOM, CHECKINC, CHECKCAS, *
*                   CHECKSUP                *
*   PARAMETER(S):  NONE                      *
*
* PURPOSE:  This module allows the user to select *
*   the type of records to enter into the system. *
*-----*

```

```

clear
store 0 to choice

do while choice < 5
  clear
  @3,15 say "ENTER NEW RECORDS"
  @5,15 say "Choose a selection from below."
  @8,15 say "1.  Enter a complaint card."
  @10,15 say "2.  Enter an incident report."
  @12,15 say "3.  Enter a case report."
  @14,15 say "4.  Enter a supplemental report."
  @16,15 say "5.  Exit to main menu."
  @19,15 say "Enter your selection:" get choice picture "@z"
  read
  @19,15 clear
  if choice > 5 .or. choice < 1
    @19,15 say "Please review the options shown above,"
    @20,15 say "and this time select one of them!"
    store 0 to choice
    wait
    clear
  else
    do case
      case choice = 1
        do checkcom
      case choice = 2
        do checkinc
      case choice = 3
        do checkcas
      case choice = 4
        do checksup
      case choice = 5
        return
    endcase choice
    store 0 to choice
  endif
enddo

```

```

* MANIPULA.PRG
* ~~~~~*
* PROGRAM NAME:  MANIPULA                NUMBER:  2.1  *
*
* CALLING PROGRAM(S):  ENTRY              *
*   PARAMETER(S):  NONE                  *
* PROGRAM(S) CALLED:  EDITMENU,PRINTMENU,TALLIES, *
*                   SEARCH                *
*   PARAMETER(S):  NONE                  *
*
* PURPOSE:  This module gives the user the options *
*   of editing,printing,searching, or tallying  *
*   records.                                   *
* ~~~~~*

```

```

clear
store 0 to choice
do while choice < 5
  clear
  @3,15 say "USE CURRENT RECORDS"
  @6,15 say "1.  Edit current records."
  @8,15 say "2.  Print current record."
  @10,15 say "3.  Calculate tallies/prepare daily report."
  @12,15 say "4.  Search current records."
  @14,15 say "5.  Exit to main menu."
  @17,15 say "Enter your selection:" get choice picture "@z"
  read
  @17,15 clear
  if choice > 5 .or. choice < 1
    @17,15 say "Please review the options shown above,"
    @18,15 say "and this time select one of them!"
    store 0 to choice
    wait
    clear
  else
    do case
      case choice = 1
        do editmenu
      case choice = 2
        do printmenu
      case choice = 3
        do tallies
      case choice = 4
        do search
      case choice = 5
        return
    endcase choice
    store 0 to choice
  endif
enddo

```

```

* CHECKCOM.PRG
*-----*
* PROGRAM NAME:  CHECKCOM          NUMBER:  3.0  *
*
* CALLING PROGRAM(S):  ENTRY      *
*   PARAMETER(S):  NONE          *
* PROGRAM(S) CALLED:  COMPLAIN    *
*   PARAMETER(S):  TCRNUM        *
*
* PURPOSE:  This module prompts the user for a case *
* report number and verifies that a complaint card *
* has not already been filed with that number.    *
* The number is passed for inclusion in the       *
* complaint card information.  Valid case report  *
* numbers take the form:  0186-111, where        *
* "01" is the month of January, "86" is the     *
* current year, and "111" is the sequential number *
* for complaints filed that month.              *
*-----*

```

```

store .f. to valid
do while .not. valid
  store "00000000" to tcrnum
  clear
  @5,15 say "ENTER A CASE REPORT NUMBER" get tcrnum
  read
  use complain
  locate for crnum = tcrnum
  if crnum = tcrnum
    store " " to response
    @7,15 say "This complaint has already been filed."
    @9,15 say "If you wish to return to the menu,"
    @10,15 say "type 'no'.  If you wish to try another"
    @11,15 say "number, type `yes`." get response;
    picture "!!!"
    read
    if response = "NO"
      clear all
      return
    endif
  else
    close database
    do complain with tcrnum
    return
  endif
enddo

```

```

* CHECKINC.PRG
*-----*
* PROGRAM NAME:  CHECKINC                NUMBER:  3.1  *
*
* CALLING PROGRAM(S):  ENTRY              *
*   PARAMETER(S):  NONE                  *
* PROGRAM(S) CALLED:  INCIDENT, INC2     *
*   PARAMETER(S):  TCRNUM                *
*
* PURPOSE:  This module allows the user to enter a *
*   case report number, checking to be sure that no *
*   previous incident report has been filed with *
*   that same number. Also, if no complaint card of *
*   that number has been filed, the user is notified *
*   to remind him to enter the information later. *
*   The case report number is passed to be stored in *
*   the incident report information. *
*-----*

```

```

store .f. to valid
do while .not. valid
  store "00000000" to tcrnum
  clear
  @5,15 say "ENTER AN INCIDENT REPORT NUMBER" get tcrnum
  read
  use incident
  locate for crnum = tcrnum
  if crnum = tcrnum
    store " " to response
    @7,15 say "This report has already been filed."
    @9,15 say "If you wish to return to the menu,"
    @10,15 say "type 'no'. If you wish to try another"
    @11,15 say "number, type 'yes'." get response picture ;
    "!!!"
    read
    if response = "NO"
      clear all
      return
    endif
  else
    close database
    use complain
    locate for crnum = tcrnum
    if crnum <> tcrnum
      @7,15 say "This report has NO corresponding complaint"
      @8,15 say "card. Please be sure to enter one in "
      @9,15 say "order to have a complete report."
      wait
    endif
    close database
    do incident with tcrnum
    do inc2 with tcrnum
    return
  endif
enddo

return

```

```

* CHECKCAS.PRG
*-----*
* PROGRAM NAME:  CHECKCAS                NUMBER:  3.2  *
*
* CALLING PROGRAM(S):  ENTRY              *
*   PARAMETER(S):     NONE                *
* PROGRAM(S) CALLED:  CASERPT, CASE2, CASE3, CASE4  *
*   PARAMETER(S):     TCRNUM              *
*
* PURPOSE:  This module allows the user to enter a  *
*   case report number, verifying that a case report *
*   of this number has not already been entered.    *
*   Also, if no complaint of this number has been  *
*   entered, the user is notified to enter the     *
*   complaint later.                               *
*-----*

store " " to tcrnum
store .f. to valid
do while .not. valid
  store "00000000" to tcrnum
  clear
  @5,15 say "ENTER A CASE REPORT NUMBER" get tcrnum
  read
  use case
  locate for crnum = tcrnum
  if crnum = tcrnum
    store " " to response
    @7,15 say "This report has already been filed."
    @9,15 say "If you wish to return to the menu,"
    @10,15 say "type 'no'. If you wish to try another"
    @11,15 say "number, type `yes`." get response;
    picture "!!!"
    read
    if response = "NO"
      clear all
      return
    endif
  else
    close database
    use complain
    locate for crnum = tcrnum
    if crnum <> tcrnum
      @7,15 say "This report has NO corresponding ";
      +"complaint"
      @8,15 say "card. Please be sure to enter one in "
      @9,15 say "order to have a complete report."
      wait
    endif
    close database
    do caserpt with tcrnum
    do case2
    do case3
    do case4
    return
  endif
enddo

return

```

```

* CHECKSUP.PRG
*-----*
* PROGRAM NAME:  CHECKSUP          NUMBER:  3.3  *
*
* CALLING PROGRAM(S):  ENTRY      *
*   PARAMETER(S):  NONE          *
* PROGRAM(S) CALLED:  SUPPLEMN    *
*   PARAMETER(S):  TCRNUM        *
*
* PURPOSE:  This module allows the user to enter a *
*   case report number, verifying that either a case *
*   or incident report of the same number has *
*   already been entered. *
*-----*

```

```

store " " to tcrnum
store " " to response
store .f. to valid
store .f. to go_ahead
do while .not. valid
  store "00000000" to tcrnum
  clear
  @5,15 say "ENTER A REPORT NUMBER" get tcrnum
  read
  use case
  locate for crnum = tcrnum
  if crnum = tcrnum
    store .t. to go_ahead
  endif
  use incident
  locate for crnum = tcrnum
  if crnum = tcrnum
    store .t. to go_ahead
  endif
  if .not. go_ahead
    @7,15 say "No case or incident report has been filed."
    @9,15 say "If you wish to return to the menu,"
    @10,15 say "type 'no'. If you wish to try another"
    @11,15 say "number, type `yes`." get response;
    picture "!!!"
    read
    if response = "NO"
      clear all
      return
    endif
  else
    close database
    do supplemn with tcrnum
    return
  endif
enddo

return

```

```

* EDITMENU.PRG
*-----*
* PROGRAM NAME:  EDITMENU          NUMBER:  3.4  *
*
* CALLING PROGRAM(S):  MANIPULA      *
*   PARAMETER(S):  NONE              *
* PROGRAM(S) CALLED:  EDITREC       *
*   PARAMETER(S):  NONE              *
*
* PURPOSE:  This module gives the user the option of *
*   which type of report to edit.   *
*-----*

```

```

clear
store 0 to choice
do while choice < 5
  clear
  @3,15 say "EDIT CURRENT RECORDS"
  @6,15 say "1.  Edit a complaint card."
  @8,15 say "2.  Edit an incident report."
  @10,15 say "3.  Edit a case report."
  @12,15 say "4.  Edit a supplemental report."
  @14,15 say "5.  Exit to previous menu."
  @17,15 say "Enter your selection:" get choice picture "@z"
  read
  @17,15 clear
  if choice > 5 .or. choice < 1
    @17,15 say "Please review the options shown above,"
    @18,15 say "and this time select one of them!"
    store 0 to choice
    wait
    clear
  else
    do case
      case choice = 1
        store "complain" to varbfile
      case choice = 2
        store "incident" to varbfile
      case choice = 3
        store "case" to varbfile
      case choice = 4
        store "suppleme" to varbfile
      case choice = 5
        return
    endcase choice
    do editrec
    store 0 to choice
  endif
enddo

```

```

* PRINTMEN.PRG
*-----*
* PROGRAM NAME:  PRINTMENU          NUMBER:  3.5  *
*
* CALLING PROGRAM(S):  MANIPULA      *
*   PARAMETER(S):  NONE              *
* PROGRAM(S) CALLED:  PRINTREC      *
*   PARAMETER(S):  NONE              *
*
* PURPOSE:  This module allows the user to choose *
*   the type of report he wants printed.  After  *
*   the user types a valid selection, a variable is *
*   assigned to represent the database that will be *
*   needed to find the report.          *
*-----*

```

```

clear
store 0 to choice
do while choice < 5
  clear
  @3,15 say "PRINT CURRENT RECORDS"
  @6,15 say "1.  Print a complaint card."
  @8,15 say "2.  Print an incident report."
  @10,15 say "3.  Print a case report."
  @12,15 say "4.  Print a supplemental report."
  @14,15 say "5.  Exit to previous menu."
  @17,15 say "Enter your selection:" get choice picture "@z"
  read
  @17,15 clear
  if choice > 5 .or. choice < 1
    @17,15 say "Please review the options shown above,"
    @18,15 say "and this time select one of them!"
    store 0 to choice
    wait
    clear
  else
    do case
      case choice = 1
        store "complain" to varbfile
      case choice = 2
        store "incident" to varbfile
      case choice = 3
        store "case" to varbfile
      case choice = 4
        store "suppleme" to varbfile
      case choice = 5
        return
    endcase choice
    do printrec
    store 0 to choice
  endif
enddo

```

```

* TALLIES.PRG
*-----*
* PROGRAM NAME:  TALLIES                NUMBER:  3.6  *
*
* CALLING PROGRAM(S):  MANIPULA          *
*   PARAMETER(S):  NONE                  *
* PROGRAM(S) CALLED:  SINGLE_T, VIEW_T, DAILY_SU    *
*   PARAMETER(S):  NONE                  *
*
* PURPOSE:  This module allows the user to work with *
*   entered records by selecting to either tally,   *
*   see tallies, or make the daily report.         *
*-----*

clear
set talk off
set intensity off

store 0 to choice
do while choice < 5
  @3,15 say "CALCULATE TALLIES/PREPARE DAILY SUMMARY"
  @6,15 say "1. Calculate a single day's tallies"
  @8,15 say "2. View tallies over range of months"
  @10,15 say "3. Prepare Daily Summary"
  @12,15 say "4. Exit to previous menu"
  @17,15 say "Enter your selection:  " get choice picture "@z"
  read
  @17,15 clear
  if choice > 5 .or. choice < 1
    @17,15 say "Please review the options shown above,"
    @18,15 say "and this time select one of them!"
    store 0 to choice
    wait
    clear
  else
    do case
      case choice = 1
        do single_t
      case choice = 2
        do view_t
      case choice = 3
        do daily_sum
      case choice = 4
        clear
        return
    endcase
    store 0 to choice
  endif
enddo

```

```

* SEARCH.PRG
*-----*
* PROGRAM NAME:  SEARCH                NUMBER:  3.7  *
*
* CALLING PROGRAM(S):  MANIPULA        *
*   PARAMETER(S):  NONE                *
* PROGRAM(S) CALLED:  NAME, CRNO, CRIME, STATUS,   *
*   VEHICLE, TYPE_ID, RANGE          *
*   PARAMETER(S):  NONE                *
*
* PURPOSE:  This module allows the user to choose *
*   information for which to search for reports.  *
*   The most widely-used searches are presented  *
*   here; other searches can be done directly in *
*   interactive mode with dBASE III PLUS.        *
*-----*

```

```

clear
store 0 to choice
do while choice < 8
  @3,15 say "FIND DESIRED RECORDS"
  @6,15 say "Search on one of the following options. . ."
  @8,15 say "1. Name"
  @9,15 say "2. Report number"
  @10,15 say "3. Type of crime/incident"
  @11,15 say "4. Status of investigation"
  @12,15 say "5. Vehicle information"
  @13,15 say "6. Type of personal involvement"
  @14,15 say "7. Range of dates"
  @15,15 say "8. Return to previous menu"
  @18,15 say "Enter your selection:" get choice picture "@z"
  read
  @18,15 clear
  if choice > 8 .or. choice < 1
    @18,15 say "Please review the options shown above,"
    @19,15 say "and this time select one of them!"
    store 0 to choice
    wait
    @18,0 clear
  else
    do case
      case choice = 1
        do name
      case choice = 2
        do crno
      case choice = 3
        do crime
      case choice = 4
        do status
      case choice = 5
        do vehicle
      case choice = 6
        do type_id
      case choice = 7
        do range
      case choice = 8
        return
    endcase
    store 0 to choice
  endif
enddo

```

```

* COMPLAIN.PRG
*-----*
* PROGRAM NAME:  COMPLAIN          NUMBER:  4.0  *
*
* CALLING PROGRAM(S):  CHECKCOM      *
*     PARAMETER(S):  TCRNUM          *
* PROGRAM(S) CALLED:  NONE          *
*     PARAMETER(S):  NONE          *
*
* PURPOSE:  This module allows for the entry of a
* complaint card.  Tcrnum is used to set the case
* report number (CRNUM).  The complainant's name
* and address are stored in the MASTER database.
* All other information entered is stored in the
* COMPLAIN database.
*-----*

```

```

PARAMETERS tcrnum
set intensity on
clear
select 2
use master
append blank
select 3
use complain
append blank

```

```

@3,5 say "ENTER COMPLAINT CARD FOLLOWING . . ."
@5,5 say "COMPLAINT:  " GET COMPLAINT PICTURE ;
      "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
REPLACE CRNUM WITH TCRNUM
@6,5 say "C R NUMBER:  "+CRNUM
@7,5 SAY "REC'D BY:  " GET PE PICTURE "99"
@8,5 SAY "DATE:  " GET DATE_REC
@9,5 SAY "TIME:  " GET TIME_REC PICTURE "9999"
@10,5 SAY "LOCATION:  " GET LOCATION PICTURE ;
      "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"

```

```

SELECT 2
REPLACE CRNUM WITH TCRNUM
@11,5 SAY "COMPLAINANT--LAST:  " GET LASTNAME PICTURE ;
      "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
@12,18 SAY "FIRST:  " GET FIRSTNAME PICTURE "!!!!!!!!!!!!!"
@12,40 SAY "MIDDLE:  " GET MIDDLE PICTURE "!"
REPLACE CITY WITH "MUNCIE",STATE WITH "IN",ZIP WITH "47306"
@13,5 SAY "ADDRESS--STREET:  " GET STREET PICTURE ;
      "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
@14,14 SAY "CITY:  " GET CITY PICTURE "!!!!!!!!!!!!!"
@14,35 SAY "STATE:  " GET STATE PICTURE "!"
@14,47 SAY "ZIP:  " GET ZIP
REPLACE IDENTIFIER WITH "C"

```

```

SELECT 3
@15,5 SAY "PHONE:  " GET PHONE
@17,5 SAY "CAR (1):  " GET CAR_NUM PICTURE "99"
@17,20 SAY "UNIT (1):  " GET UNIT_NUM PICTURE "99"
@18,5 SAY "CAR (2):  " GET CAR_NUM_A PICTURE "99"
@18,20 SAY "UNIT (2):  " GET UNIT_NUM_A PICTURE "99"
@19,5 SAY "10-23:  " GET C10_23 PICTURE "9999"
@19,20 SAY "10-24:  " GET C10_24 PICTURE "9999"

```

```

READ

```

SET FORMAT TO DETAILS
READ
SET FORMAT TO

** 4.0 **

22

CLEAR ALL

SET INTENSITY OFF

RETURN

* INCIDENT (see INC2 for comments)
PARAMETERS TCRNUM

** 4.1 **

23

```
clear
select 2
use master
select 3
use incident
append blank
select 4
use complain
set intensity on
SELECT 3
@3,5 say "ENTER AN INCIDENT REPORT"
@5,5 SAY "PAGE: " GET PG_NUM
@5,15 SAY "OF: " GET TOTAL_PGS
REPLACE CRNUM WITH TCRNUM
@6,5 say "INCIDENT REPORT NUMBER: "+CRNUM
@7,5 SAY "INCIDENT TYPE: " GET INC_TYPE PICTURE "999"
@8,5 SAY "SUPERVISORY CORRECTION: " GET SUP_CORR;
    PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
@9,5 SAY "DATE OF OCCURRENCE: " GET DATE_OCC
@9,36 SAY "TIME OF OCCURRENCE: " GET TIME_OCC PICTURE "9999"
SELECT 4
LOCATE FOR CRNUM = TCRNUM
@10,5
? "    DATE REPORTED: ",DATE_REC
@12,5 SAY "TIME REPORTED: "+TIME_REC
SELECT 3
@13,5 SAY "HOW REPORTED: " GET HOW_REPT PICTURE "!!!!!!!!!!!!"
@14,5 SAY "RURAL/URBAN: " GET TYPE_LOCAL PICTURE "!"
SELECT 4
LOCATE FOR CRNUM = TCRNUM
@15,5 SAY "RECEIVED BY: "+PE
SELECT 2
LOCATE FOR CRNUM = TCRNUM .AND. IDENTIFIER = "C"
@16,5 SAY "REPORTED BY--LAST: "+LASTNAME
@17,18 SAY "FIRST: "+FIRSTNAME
@17,40 SAY "MIDDLE: "+MIDDLE
@18,5 SAY "ADDRESS--STREET: "+STREET
@19,14 SAY "CITY: "+CITY
@19,35 SAY "STATE: "+STATE
@19,47 SAY "ZIP: "+ZIP

SELECT 4
LOCATE FOR CRNUM = TCRNUM
@20,5 SAY "HOME PHONE: "+PHONE
SELECT 3
@20,32 SAY "BUSINESS PHONE: " GET REPT_BPHON
READ
CLEAR

@3,5 SAY "EXACT LOCATION OF INCIDENT: " GET CAM_LOCAL;
    PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
REPLACE CITY_INC WITH "MUNCIE",COUNTY_INC WITH "DELAWARE";
    STATE_INC WITH "IN"

@4,5 SAY "CITY: " GET CITY_INC PICTURE "!!!!!!!!!!!!"
@5,5 SAY "COUNTY: " GET COUNTY_INC PICTURE "!!!!!!!!!!!!"
@5,30 SAY "STATE: " GET STATE_INC PICTURE "!!"
STORE " " TO RESPONSE
@7,5 SAY "IS THERE AN INVOLVED PERSON? (Y/N) " GET RESPONSE;
    PICTURE "!"
```

```

READ
IF RESPONSE = "Y"
  SELECT 2
  APPEND BLANK
  REPLACE CRNUM WITH TCRNUM
  @9,5 SAY "HOW PERSON IS INVOLVED: " GET IDENTIFIER;
  PICTURE "!"
  @10,5 SAY "NAME--LAST: " GET LASTNAME;
  PICTURE "!!!!!!!!!!!!!!!!!!!!!!"
  @11,11 SAY "FIRST: " GET FIRSTNAME PICTURE "!!!!!!!!!!!!!!"
  @11,33 SAY "MIDDLE: " GET MIDDLE PICTURE "!"
  @12,5 SAY "ADDRESS--STREET: "GET STREET;
  PICTURE "!!!!!!!!!!!!!!!!!!!!!!"
  REPLACE CITY WITH "MUNCIE",STATE WITH "IN",;
  ZIP WITH "47306"
  @13,14 SAY "CITY: " GET CITY PICTURE "!!!!!!!!!!!!!!"
  @13,35 SAY "STATE: " GET STATE PICTURE "!"
  @13,47 SAY "ZIP: " GET ZIP
  SELECT 3
  @14,5 SAY "HOME PHONE: " GET INVA_PHONE
  @14,42 SAY "BUSINESS PHONE: " GET INVA_BPHON
  @15,5 SAY "SSN: " GET INVA_SSN
  @15,25 SAY "DOB: " GET INVA_DOB
  @16,5 SAY "DESCRIPTION: " GET INVA_PHYS;
  PICTURE "!!!!!!!!!!!!!!!!!!!!!!"
  REPLACE INVA_IDACS WITH .N.,INVA_NCICS WITH .N.
  @17,5 SAY "IDACS: " GET INVA_IDACS PICTURE "L"
  @17,17 SAY "NCICS: " GET INVA_NCICS PICTURE "L"
  STORE " " TO ANSWER
  @19,5 SAY "IS THERE ANOTHER INVOLVED PERSON? (Y/N)";
  GET ANSWER PICTURE "!"
  READ

```

```

IF ANSWER = "Y"
  CLEAR
  SELECT 2
  APPEND BLANK
  REPLACE CRNUM WITH TCRNUM
  @3,5 SAY "HOW PERSON IS INVOLVED: " GET IDENTIFIER;
  PICTURE "!"
  @4,5 SAY "NAME--LAST: " GET LASTNAME;
  PICTURE "!!!!!!!!!!!!!!!!!!!!!!"
  @5,18 SAY "FIRST: " GET FIRSTNAME PICTURE "!!!!!!!!!!!!!!"
  @5,40 SAY "MIDDLE: " GET MIDDLE PICTURE "!"
  REPLACE CITY WITH "MUNCIE",STATE WITH "IN",;
  ZIP WITH "47306"
  @6,5 SAY "ADDRESS--STREET: " GET STREET PICTURE;
  "!!!!!!!!!!!!!!!!!!!!!!"
  @7,14 SAY "CITY: " GET CITY PICTURE "!!!!!!!!!!!!!!"
  @7,35 SAY "STATE: " GET STATE PICTURE "!"
  @7,47 SAY "ZIP: " GET ZIP

  SELECT 3
  @8,5 SAY "HOME PHONE: " GET INVB_PHONE
  @8,35 SAY "BUSINESS PHONE: " GET INVB_BPHON
  @9,5 SAY "SSN: " GET INVB_SSN
  @9,25 SAY "DOB: " GET INVB_DOB
  @10,5 SAY "DESCRIPTION: " GET INVB_PHYS;
  PICTURE "!!!!!!!!!!!!!!!!!!!!!!"
  REPLACE INVB_IDACS WITH .N., INVB_NCICS WITH .N.
  @11,5 SAY "IDACS: " GET INVB_IDACS PICTURE "L"
  @11,15 SAY "NCICS: " GET INVB_NCICS PICTURE "L"
  READ
  ENDIF

```

```

  ENDIF
  CLEAR

```

```
@3,5 SAY "REVIEWER'S NAME--LAST: " GET REV_LN PICTURE;  
"!!!!!!!!!!!!!!!!!!!!!"  
@4,22 SAY "FIRST: " GET REV_FN PICTURE "!!!!!!!!!!!!!"  
@4,45 SAY "MIDDLE: " GET REV_MI PICTURE "!"  
@5,5 SAY "PE: " GET REV_PE PICTURE "99"  
@5,15 SAY "DATE: " GET DATE_REV  
@6,5 SAY "STATUS (S,A,C): " GET REVRECSTAT PICTURE "!"  
@8,5 SAY "OFFICER'S NAME--LAST: " GET OFF_LN PICTURE;  
"!!!!!!!!!!!!!!!!!!!!!"  
@9,21 SAY "FIRST: " GET OFF_FN PICTURE "!!!!!!!!!!!!!"  
@9,44 SAY "MIDDLE: " GET OFF_MI PICTURE "!"  
@10,5 SAY "PE: " GET OFF_PE PICTURE "99"  
@10,15 SAY "DATE: " GET DATE_REPT  
read  
return
```

```

*INC2.PRG (continuation of incident.prg)
*~~~~~*
* PROGRAM NAME:  INC2                NUMBER:  4.2  *
*              (INCIDENT continuation---no.: 4.1) *
*
* CALLING PROGRAM(S):  CHECKINC      *
*   PARAMETER(S):    TCRNUM          *
* PROGRAM(S) CALLED:  SUPPLEMN      *
*   PARAMETER(S):    TCRNUM          *
*
* PURPOSE:  This module, along with INCIDENT, allows *
* the user to enter an incident report.  The *
* TCRNUM is used to identify the correct record, *
* storing the information in the INCIDENT database.*
* Involved person's information is stored in the *
* MASTER database.  Users are given the option of *
* entering a supplemental report directly after *
* entering the incident report. *
*-----*

```

```

parameters tcrnum
set format to narrative
read
set format to

```

```

select 4
replace type_repor with "I"

```

```

clear
store " " to response
@3,5 say "Do you want to enter a supplemental report now? (Y/N)";
  get response picture "!"
read
if response = "Y"
  store .Y. to supp_rep
  do supplemn with tcrnum
endif

```

```

clear all
clear
set intensity off
return

```

```
*CASEDOC.PRG
*-----*
* PROGRAM NAME:  CASERPT                NUMBER:  4.3  *
*                (also CASE2 (no. 4.4), CASE3        *
*                (no. 4.5), CASE4 (no. 4.6))        *
*                                                    *
* CALLING PROGRAM(S):  CHECKCAS                *
*   PARAMETER(S):  TCRNUM (to CASERPT only)        *
* PROGRAM(S) CALLED:  SUPPLEMN (from CASE4 only)   *
*   PARAMETER(S):  TCRNUM                *
*                                                    *
* PURPOSE:  This module, as well as CASE2, CASE3,  *
*   and CASE4, allows the user to enter information *
*   for a case report.  The case report number is  *
*   passed to be stored with the information in the *
*   CASE database.  Names (witnesses, suspects,    *
*   victims) and addresses are stored in the MASTER *
*   database.  When the report is completely        *
*   entered, CASE4 gives the user the option of    *
*   entering a supplemental report.                *
*-----*
```



```

CLEAR
@3,5 SAY "RANGE FOR DATE OF OCCURRENCE--EARLIEST:  " ;
    GET D_OCC_EARL
@3,56 SAY "LATEST:  " GET D_OCC_LATE
@4,5 SAY "RANGE FOR TIME OF OCCURRENCE--EARLIEST:  " ;
    GET T_OCC_EARL
@4,56 SAY "LATEST:  " GET T_OCC_LATE
SELECT 4

```

```

LOCATE FOR CRNUM = TCRNUM
@4,72
? "    DATE REPORTED:  ",DATE_REC
@5,36 SAY "TIME REPORTED:  "+TIME_REC
SELECT 3

```

```

@6,5 SAY "HOW REPORTED:  " GET HOW_REPT PICTURE "!!!!!!!!!!!!!"
@7,5 SAY "RURAL/URBAN:  " GET TYPE_LOCAL PICTURE "!"
SELECT 4

```

```

@8,5 SAY "RECEIVED BY:  "+PE
SELECT 2

```

```

LOCATE FOR CRNUM = TCRNUM .AND. IDENTIFIER = "C"
@9,5 SAY "REPORTED BY--LAST:  "+LASTNAME
@10,18 SAY "FIRST:  "+FIRSTNAME
@10,40 SAY "MIDDLE:  "+MIDDLE
@11,5 SAY "ADDRESS--STREET:  "+STREET
@12,14 SAY "CITY:  "+CITY
@12,35 SAY "STATE:  "+STATE
@12,47 SAY "ZIP:  "+ZIP
SELECT 4

```

```

@13,5 SAY "HOME PHONE:  "+PHONE
SELECT 3

```

```

@13,32 SAY "BUSINESS PHONE:  " GET REPT_BPHON
@14,5 SAY "EXACT LOCATION OF OFFENSE:  " GET CAM_LOCAL ;
    PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
REPLACE CITY_OFF WITH "MUNCIE",TOWN_OFF WITH "CENTER",COUNTY_OFF;
    WITH "DELAWARE",STATE_OFF WITH "IN"
@15,5 SAY "CITY:  " GET CITY_OFF PICTURE "!!!!!!!!!!!!!"
@16,5 SAY "TOWNSHIP:  " GET TOWN_OFF PICTURE "!!!!!!!!!!!!!"
@17,5 SAY "COUNTY:  " GET COUNTY_OFF PICTURE "!!!!!!!!!!!!!"
@18,5 SAY "STATE:  " GET STATE_OFF PICTURE "!"

```

```

@19,5 SAY "WAS THERE A WITNESS TO THE CRIME? (Y/N) " GET ;
    WITNESS_YN PICTURE "L"
READ

```

```

CLEAR
IF WITNESS_YN
    @4,5 SAY "WITNESS/NEIGHBORHOOD CHECK (W/N):  " GET ;
        TYPE_WIT PICTURE "!"
    SELECT 2
    APPEND BLANK
    @5,5 SAY "WITNESS NAME--LAST:  " GET LASTNAME ;
        PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
    @6,19 SAY "FIRST:  " GET FIRSTNAME PICTURE "!!!!!!!!!!!!!"
    @6,41 SAY "MIDDLE:  " GET MIDDLE PICTURE "!"
    @7,5 SAY "ADDRESS--STREET:  " GET STREET ;
        PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
    REPLACE CITY WITH "MUNCIE",STATE WITH "IN",ZIP WITH "47306"
    @8,14 SAY "CITY:  " GET CITY PICTURE "!!!!!!!!!!!!!"
    @8,35 SAY "STATE:  " GET STATE PICTURE "!"
    @8,47 SAY "ZIP:  " GET ZIP

```

REPLACE CRNUM WITH TCRNUM, IDENTIFIER WITH "W" ** 4.3 **

SELECT 3

@9,5 SAY "HOME PHONE: " GET WIT_PHONE

@9,32 SAY "BUSINESS PHONE: " GET WIT_BPHON

@10,5 SAY "INTERVIEWED? (Y/N) " GET WIT_INTVU PICTURE "L"

@11,5 SAY "STATEMENT? (Y/N) " GET WIT_STTMT PICTURE "L"

READ

CLEAR

RETURN

```

*CASE2.PRG (continuation of caserpt.prg)                ** 4.4 **
STORE " " TO RESPONSE
IF WITNESS_YN
  @12,5 SAY "IS THERE A SECOND WITNESS? (Y/N) " GET RESPONSE;
  PICTURE "!"
  READ
ENDIF
IF RESPONSE = "Y"
  CLEAR
  @3,5 SAY "WITNESS/NEIGHBORHOOD CHECK (W/N): ";
  GET TYPE_WITB PICTURE "!"
  SELECT 2
  APPEND BLANK
  @4,5 SAY "WITNESS NAME--LAST: " GET LASTNAME ;
  PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
  @5,19 SAY "FIRST: " GET FIRSTNAME PICTURE "!!!!!!!!!!!!!!!!!"
  @5,41 SAY "MIDDLE: " GET MIDDLE PICTURE "!"
  @6,5 SAY "ADDRESS--STREET: " GET STREET ;
  PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
  REPLACE CITY WITH "MUNCIE",STATE WITH "IN", ;
  ZIP WITH "47306"
  @7,14 SAY "CITY: " GET CITY PICTURE "!!!!!!!!!!!!!!!!!"
  @7,35 SAY "STATE: " GET STATE PICTURE "!"
  @7,47 SAY "ZIP: " GET ZIP
  REPLACE CRNUM WITH TCRNUM,IDENTIFIER WITH "W"
  SELECT 3
  @9,5 SAY "HOME PHONE: " GET WITB_PHONE
  @9,32 SAY "BUSINESS PHONE: " GET WITB_BPHON
  @10,5 SAY "INTERVIEWED? (Y/N) " GET WITB_INTVU PICTURE "L"
  @11,5 SAY "STATEMENT? (Y/N) " GET WITB_STTMT PICTURE "L"
ENDIF
@12,5 SAY "IS THERE A SUSPECT? (Y/N) " GET SUSPECT_YN ;
  PICTURE "L"
READ
CLEAR
IF SUSPECT_YN
  @3,5 SAY "SUSPECT--NAMED: " GET SUS_NAMED PICTURE "L"
  @4,14 SAY "KNOWN: " GET SUS_KNOWN PICTURE "L"
  @5,14 SAY "KNOWN LOCATION: " GET SUS_LOCAL PICTURE "L"
  @6,14 SAY "IDENTIFIED: " GET SUS_IDENTD PICTURE "L"
  @7,14 SAY "PREVIOUSLY SEEN: " GET SUS_SEEN PICTURE "L"
  @8,14 SAY "DESCRIPTION: " GET SUS_DESCRB PICTURE "L"
  SELECT 2
  APPEND BLANK
  @9,5 SAY "NAME--LAST: " GET LASTNAME ;
  PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
  @10,11 SAY "FIRST: " GET FIRSTNAME PICTURE "!!!!!!!!!!!!!!!!!"
  @10,33 SAY "MIDDLE: " GET MIDDLE PICTURE "!"
  @11,5 SAY "ADDRESS--STREET: " GET STREET ;
  PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
  REPLACE CITY WITH "MUNCIE",STATE WITH "IN", ZIP WITH "47306"
  @12,14 SAY "CITY: " GET CITY PICTURE "!!!!!!!!!!!!!!!!!"
  @12,35 SAY "STATE: " GET STATE PICTURE "!"
  @12,47 SAY "ZIP: " GET ZIP
  REPLACE CRNUM WITH TCRNUM,IDENTIFIER WITH "S"
  SELECT 3
  @13,5 SAY "DOB: " GET SUS_DOB
  @13,21 SAY "SSN: " GET SUS_SSN
  STORE " " TO T1_PHYS
  STORE " " TO T2_PHYS

```



```

store " " to t1_phys
store " " to t2_phys
STORE " " TO RESPONSE
IF SUSPECT_YN
  @18,5 SAY "IS THERE A SECOND SUSPECT? (Y/N) " ;
  GET RESPONSE PICTURE "!"
  READ
  CLEAR
  IF RESPONSE = "Y"
    @3,5 SAY "SUSPECT--NAMED: " GET SUSB_NAMED PICTURE "L"
    @4,14 SAY "KNOWN: " GET SUSB_KNOWN PICTURE "L"
    @5,14 SAY "KNOWN LOCATION: " GET SUSB_LOCAL PICTURE "L"
    @6,14 SAY "IDENTIFIED: " GET SUSB_IDENT PICTURE "L"
    @7,14 SAY "PREVIOUSLY SEEN: " GET SUSB_SEEN PICTURE "L"
    @8,14 SAY "DESCRIPTION: " GET SUSB_DESCR PICTURE "L"
    SELECT 2
    APPEND BLANK
    @9,14 SAY "NAME--LAST: " GET LASTNAME ;
    PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
    @10,11 SAY "FIRST: " GET FIRSTNAME ;
    PICTURE "!!!!!!!!!!!!!!!!!"
    @10,33 SAY "MIDDLE: " GET MIDDLE PICTURE "!"
    @11,5 SAY "ADDRESS--STREET: " GET STREET ;
    PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
    REPLACE CITY WITH "MUNCIE",STATE WITH "IN",ZIP WITH ;
    "47306"
    @12,14 SAY "CITY: " GET CITY PICTURE "!!!!!!!!!!!!!!!!!"
    @12,35 SAY "STATE: " GET STATE PICTURE "!"
    @12,47 SAY "ZIP: " GET ZIP
    REPLACE CRNUM WITH TCRNUM,IDENTIFIER WITH "S"
    SELECT 3
    @13,5 SAY "DOB: " GET SUSB_DOB
    @13,21 SAY "SSN: " GET SUSB_SSN
    @14,5 SAY "DESCRIPTION: " GET T1_PHYS PICTURE;
    "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
    @15,19 GET T2_PHYS PICTURE;
    "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
    @16,5 SAY "IDENTIFIED BY: " GET SUSB_ID_BY ;
    PICTURE "!!!!!!!!!!!!!"
    @16,35 SAY "ARRESTED? (Y/N) " GET SUSB_ARRES ;
    PICTURE "L"
    @17,5 SAY "MUG? (Y/N) " GET SUSB_MUG PICTURE "L"
    @17,20 SAY "PRINT (Y/N) " GET SUSB_PRINT PICTURE "L"
    @17,35 SAY "C.R. CHECK (Y/N) " GET SUSB_RECORD;
    PICTURE "L"
  
```

```

  ENDIF
ENDIF
SELECT 3
@18,5 SAY "IS THERE A VEHICLE IDENTIFIED? (Y/N) " ;
  GET VEH_INVOLV PICTURE "L"
  READ
  REPLACE SUSB_PHYS WITH T1_PHYS+T2_PHYS
  CLEAR
  IF VEH_INVOLV
    @3,5 SAY "VEHICLE--SUSPECT: " GET VEH_SUSPCT PICTURE "L"
    @4,14 SAY "STOLEN: " GET VEH_STOLEN PICTURE "L"
    @5,14 SAY "RECOVERED: " GET VEH_RECQVD PICTURE "L"
    @6,14 SAY "OTHER: " GET VEH_OTHER PICTURE "L"
    @7,5 SAY "MAKE: " GET VEH_MAKE PICTURE "!!!!!!!"
    @7,20 SAY "COLOR: " GET VEH_COLOR PICTURE "!!!!"
    @7,33 SAY "YEAR: " GET VEH_YEAR PICTURE "99"
    @7,44 SAY "MODEL: " GET VEH_MODEL PICTURE "!!!!!!!"
  
```

```

@8,5 SAY "BODY S.: " GET VEH_BODY PICTURE "!!!!"          ** 4.5 **
@8,21 SAY "LIC. YR.: " GET VEH_LIC_YR PICTURE "99"
@8,36 SAY "LIC. ST.: " GET VEH_LIC_ST PICTURE "!!"
@9,5 SAY "LIC. NO.: " GET VEH_LIC_NO PICTURE "!!!!!!!!!!!!!!" 34
@9,30 SAY "WHERE HELD: " GET VEH_HOLD PICTURE "!!!!!!!!!!"
@10,5 SAY "VIN: " GET VEH_ID_NO ;
      PICTURE "!!!!!!!!!!!!!!!!!!!!!!"
ENDIF
@13,5 SAY "SIGNIFICANT MO OR LIMITED OPPORTUNITY? (Y/N) " ;
      GET MO PICTURE "L"
READ
IF MO
  @14,5 SAY "SIGNIFICANT MO? (Y/N) " GET SIGNIF_MO PICTURE "L"
  @14,30 SAY "LIMITED OPPORTUNITY? (Y/N) " ;
      GET LIMD_OPP PICTURE "L"
  @15,5 SAY "DESCRIBE: " GET S_L_DESCRP PICTURE ;
  "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
  @16,5 SAY "MOTIVE: " GET MOTIVE ;
      PICTURE "!!!!!!!!!!!!!!!!!!!!!!"
ENDIF
@17,5 SAY "TRACEABLE PROPERTY/SIGNIF. PHYS. EVIDENCE? (Y/N) " ;
      GET PROP_EVID PICTURE "L"
READ
CLEAR
RETURN

```

```

*CASE4.PRG (continuation of case3)          ** 4.6 **
IF PROP_EVID
  @3,5 SAY "TRACEABLE PROPERTY? (Y/N) " GET TRACE_PROP ;
    PICTURE "L"
  @4,5 SAY "SIGNIFICANT PHYSICAL EVIDENCE? (Y/N) " GET ;
    PHYS_EVID PICTURE "L"
  @5,5 SAY "PHOTO? (Y/N) " GET PHOTO PICTURE "L"
  @5,21 SAY "FINGERPRINT? (Y/N) " GET F_PRINT PICTURE "L"
  @5,43 SAY "OTHER? (Y/N) " GET EVID_OTHER PICTURE "L"
  @6,5 SAY "SCENE TECH. (Y/N) " GET SCENE_TECH PICTURE "L"
  @6,26 SAY "PE NO. TECH.: " GET SCENE_PE PICTURE "99"
  @7,5 SAY "PROPERTY RECOVERED? (Y/N) " GET PROP_RECOV ;
    PICTURE "L"
  @8,5 SAY "COLLECTED EVIDENCE? (Y/N) " GET COLL_EVID ;
    PICTURE "L"
  @9,5 SAY "INSTRUMENT, FORCE, WEAPON: " GET WEAPON_DES ;
    PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!"
endif
@10,5 SAY "PROPERTY STOLEN/RECOVERED STOLEN? (S/R) " GET ;
  MISS_PROP PICTURE "!"
@11,5 SAY "DESCRIPTION: " GET MPROP_DESC PICTURE ;
  "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
@12,5 SAY "IDENTIFICATION NO.: " GET MPROP_ID ;
  PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
@13,5 SAY "VALUE: $" GET MPROP_VAL PICTURE "###,###.##"
@14,5 SAY "PROPERTY STOLEN/RECOVERED STOLEN? (S/R) " GET ;
  MISS_PROP2 PICTURE "!"
@15,5 SAY "DESCRIPTION: " GET MPROP2_DES PICTURE ;
  "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
@16,5 SAY "IDENTIFICATION NO.: " GET MPROP2_ID ;
  PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
@17,5 SAY "VALUE: $" GET MPROP2_VAL PICTURE "###,###.##"
READ
CLEAR
@3,5 SAY "PROPERTY STOLEN/RECOVERED STOLEN? (S/R) " GET ;
  MISS_PROP3 PICTURE "!"
@4,5 SAY "DESCRIPTION: " GET MPROP3_DES PICTURE ;
  "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
@5,5 SAY "IDENTIFICATION NO.: " GET MPROP3_ID ;
  PICTURE "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
@6,5 SAY "VALUE: $" GET MPROP3_VAL PICTURE "###,###.##"
@7,5 SAY "IDACS? (Y/N) " GET IDACS PICTURE "L"
@7,22 SAY "NCIC? (Y/N) " GET NCIC PICTURE "L"
@8,5 SAY "TOTAL VALUE RECOVERED STOLEN: $" GET TOT_MPROPV ;
  PICTURE "##,###,###.##"
@9,5 SAY "TOTAL VALUE STOLEN: $" GET TOT_RECVAL ;
  PICTURE "##,###,###.##"
READ
SET FORMAT TO NARRATIVE
READ
SET FORMAT TO
STORE CRNUM TO TCRNUM

SELECT 4
REPLACE TYPE_REPOR WITH "C"

CLOSE DATABASE
USE CASESTAT
APPEND BLANK
clear
REPLACE CRNUM WITH TCRNUM
@3,5 SAY "INITIAL OFFICER STATUS: " GET INIT_STAT PICTURE "!"
@4,5 SAY "RECOMMEND TO CONTINUE: " GET INIT_REC PICTURE "!"

```

@5,5 SAY "INITIAL OFFICER NAME--LAST: " GET INIT_LN ; **4.6 **

PICTURE "!!!!!!!!!!!!!!!!!!!!!!"

@6,27 SAY "FIRST: " GET INIT_FN PICTURE "!!!!!!!!!!!!!!"

@6,50 SAY "MIDDLE: " GET INIT_MI PICTURE "!"

36

@7,5 SAY "PE: " GET INIT_PE PICTURE "99"

@7,15 SAY "DATE: " GET INIT_DATE

@8,5 SAY "FIELD SUPERVISOR STATUS: " GET SUP_STAT PICTURE "!"

@9,5 SAY "RECOMMEND TO CONTINUE: " GET SUP_REC PICTURE "!"

@10,5 SAY "FIELD SUPERVISOR NAME--LAST: " GET SUP_LN ;

PICTURE "!!!!!!!!!!!!!!!!!!!!!!"

@11,28 SAY "FIRST: " GET SUP_FN PICTURE "!!!!!!!!!!!!!!"

@11,50 SAY "MIDDLE: " GET SUP_MI PICTURE "!"

@12,5 SAY "PE: " GET SUP_PE PICTURE "99"

@12,15 SAY "DATE: " GET SUP_DATE

@13,5 SAY "INVESTIGATIVE COORDINATOR STATUS: " GET INVC_REC ;

PICTURE "!"

@14,5 SAY "INVEST. COORD. NAME--LAST: " GET INVC_LN ;

PICTURE "!!!!!!!!!!!!!!!!!!!!!!"

@15,26 SAY "FIRST: " GET INVC_FN PICTURE "!!!!!!!!!!!!!!"

@15,48 SAY "MIDDLE: " GET INVC_MI PICTURE "!"

@16,5 SAY "PE: " GET INVC_PE PICTURE "99"

@16,15 SAY "DATE: " GET INVC_DATE

@16,32 SAY "STATUS: " GET FINAL_STAT PICTURE "!"

@17,5 SAY "ASSIGNED INVESTIGATOR NAME--LAST: " GET AINV_LN ;

PICTURE "!!!!!!!!!!!!!!!!!!!!!!"

@18,33 SAY "FIRST: " GET AINV_FN PICTURE "!!!!!!!!!!!!!!"

@18,55 SAY "MIDDLE: " GET AINV_MI PICTURE "!"

@19,5 SAY "PE: " GET AINV_PE PICTURE "99"

@19,15 SAY "DATE: " GET AINV DATE

READ

CLEAR

store " " to response

@3,5 say "Do you want to enter a supplemental report now? (Y/N) " ;

get response picture "!"

read

if response = "Y"

do supplemn with tcrnum

endif

clear all

SET INTENSITY OFF

RETURN

```
* SUPPLEMN.PRG
* ~~~~~*
* PROGRAM NAME:  SUPPLEMN          NUMBER:  4.7  *
*
* CALLING PROGRAM(S):  CHECKSUP, INC2, CASE4      *
*   PARAMETER(S):  TCRNUM                        *
* PROGRAM(S) CALLED:  NONE                       *
*   PARAMETER(S):  NONE                         *
*
* PURPOSE:  This module allows the user to enter *
*   the narrative portion of the supplemental report.*
* ~~~~~*
```

```
parameters tcrnum
clear
use suppleme
append blank
```

```
replace crnum with tcrnum
set format to narrative
read
set format to
clear all
return
```

```

* EDITREC.PRG
*-----*
* PROGRAM NAME:  EDITREC                NUMBER:  4.8  *
*
* CALLING PROGRAM(S):  EDITMENU          *
*   PARAMETER(S):  NONE                  *
* PROGRAM(S) CALLED:  EDITCOM,EDITCASE,EDITINC,    *
*                   EDITSUP              *
*   PARAMETER(S):  TCRNUM                *
*
* PURPOSE:  This module prompts the user for      *
*   a case report number and verifies that a report *
*   of that number and type has been filed.      *
*-----*

```

```

SET INTENSITY ON
store .f. to valid
do while .not. valid
  store "00000000" to tcrnum
  clear
  @5,15 say "ENTER A REPORT NUMBER" get tcrnum
  read
  use &varbfile
  locate for crnum = tcrnum
  if crnum <> tcrnum
    store " " to response
    @7,15 say "This record has NOT been filed."
    @9,15 say "If you wish to return to the menu,"
    @10,15 say "type 'yes'. If you wish to try another"
    @11,15 say "number, type `no`." get response;
    picture "!!!"
    read
    if response = "YES"
      set intensity off
      close database
      return
    endif
  else
    close database
    clear
    do case
      case varbfile = "complain"
        do editcom with tcrnum
      case varbfile = "case"
        do editcase with tcrnum
      case varbfile = "incident"
        do editinc with tcrnum
      case varbfile = "suppleme"
        do editsup with tcrnum
    endcase

    set intensity off
    close database
    return
  endif
enddo

```

```

* PRINTREC.PRG
*-----*
* PROGRAM NAME:  PRINTREC                NUMBER:  4.9  *
*
* CALLING PROGRAM(S):  PRINTMENU          *
*   PARAMETER(S):  NONE                    *
* PROGRAM(S) CALLED:  PRINTCOM, PRINTCASE, PCASE2,   *
*                   PCASE3, PCASE4, PRINTINC,       *
*                   PRINTSUP                 *
*   PARAMETER(S):  TCRNUM, RESPONSE           *
*
* PURPOSE:  This module prompts the user to enter   *
*   number of the report he wants printed.  The     *
*   number is checked to verify a report has been   *
*   entered.  The correct printing routine is then  *
*   called, passing the TCRNUM of the desired report.*
*   RESPONSE is used by the routines to print a case *
*   report as a means of passing whether the report *
*   is printed on paper or simply on the screen.    *
*-----*

```

```

set intensity on
store .f. to valid
do while .not. valid
  store "00000000" to tcrnum
  clear
  @5,15 say "ENTER A REPORT NUMBER" get tcrnum
  read
  use &varbfile
  locate for crnum = tcrnum
  if crnum <> tcrnum
    store " " to response
    @7,15 say "This record has NOT been filed."
    @9,15 say "If you wish to return to the menu,"
    @10,15 say "type 'yes'.  If you wish to try another"
    @11,15 say "number, type 'no'." get response ;
    picture "!!!"
    read
    if response = "YES"
      set intensity off
      close database
      return
    endif
  else
    close database
    clear
    do case
      case varbfile = "complain"
        do printcom with tcrnum
      case varbfile = "case"
        store " " to response
        do printcas with tcrnum,response
        do pcase2 with tcrnum,response
        do pcase3 with tcrnum,response
        do pcase4 with tcrnum,response
      case varbfile = "incident"
        do printinc with tcrnum
      case varbfile = "suppleme"
        do printsup with tcrnum
    endcase

    set intensity off
    close database

```

```
        return
    endif
enddo
```

** 4.9 **

```

*SINGLE_T.PRG
*-----*
* PROGRAM NAME:  SINGLE_T          NUMBER:  4.10 *
*
* CALLING PROGRAM(S):  TALLIES          *
*   PARAMETER(S):  NONE                *
* PROGRAM(S) CALLED:  NONE                *
*   PARAMETER(S):  NONE                *
*
* PURPOSE:  This module shows the user the last date *
*   for which tallies were run and prompts the user *
*   for the date which should be run now.  THE DATE *
*   MUST BE LATER THAN THE LAST DATE FOR WHICH THE *
*   TALLIES WERE CALCULATED!  If the date is the *
*   start of a new month, the old tallies for that *
*   month are cleared.  Then tallies for the speci- *
*   fied date are calculated and stored in the *
*   specific monthly total.            *
*-----*

```

```

clear
store " / / " to cdate
select 5
use complain
select 4
use case
select 3
use incident
select 2
use t_month index t_index
locate for type_off = "001"
@3,5 say "The last date for which tallies were calculated was ";
      + dtoc(date_upd)
@6,5 say "Enter date for which you'd like to calculate now. "
@7,5 say "Date must be LATER than date above!!!" ;
      get cdate picture "@D"
read
@9,5 say "Please wait for tallying . . ."
store ctod(cdate) to new_date
store month(new_date) to m_pos
if m_pos > month(date_upd)
  goto top
do case
  case m_pos = 1
    replace all jan_count with 0
  case m_pos = 2
    replace all feb_count with 0
  case m_pos = 3
    replace all mar_count with 0
  case m_pos = 4
    replace all apr_count with 0
  case m_pos = 5
    replace all may_count with 0
  case m_pos = 6
    replace all jun_count with 0
  case m_pos = 7
    replace all jul_count with 0
  case m_pos = 8
    replace all aug_count with 0
  case m_pos = 9
    replace all sep_count with 0
  case m_pos = 10

```

```

        replace all oct_count with 0
    case m_pos = 11
        replace all nov_count with 0
    case m_pos = 12
        replace all dec_count with 0
    endcase
endif
select 5
locate for date_rec = new_date
do while .not. EOF()
    store crnum to tcrnum
    store type_repor to repor_type
    if repor_type = "I"
        select 3
        locate for tcrnum = crnum
        store inc_type to off_type
    else if repor_type = "C"
        select 4
        locate for tcrnum = crnum
        store type_off to off_type
    endif
endif
select 2
goto top
seek off_type
do case
    case m_pos = 1
        replace jan_count with jan_count + 1
    case m_pos = 2
        replace feb_count with feb_count + 1
    case m_pos = 3
        replace mar_count with mar_count + 1
    case m_pos = 4
        replace apr_count with apr_count + 1
    case m_pos = 5
        replace may_count with may_count + 1
    case m_pos = 6
        replace jun_count with jun_count + 1
    case m_pos = 7
        replace jul_count with jul_count + 1
    case m_pos = 8
        replace aug_count with aug_count + 1
    case m_pos = 9
        replace sep_count with sep_count + 1
    case m_pos = 10
        replace oct_count with oct_count + 1
    case m_pos = 11
        replace nov_count with nov_count + 1
    case m_pos = 12
        replace dec_count with dec_count + 1
    endcase
select 5
continue
enddo
select 2
goto top
replace date_upd with new_date
clear all
@9,5 say "Tallying is complete.
wait
clear
return

```

```

*VIEW_T.PRG
*-----*
* PROGRAM NAME:  VIEW_T                NUMBER:  4.11 *
*
* CALLING PROGRAM(S):  TALLIES          *
*   PARAMETER(S):  NONE                 *
* PROGRAM(S) CALLED:  FIND_START       *
*   PARAMETER(S):  B, START, HHEAD     *
*
* PURPOSE:  This module allows the user to see the *
* tallies for the 110 types of activity entered on *
* case and incident reports.  The user is asked to *
* specify for what months he wants the tallies (up *
* to 12 months are available).  An option is also *
* available to print the report to paper.  B is *
* used to designate the starting month's numeric *
* equivalent; START and HHEAD are returned with *
* information to be printed in the report. *
*-----*

```

```

clear
use t_month index t_index

store " " to hhead
store " " to start
store " " to bmonth
store " " to emonth
@3,5 say "View tallies allows you to specify seeing up to 12 contiguous"
@4,5 say "months of totals.  Only the most recent 12 months data"
@5,5 say "is available."
do while bmonth < "01" .or. bmonth > "12" .or. emonth < "01" ;
    .or. emonth > "12"
    @7,5 say "Enter number of beginning of range: " get bmonth;
    picture "99"
    @9,5 say "Enter number of end of range: " get emonth;
    picture "99"
    read
enddo

do case
    case emonth = "01"
        end = "jan_count"
    case emonth = "02"
        end = "feb_count"
    case emonth = "03"
        end = "mar_count"
    case emonth = "04"
        end = "apr_count"
    case emonth = "05"
        end = "may_count"
    case emonth = "06"
        end = "jun_count"
    case emonth = "07"
        end = "jul_count"
    case emonth = "08"
        end = "aug_count"
    case emonth = "09"
        end = "sep_count"
    case emonth = "10"
        end = "oct_count"
    case emonth = "11"
        end = "nov_count"

```

```

    case emonth = "12"
        end = "dec_count"
    endcase
store " " to response
@12,5 say "Do you want the report printed on paper? (Y/N) ";
    get response picture "!"
read
if response = "Y"
    set print on
    set margin to 0
endif
set heading off
if bmonth = emonth
    do find_start with val(bmonth),start,hhead
    if response = "Y"
        ?" OFFENSE      ", "&hhead"
        list all type_off,type_name,&start off
    else
        ?" OFFENSE      ", "&hhead"
        display all type_off,type_name,&start off
    endif
else
    @14,5 say "Please wait . . ."
    store val(bmonth) to b
    store val(emonth) to e
    do find_start with b,start,hhead
    store start to disp_string
    store hhead to head_string
    do while b <> e
        b = b + 1
        if b = 13
            b = 1
        endif
        do find_start with b,start,hhead
        store disp_string+" "+start to disp_string
        store head_string+" "+hhead to head_string
    enddo
    if response = "Y"
        ?" OFFENSE      ", "&head_string"
        list all type_off,type_name,&disp_string off
    else
        ?" OFFENSE      ", "&head_string"
        display all type_off,type_name,&disp_string off
    endif
endif
endif

set memowidth to
set print off
set device to screen
wait
set heading on
clear
clear all
return

```

```

*DAILY_SUM.PRG
*~~~~~*
* PROGRAM NAME:  DAILY_SUM          NUMBER:  4.12 *
*
* CALLING PROGRAM(S):  TALLIES          *
*   PARAMETER(S):  NONE                *
* PROGRAM(S) CALLED:  NONE              *
*   PARAMETER(S):  NONE                *
*
* PURPOSE:  This module allows the user to prepare *
* the daily summary report for a specified date.  *
* The COMPLAIN database is searched to find all   *
* complaints entered on that date.  Then more    *
* information is taken from the CASE or INCIDENT  *
* database and the MASTER database.  Only select  *
* information is displayed to allow the user to  *
* enter a summary.  Information for the daily    *
* reports is stored in the DAILY_REP database,   *
* which is purged at the start of this routine,  *
* since the daily summaries do not need to be kept.*
*
-----*

```

```

clear
store " / / " to bdate
select 2
use complain
select 3
use case
select 4
use incident
select 5
use master
select 6
use t_month
select 7
use daily_rep
delete all
pack

```

```

@3,5 say "PREPARE DAILY SUMMARY REPORT"
@5,5 say "Enter date for which you want to prepare report. "
@6,5 say "Date: " get bdate picture "@D"
read

```

```

clear
@3,5 say "INFORMATION FROM REPORTS--TYPE YOUR SUMMARY"
select 2
locate for ctod(bdate) = date_rec
do while .not. EOF()
  @5,5 clear
  @5,5 say "DATE: " +dtoc(date_rec)
  @6,5 say "COMPLAINT: " +complaint
  @7,5 say "CR NUMBER: " +crnum
  store crnum to tcrnum
  store time_rec to ttime
  if type_repor = "I"
    select 4
    locate for tcrnum = crnum
    @8,5 say "TYPE OF OFFENSE: " +inc_type
    store inc_type to type
    select 6
    locate for type = type off

```

```

** 4.12 **
@8,30 say type_name
select 5
locate for tcrnum = crnum .and. identifier <> "C"
@9,5 say "INVOLVED PERSON: "+lastname+", "+firstname+; 46
      " "+middle+" (" +identifier+)"
continue
@10,5 say "INVOLVED PERSON: "+lastname+", "+firstname+;
      " "+middle+" (" +identifier+)"
select 4
@12,5 say "NARRATIVE: "
?? narrative
wait
@12,0 clear
else if type_report = "C"
select 3
locate for tcrnum = crnum
@8,5 say "TYPE OF OFFENSE: "+type_off
store type_off to type
select 6
locate for type = type_off
@8,30 say type_name
select 5
locate for tcrnum = crnum .and. identifier <> "C"
store 8 to line_ct
do while .not. EOF()
  line_ct = line_ct + 1
  @ line_ct,5 say "NAME: "+lastname+", "+firstname+;
    " "+middle+" (" +identifier+)"
  continue
enddo
select 3
@14,5 say "NARRATIVE: "
?? narrative
wait
@14,0 clear
endif
endif
select 7
append blank
replace crnum with tcrnum,time_rec with ttime
set format to summarize
read
set format to
clear
select 2
continue
enddo
store " " to response
@3,5 say "Do you want the report printed on paper? (Y/N) ";
  get response picture "!"
read
set memowidth to 40
if response = "Y"
  set device to print
  set margin to 0
  store "to print" to print_cont
else
  store " " to print_cont
endif
clear
select 7
goto top
@3,20 say "Ball State University Police Department"
@4,20 say "          Daily Summary"
@5,20 say "          "+bdate
@9,0

```

```
set device to screen
set heading off
do while .not. EOF()
    set margin to 10
    display crnum,time_rec off &print_cont
    set margin to 35
    display summary off &print_cont
    skip
enddo
set heading on
set print off
wait

clear
clear all
return
```

```

* NAME.PRG
*-----*
* PROGRAM NAME:  NAME                      NUMBER:  4.13 *
*
* CALLING PROGRAM(S):  SEARCH                *
*   PARAMETER(S):  NONE                      *
* PROGRAM(S) CALLED:  N_COMP, N_VICTIM, N_WIT, *
*                   N_OFFEND, N_SUSPEC, N_MP, *
*                   N_APPLIC, N_RUN          *
*   PARAMETER(S):  NONE                      *
*
* PURPOSE:  This module allows the user to specify *
*   the type of name on which to search the MASTER *
*   database.  Individual programs are called for *
*   the type of search selected.             *
*-----*

```

```

clear
store 0 to choice
do while choice < 9
  @3,15 say "SEARCH ON NAME"
  @6,15 say "1.  Complainant"
  @7,15 say "2.  Victim"
  @8,15 say "3.  Witness"
  @9,15 say "4.  Offender"
  @10,15 say "5.  Suspect"
  @11,15 say "6.  Missing person"
  @12,15 say "7.  Applicant"
  @13,15 say "8.  Runaway"
  @14,15 say "9.  Return to previous menu"
  @16,15 say "Enter your selection:" get choice picture;
  "@z"
read
@16,15 clear
if choice > 9 .or. choice < 1
  @16,15 say "Please review the options shown above,"
  @17,15 say "and this time select one to them!"
  store 0 to choice
  wait
  @16,0 clear
else
  do case
    case choice = 1
      do n_comp
    case choice = 2
      do n_victim
    case choice = 3
      do n_wit
    case choice = 4
      do n_offend
    case choice = 5
      do n_suspec
    case choice = 6
      do n_mp
    case choice = 7
      do n_applic
    case choice = 8
      do n_run
    case choice = 9
      clear
      return
  endcase
  store 0 to choice

```

endif
enddo

** 4.13 **

```

* CRNO.PRG
*-----*
* PROGRAM NAME:  CRNO                      NUMBER:  4.14 *
*
* CALLING PROGRAM(S):  SEARCH              *
*   PARAMETER(S):  NONE                    *
* PROGRAM(S) CALLED:  NONE                 *
*   PARAMETER(S):  NONE                    *
*
* PURPOSE:  This module prompts the user to enter a *
*   case report number then searches the complaint *
*   database to see if a report of that number has *
*   been entered.  Pertinent information from that *
*   report is displayed if the report exists.      *
*-----*

```

```

clear
set intensity on
store "0000-000" to tcrnum
select 2
use complain
@3,5 say "SEARCHING ON REPORT NUMBER"
@6,5 say "Enter a report number:  " get tcrnum
read
@8,5 say "Search in progress . . ."
locate for crnum = tcrnum
if crnum = tcrnum
    display complaint,date_rec,time_rec for crnum = tcrnum
else
    @13,5 say "The database has no report of that number."
endif
@16,15
wait
clear
set intensity off
clear all
return

```

```

* CRIME.PRG
*-----*
* PROGRAM NAME:  CRIME                      NUMBER:  4.15 *
*
* CALLING PROGRAM(S):  SEARCH                *
*   PARAMETER(S):  NONE                      *
* PROGRAM(S) CALLED:  NONE                  *
*   PARAMETER(S):  NONE                      *
*
* PURPOSE:  This module allows the user to search *
* the CASE and INCIDENT databases for a particular *
* type of crime or activity.  The code numbers *
* 001 to 110 should be used to start the search. *
* Information from all reports with that code as *
* the type of activity will be displayed.      *
*-----*

```

```

clear
set intensity on
store " " to ttype
select 2
use incident
select 3
use case
@3,15 say "SEARCHING ON TYPE OF CRIME/INCIDENT"
@6,15 say "Enter number code: " get ttype picture "999"
read
@9,15 say "Search in progress . . ."
locate for ttype = type_off
do while .not. EOF()
    display crnum,d_occ_earl,d_occ_late,cam_local
    continue
enddo
select 2
locate for ttype = inc_type
do while .not. EOF()
    display crnum,date_occ,cam_local
    continue
enddo
wait
clear
@3,15 say "The database has no more reports for this type."
wait
clear
clear all
set intensity off
return

```

```

* STATUS.PRG
* ~~~~~*
* PROGRAM NAME:  STATUS                      NUMBER:  4.16 *
*
* CALLING PROGRAM(S):  SEARCH                *
*   PARAMETER(S):  NONE                      *
* PROGRAM(S) CALLED:  ALL_CASE, ALL_INC, ONE_CASE, *
*                   ONE_INC                 *
*   PARAMETER(S):  NONE                      *
*
* PURPOSE:  This module gives the user options to *
*   run routines to find all active case or incident *
*   reports or to report the status of one specific *
*   report.                                     *
* ~~~~~*

```

```

clear
store 0 to choice
do while choice < 5
  @3,15 say "SEARCH ON STATUS"
  @6,15 say "1. All active case reports"
  @7,15 say "2. All active incident reports"
  @8,15 say "3. One specific case report"
  @9,15 say "4. One specific incident report"
  @10,15 say "5. Return to previous menu"
  @16,15 say "Enter your selection:" get choice picture;
  "@z"
read
@16,15 clear
if choice > 5 .or. choice < 1
  @16,15 say "Please review the options shown above,"
  @17,15 say "and this time select one to them!"
  store 0 to choice
  wait
  @16,0 clear
else
  do case
    case choice = 1
      do all_case
    case choice = 2
      do all_inc
    case choice = 3
      do one_case
    case choice = 4
      do one_inc
    case choice = 5
      clear
      return
    endcase
  store 0 to choice
endif
enddo

```

```

* VEHICLE.PRG
*-----*
* PROGRAM NAME:  VEHICLE                NUMBER:  4.17 *
*
* CALLING PROGRAM(S):  SEARCH                *
*   PARAMETER(S):  NONE                *
* PROGRAM(S) CALLED:  V_MAKE, V_COLOR, V_YEAR, *
*                   V_MODEL, V_BODY, V_LIC_YR, *
*                   V_LIC_ST, V_LIC_NO        *
*
* PURPOSE:  This module offers the user options for *
*   a vehicle's feature on which to search.  The *
*   appropriate routine is then run.          *
*-----*

```

```

clear
store 0 to choice
do while choice < 9
  @3,15 say "SEARCH ON VEHICLE"
  @6,15 say "1.  Make"
  @7,15 say "2.  Color"
  @8,15 say "3.  Year"
  @9,15 say "4.  Model"
  @10,15 say "5.  Body"
  @11,15 say "6.  License year"
  @12,15 say "7.  License state"
  @13,15 say "8.  License number"
  @14,15 say "9.  Return to previous menu"
  @16,15 say "Enter your selection:" get choice picture;
  "@z"
read
@16,15 clear
if choice > 9 .or. choice < 1
  @16,15 say "Please review the options shown above,"
  @17,15 say "and this time select one to them!"
  store 0 to choice
  wait
  @16,0 clear
else
  do case
    case choice = 1
      do v_make
    case choice = 2
      do v_color
    case choice = 3
      do v_year
    case choice = 4
      do v_model
    case choice = 5
      do v_body
    case choice = 6
      do v_lic_yr
    case choice = 7
      do v_lic_st
    case choice = 8
      do v_lic_no
    case choice = 9
      clear
      return
  endcase
  store 0 to choice
endif
enddo

```

```

* TYPE_ID.PRG
*-----*
* PROGRAM NAME:  TYPE_ID                NUMBER:  4.18 *
*
* CALLING PROGRAM(S):  SEARCH          *
*   PARAMETER(S):  NONE                *
* PROGRAM(S) CALLED:  INV_SEAR        *
*   PARAMETER(S):  NONE                *
*
* PURPOSED:  This module offers the user options for *
* what type of person for which to search the *
* MASTER database.  INV_CODE is used as a macro to *
* represent the user's choice of types.  INV_SEAR *
* performs the actual search.          *
*-----*

```

```

clear
store 0 to choice
do while choice < 9
  @3,15 say "SEARCH ON TYPE OF INVOLVEMENT"
  @6,15 say "1.  Complainant"
  @7,15 say "2.  Victim"
  @8,15 say "3.  Witness"
  @9,15 say "4.  Offender"
  @10,15 say "5.  Suspect"
  @11,15 say "6.  Missing person"
  @12,15 say "7.  Applicant"
  @13,15 say "8.  Runaway"
  @14,15 say "9.  Return to previous menu"
  @16,15 say "Enter your selection:" get choice picture;
  "@z"
read
@16,15 clear
if choice > 9 .or. choice < 1
  @16,15 say "Please review the options shown above,"
  @17,15 say "and this time select one to them!"
  store 0 to choice
  wait
  @16,0 clear
else
  do case
    case choice = 1
      store "C" to inv_code
    case choice = 2
      store "V" to inv_code
    case choice = 3
      store "W" to inv_code
    case choice = 4
      store "O" to inv_code
    case choice = 5
      store "S" to inv_code
    case choice = 6
      store "M" to inv_code
    case choice = 7
      store "A" to inv_code
    case choice = 8
      store "R" to inv_code
    case choice = 9
      clear
      return
  endcase
do inv_sear
store 0 to choice

```

endif
enddo

** 4.18 **

```

*RANGE.PRG
*-----*
* PROGRAM NAME:  RANGE                      NUMBER:  4.19 *
*
* CALLING PROGRAM(S):  SEARCH                *
*   PARAMETER(S):  NONE                      *
* PROGRAM(S) CALLED:  NONE                    *
*   PARAMETER(S):  NONE                      *
*
* PURPOSE:  This module prompts the user for two *
*   dates to specify the range of the search.  The *
*   second date may be the same as the first date, *
*   but the second date should NOT be chronologi- *
*   cally earlier than the first date!  The COMPLAIN *
*   database is used to display pertinent informa- *
*   tion about complaints issued during that range *
*   of dates. *
*-----*

```

```

clear
set intensity on
select 2
use complain

store " / / " to bdate
store " / / " to edate
@3,5 say "SEARCHING ON RANGE OF DATES"
@6,5 say "Enter beginning date: " get bdate picture "@D"
@8,5 say "Enter ending date: " get edate picture "@D"
@10,5 say "Make sure the correct data files are loaded "
@11,5 say "for the dates you have selected. "
read

set heading off
? " date      time CR no.  complaint"
locate for date_rec >= ctod(bdate) .and. date_rec <= ctod(edate)
do while .not. EOF()
    display next 1 date_rec,time_rec,crnum,complaint off
    continue
enddo

set heading on
wait
set intensity off
clear
clear all
return

```

```

*EDITCOM.PRG
*-----*
* PROGRAM NAME:  EDITCOM                NUMBER:  5.0  *
*
* CALLING PROGRAM(S):  EDITREC          *
*   PARAMETER(S):  TCRNUM              *
* PROGRAM(S) CALLED:  NONE              *
*   PARAMETER(S):  NONE                *
*
* PURPOSE:  This module allows the user to edit the *
*   various fields of a complaint card.  TCRNUM is  *
*   used to find the desired record.          *
*-----*

```

```

parameters tcrnum
select 2
use master
select 3
use complain

edit fields complaint,crnum,pe,date_rec,time_rec,location;
  for crnum = tcrnum
select 2
edit fields lastname,firstname,middle,street,city,state,;
  zip for crnum = tcrnum .and. identifier = "C"
select 3
edit fields phone,car_num,unit_num,car_num_a, unit_num_a,;
  cl0_23,cl0_24 for crnum = tcrnum
set format to details
edit for crnum = tcrnum
set format to
clear all
return

```

```

* EDITCASE.PRG
*-----*
* PROGRAM NAME:  EDITCASE          NUMBER:  5.1  *
*
* CALLING PROGRAM(S):  EDITREC      *
*   PARAMETER(S):  TCRNUM          *
* PROGRAM(S) CALLED:  NONE          *
*   PARAMETER(S):  NONE            *
*
* PURPOSE:  This module allows the user to edit the *
*   various fields of a case report.  TCRNUM is used *
*   to find the desired record.    *
*-----*

```

```
parameters tcrnum
```

```
select 2
use master
select 3
use case
select 4
use casestat
```

```
select 3
edit fields page_num,total_pgs,crnum,type_off,sup_corr ;
  for crnum = tcrnum
select 2
set message to "VICTIM INFORMATION"
edit fields lastname,firstname,middle,street,city,state,zip ;
  for crnum = tcrnum .and. identifier = "V"
select 3
edit fields responsibl,vic_phone,vic_bphon,vic_sex,vic_race, ;
  vic_dob,vic_age,vic_ssn,vic_injury,vic_treat ;
  for crnum = tcrnum
set message to
edit fields d_occ_earl,d_occ_late,t_occ_earl,t_occ_late, ;
  how_rept,type_local for crnum = tcrnum
select 2
set message to "REPORTER INFORMATION"
edit fields lastname,firstname,middle,street,city,state,zip ;
  for crnum = tcrnum .and. identifier = "C"
select 3
edit fields rept_bphon,cam_local,city_off,town_off,county_off, ;
  state_off for crnum = tcrnum
set message to "WITNESS INFORMATION"
edit fields witness_yn,type_wit for crnum = tcrnum
select 2
edit fields lastname,firstname,middle,street,city,state,zip ;
  for crnum = tcrnum .and. identifier = "W"
select 3
edit fields wit_phone,wit_bphon,wit_intvu,wit_sttmt,type_witb, ;
  witb_phone,witb_bphon,witb_intvu,witb_sttmt for crnum = tcrnum
set message to "SUSPECT INFORMATION"
edit fields suspect_yn,sus_named,sus_known,sus_local,sus_idntd, ;
  sus_seen,sus_descrb for crnum = tcrnum
select 2
edit fields lastname,firstname,middle,street,city,state,zip ;
  for crnum = tcrnum .and. identifier = "S"
select 3
edit fields sus_dob,sus_ssn,sus_phys,sus_id_by,sus_arrest, ;
  sus_mug,sus_print,sus_record,susb_named,susb_known, ;
  susb_local,susb_idnt,susb_seen,susb_descr ;
  for crnum = tcrnum
edit fields susb_dob,susb_ssn,susb_phys,susb_id_by, ;

```

```
    susb_arres,susb_mug,susb_print,susb_record for crnum = tcrnum
set message to "VEHICLE INFORMATION"          ** 5.1 **
edit fields veh_involv,veh_suspct,veh_stolen,veh_recovd,veh_other, ; 59
    veh_make,veh_color,veh_year,veh_model,veh_body,veh_lic_yr, ;
    veh_lic_st,veh_lic_no,veh_hold,veh_id_no for crnum = tcrnum
set message to "SIGNIFICANT MO/LIMITED OPPORTUNITY"
edit fields mo,signif_mo,limd_opp,s_1_descrp,motive ;
    for crnum = tcrnum
set message to "TRACEABLE PROPERTY/SIGNIF. PHYS. EVIDENCE"
edit fields prop_evid,trace_prop,phys_evid,photo,f_print, ;
    evid_other,scene_tech,scene_pe,prop_recov,coll_evid, ;
    weapon_des,miss_prop,mprop_desc,mprop_id,mprop_val ;
    for crnum = tcrnum
edit fields miss_prop2,mprop2_des,mprop2_id,mprop2_val, ;
    miss_prop3,mprop3_des,mprop3_id,mprop3_val,idacs,ncic, ;
    tot_mpropv,tot_recval for crnum = tcrnum
set message to
set format to narrative
edit for crnum = tcrnum
set format to
select 4
edit fields init_stat,init_rec,init_ln,init_fn,init_mi,init_pe, ;
    init_date,sup_date,sup_rec,sup_ln,sup_fn,sup_mi,sup_pe, ;
    sup_date for crnum = tcrnum
edit fields invc_rec,invc_ln,invc_fn,invc_mi,invc_pe,invc_date, ;
    final_stat,ainv_ln,ainv_fn,ainv_mi,ainv_pe,ainv_date ;
    for crnum = tcrnum
clear
clear all
return
```