

ACCOUNTABILITY 2.0

JOHN EDWARD PERCEVAULT

B. Ed., University of Lethbridge, 1979

A One-Credit Project
Submitted to the Faculty of Education
of The University of Lethbridge
in Partial Fulfillment of the
Requirements for the Degree

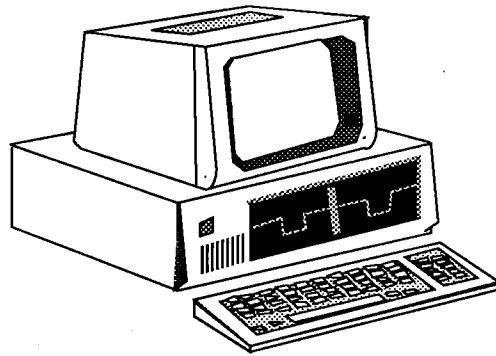
MASTER OF EDUCATION

LETHBRIDGE, ALBERTA

August, 1990

SINGLE COURSE PROJECT

ACCOUNTABILITY 2.0



SUBMITTED BY:

JOHN E. PERCEVAULT

MASTER TEACHER PROGRAM

UNIVERSITY OF LETHBRIDGE,

JUNE, 1989

For my father

**Thanks are extended to the
School Trustees, Administrators,
and Secretaries of Yellowhead
School Division No. 12. Without
the continued support, this
project would not be finished.**

**Thanks are extended to
Dr. M. Fisher, University of
Lethbridge, for his continued
assistance.**

Table of Contents

Section One - Introduction

Purchasing and Accounting - Operational Considerations	1
Past Practices and Agreed Limitations	1
New Directions	2
1987-88 Development and Implementation	2
1988-89 Coding Update	2
dBASE III Plus to dBASE IV Upgrade	3
School Level Apprehensions and Impressions	3
Hardware/Software Configuration - By School	4
Spin Off - Other dBASE Code Development and Implementation	5
Other Administration Applications	5
Classroom Integration	5
Rationale for Program	5

Section Two - Technical Reference Documentation

Loading and Main Menu	1
Data Entry - PO / Invoice / Close	2
Journal Listings (PO / Invoice)	4
S_NO Balances (Account, All)	5
Create / Edit Distributions	6
Utilities: Backup / Set up / Install / Restore	7
Program Map	8
Directory Listings	9
To do	10

Section Three - Program Listing

Menu	1
Dataent	2
Journals	27
S_no_bal	36
Tick_off	47
Allocate	48
Utility	57

Section Four - Sample Reports

Budget Distribution and Codes	1
Yellowhead Account Codes	2
Journal - Range Invoices to Yellowhead Account	3
School Budget Account Summaries	6
School Account - Details	7

SECTION ONE - INTRODUCTION



In response to a request by Mr. Klaus Puhmann, Superintendent of Schools, Yellowhead School Division No. 12, I developed the enclosed accounting package for system-wide use and for credit on my Master Teacher program at the University of Lethbridge.

Purchasing and Accounting - Operational Considerations in Yellowhead School Division No. 12

1. On authority of the school principal, purchase orders are initiated at the school level and forwarded to vendors. On receipt of goods, the school principal approves invoices for payment and invoices are forwarded to the central office accounting staff for payment.
2. All schools were implementing a manual synoptic journal to track expenditures by program area.

Examples: Administration, Art, Business Education

Each school in Yellowhead had a different set of budget areas to reflect the uniqueness of grade levels and program offering.

3. Yellowhead School Division is required to track expenditures in accordance with accounts defined by Alberta Education.

Examples: Services Contracted or Purchased, Textbooks, Supplies

An IBM System 34 minicomputer accounting system is maintained at the division central office. Invoices, approved for payment, are posted to Alberta Education accounts and monthly reports are forwarded to school administrators.

Past Practices and Agreed Limitations

1. The monthly central office report provided little useful information to a school administrator wanting financial information by program area. This report was dated and did not reflect new encumbrances initiated by the school.
2. A sequence of first recording purchase order transactions and subsequent updating of invoice values to program areas required a significant allocation of secretarial time. Projected department balances were difficult to obtain and the values were not always valid.

New Directions:

On evaluation of the effectiveness of the accounting systems implemented at both levels, it was decided to purchase or develop a microcomputer-based accounting model to reduce secretarial time and to provide easily obtainable budget summaries and reports.

The new system would meet the present organization needs without a major change to purchasing and accounting procedures. Schools would be able to track purchase orders and invoices to user-defined budget areas and account summaries would be provided in **real time**. Alberta Education account summaries were required at the school level to provide a reconciliation report to the monthly central office report.

Two schools previously piloted a microcomputer spreadsheet model for accounting, but reports were difficult to obtain, and the manipulation of a large worksheet was cumbersome. As a result, spreadsheets were rejected for system-wide adoption. Commercial accounting software packages were reviewed, but rejected for the lack of encumbrance processing, or for cost considerations.

On completion of Management Arts 3760, Topical Issues in Data Base Management, I offered to develop a program under a dBASE III Plus environment. This process allowed the development of a package to reflect the exact requirements in Yellowhead School Division.

1987-88 Development and Implementation

During the first three weeks in August, 1987, program specifications were completed and I wrote the first version of the program.

Four schools in Yellowhead had an installed office computer system and agreed to pilot the new program. During September and October, a number of program updates were completed to fix errors (primarily centering on error trapping). By January, 1988, six additional schools purchased office computers out of school capital funds and began using the accounting program.

In March, 1988, the division budget was approved, detailing the installation of an office microcomputer in the remaining Yellowhead schools and system-wide implementation began in September, 1988.

1988-89 Coding Update

After field testing the program for 18 months, the secretaries requested a significant rewrite to the data entry portion of the program to reduce data entry times, a tick-off option to assist central office reconciliation, a utilities section for forwarding to a new fiscal year, and a new routine for budget area calculations to reduce data processing times.

These revisions were completed in February, 1989 and distributed to schools along with an update from dBASE III Plus to dBASE IV.

This procedure, for updating code, allowed user input into design and significantly increased user ownership of the software. As a result, we have tentatively agreed to collect requested revisions and release a new update of this code approximately every 18 months. To assist in future program support and modification, our System 34 programmer is now training under the dBASE IV programming language and is expected to complete revisions and inservice under guidance. As our users are becoming more sophisticated, we plan to develop a record editor to allow direct access to the files. Our long range plans include inservice on, and use of the ASSIST program in dBASE.

Past inservice was provided on division inservice days and on-site support is regularly provided as requested. Technical assistance is provided by telephone support and the secretaries have developed an informal support telephone mechanism among themselves.

dBASE III Plus to dBASE IV Upgrade

dBASE was selected as the coding environment as it is the data base standard in the MSDOS world. Many courses are available to our staff on this software and this is an important consideration as staff changes.

The upgrade was purchased to ensure that our PC's are using the latest versions and that our applications are running in the most efficient manner. The dBASE IV upgrade allows the source code to be compiled, reducing data processing and input times.

School Level Apprehensions and Impressions

Most schools immediately abandoned their manual accounting systems, but two schools continued to maintain their manual books for a two-year period. As the administrators and secretaries in these schools were novices to computer use, they did not fully trust the new system, especially when the program intermittently crashed within the first few months. We now have 100 per cent adoption for the 1989-90 school year without manual books.

In our mid-sized primary schools, there was mixed reaction to the details provided by this accounting system. In these environments, it was discovered that they did not budget to program area and this was the real reason for the negative comments regarding automation. They now account by program areas.

Many of our administrators in our larger schools have stated that the success of this application alone has justified the cost of the hardware and software.

Hardware/Software Configuration - By School

- Parkland Composite (10 - 12): IBM AT, 640K, 30 MB hard drive, Okidata 2410P printer, dBASE IV
- Jubilee Junior (7 - 9): IBM AT, 640K, 30 MB hard drive, Panasonic 1595 printer, dBASE IV
- Pine Grove (4 - 6): Laser 286 AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- Westhaven (K - 6): Laser 286 AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- Evergreen (K - 3): Laser 286 AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- A. H. Dakin (K - 3): Commodore PC40 AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- Harry Collinge (8 - 12): Laser 286 AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- Mountain View (K - 7): IBM AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- Crescent Valley (K - 7): IBM AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- Roche Miette (4 - 6): Commodore PC40 AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- Fulham (K - 6): Laser 286 AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- Peers (K - 6): Laser 286 AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- Niton Central (K - 12): Commodore PC40 AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- Wildwood (K - 9): Commodore PC40 AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- Grand Trunk (8 - 12): Commodore PC40 AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV
- Evansview (K - 7): Laser 286 AT, 640K, 40 MB hard drive, Panasonic 1595 printer, dBASE IV

Spin Off - Other dBASE Code Development and Implementation

Based on our success with local coding of the encumbrance application, I completed a system-wide capital inventory/aging program in February, 1989. This program tracks capital inventory by school and location. Schools are now responsible for record maintenance and a compiled division data base is developed twice yearly for capital replacement calculations and for external audit purposes.

During June, 1989, I am completing an instructional materials fees assessment package to issue invoices, receipts, and past due notices. This program will access required student mailing information from our Columbia Students Records module and will provide a central data base for division-wide assessment of accounts in arrears.

Other Administration Applications

Our PCs were initially purchased for tracking school encumbrances, but the following application packages are used on a regular basis:

Word Processing - WordPerfect 4.2

Records Management - Columbia School System and Elementary School System (attendance, academic progress, student records)

Electronic Mail - Procomm 2.4.2 and Chairman BBS

Classroom Integration

Through the use of the Developers' Edition of dBASE IV, I have managed to legally integrate a run time version of this code as a mini unit in my Accounting 30 classes. Topics center on differentiating between the need requirements of financial accounting (debit/credit) and managerial encumbrance tracking to budget areas in public non-profit organizations.

Rationale for Program

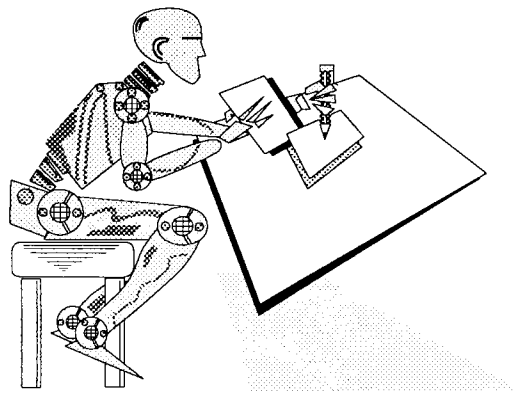
My rationale for program enrollment was to develop a broader background in MSDOS application software for business education integration. Through the development of Lotus 1-2-3 (1A) for Students, WordPerfect 4.2 for Students, and Accountability (Encumbrance Coding in dBASE IV), I feel I have met my personal objectives.

Respectfully Submitted,

John E. Percevault
Computer Program Coordinator
Yellowhead School Division No. 12

file = section one - project

SECTION TWO - TECHNICAL REFERENCE DOCUMENTATION



Loading AccountAbility

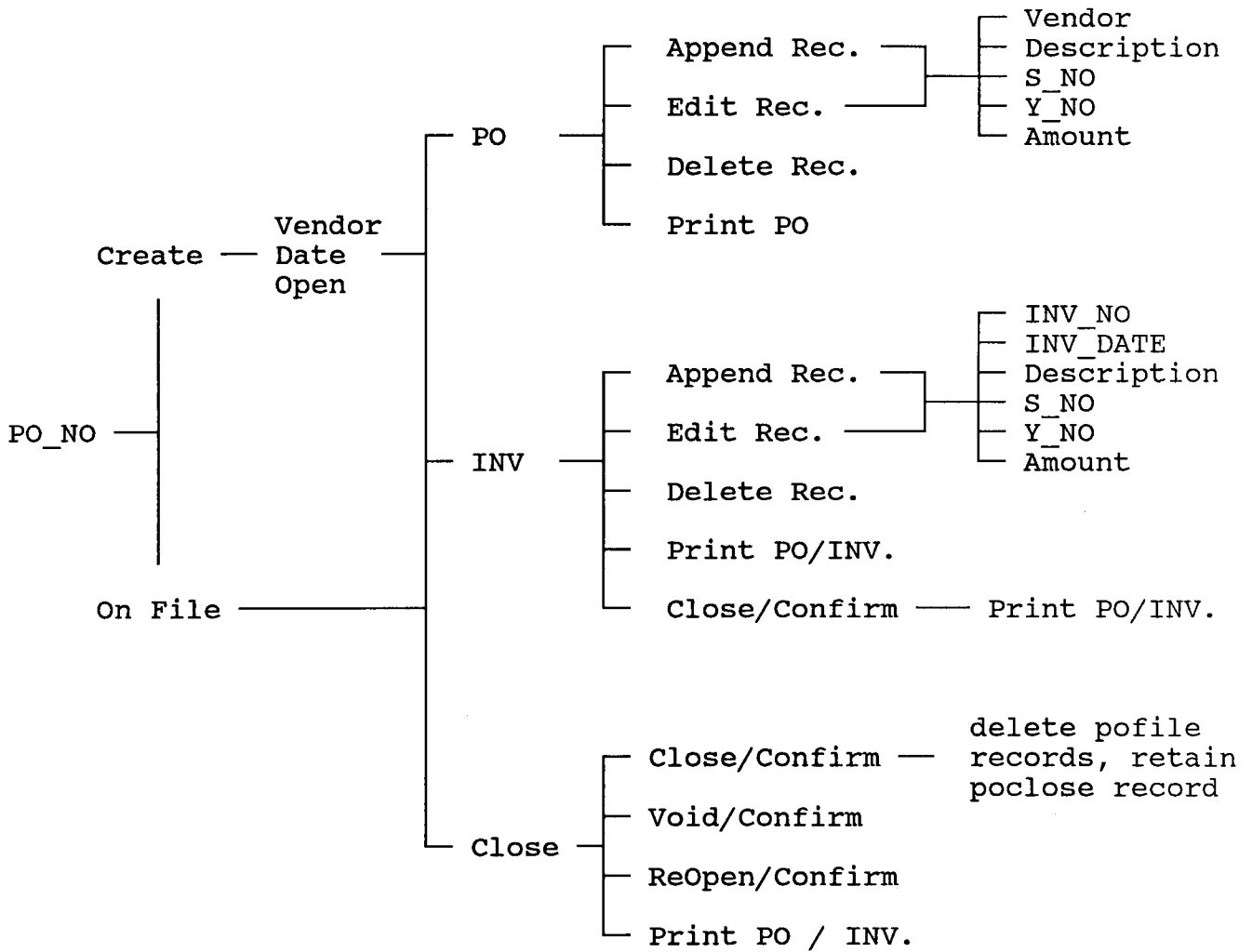
Select Accounts from the Myshell main menu. The following menu appears:

Accountability - Main Menu 2.0

- 1 - Data Entry - PO / Invoice / Close
- 2 - Journal Listings (PO, Invoice)
- 3 - S_NO Balances (Account, All)
- 4 - Reconciliation - Tick Off POs
- 5 - Create / Edit Distributions
- 6 - Utilities: Backup/Set up/Install/Restore
- 7 - Quit

Select 7

1 - Data Entry - PO / Invoice / Close



POCLOSE.DBF (cloindex.ndx, closeind.ndx)

Record#	PO_NO	Vendor	Date	Close	Tick_Off
1	620910	IBM	09/01/87	0	N

sys def data data entry sys def sys def
 entry character 15 clock open
 char 6 date 8 char 1

(The operator must manually determine when the PO is closed. If the PO is open, all PO values are used to project ending budget balances. If the PO is closed, all invoice values are used to project ending budget balances.)

POFILE.DBF (poindex.ndx, pos_no.ndx, poy_no.ndx)

Record#	PO_NO	Description	S_NO	Y_NO	Amount
1	620910	Laser Printer	2	65610	5500.00
2	620910	Toner	1	65315	165.00

sys def data data entry data data data entry
 entry character 15 entry entry numeric
 char 6 char 2 char 6 10.2

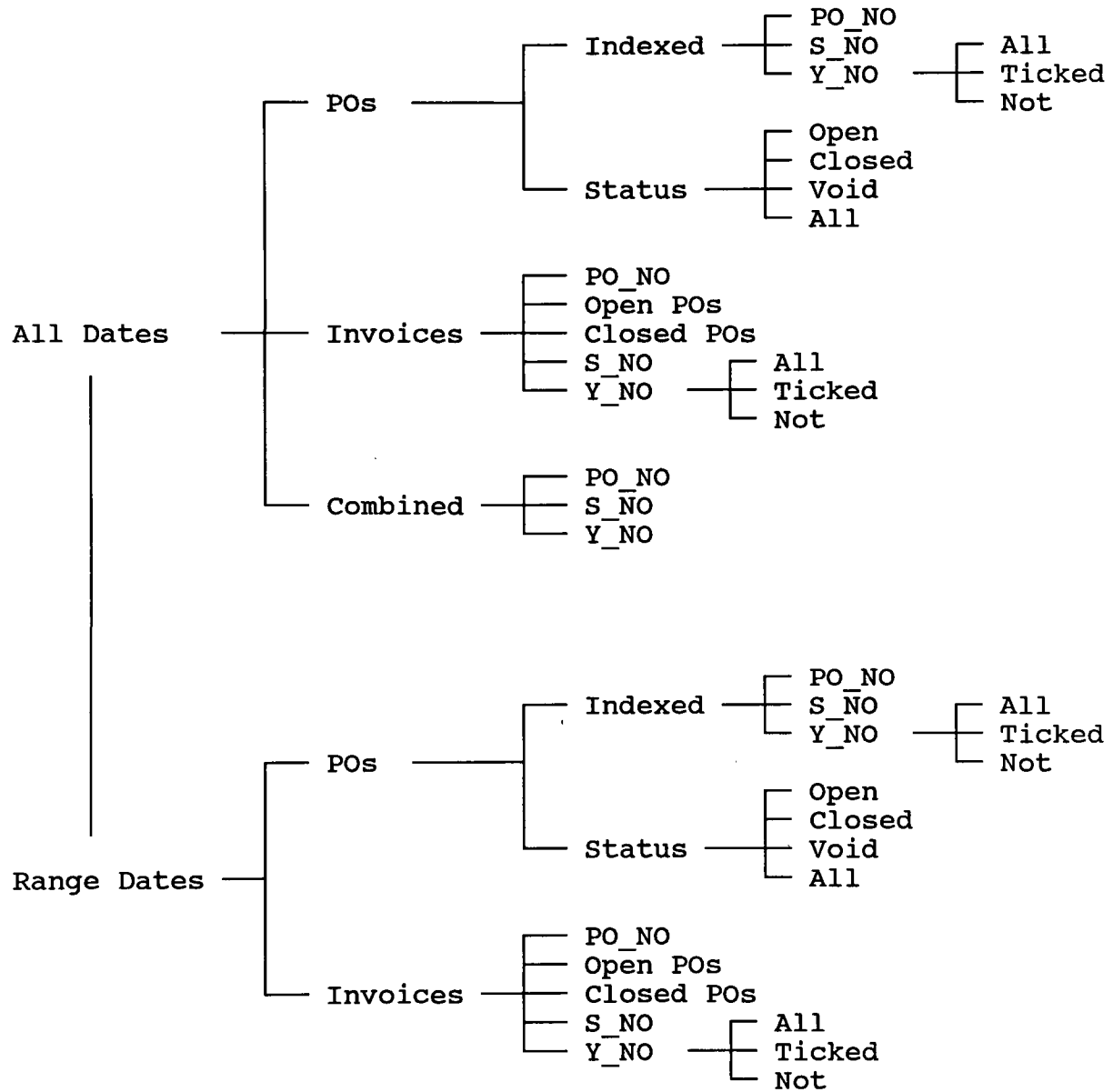
INVFILE.DBF (invindex.ndx, invs_no.ndx, invy_no.ndx)

Record#	PO_NO	INV_NO	INV_DATE	Description	S_NO	Y_NO	Amount
1	620910	I2346	09/22/87	Laser Printer	2	65610	5535.00
2	620915	456C12	09/23/87	Art Supplies	2	65315	210.00
3	620910	I2579	09/29/87	Toner	1	65315	173.00
4	620910	W345	09/30/87	Freight	2	65410	35.00

sys def data data sys def data entry data data data
 entry entry date 8 character 15 entry entry entry
 char 6 char 8 char 2 char 6 numeric
 check for legal 10.2
 accounts

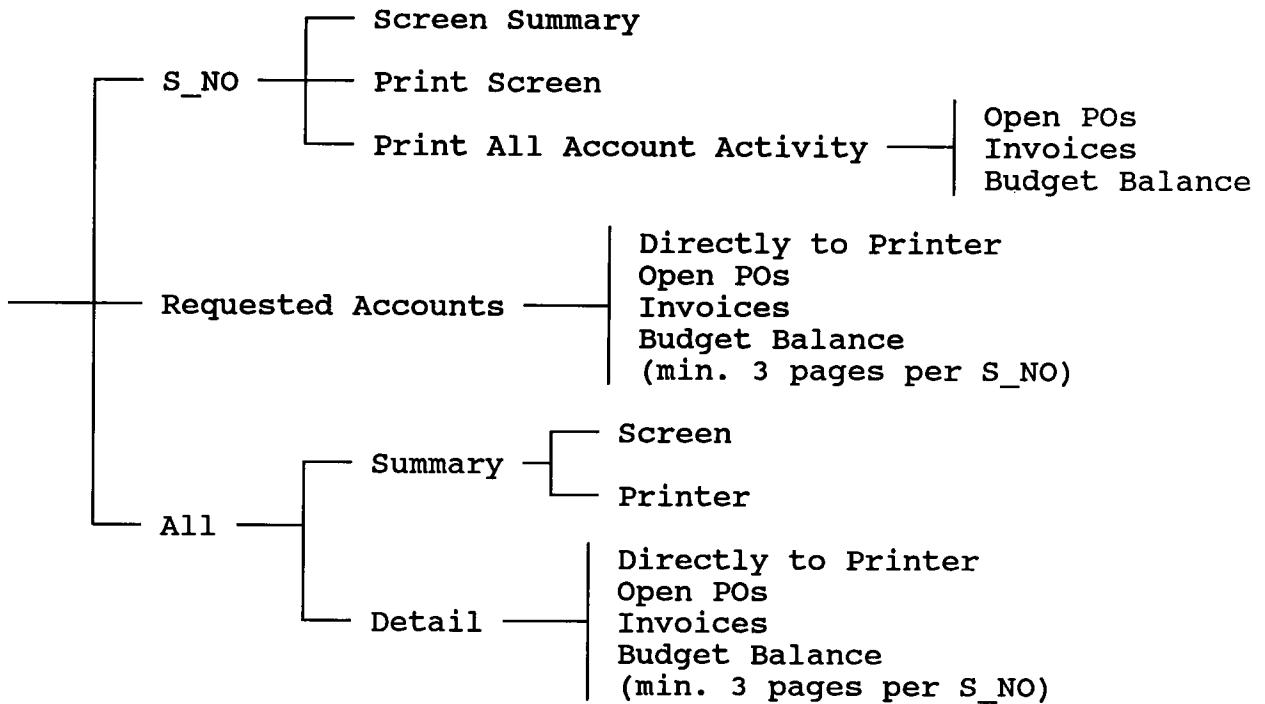
(the above records simulate three invoices to PO_NO 620910; two from the original vendor and one from the shipping company)

2 - Journal Listings (PO, Invoice)



(The American date format is used for date ranges: MM/DD/YY)

3 - S_NO Balances (Account, All)



S_NO_REQ.DBF (mdx tag S_NO) (work file for required accounts)

Record#	S_NO	Description

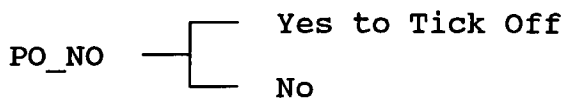
entry character 20
char 2

SUMFILE.DBF (work file - no .ndx or .mdx)

S_NO	Desc	Allocation	POTotal	INVTTotal	Commitment	Balance

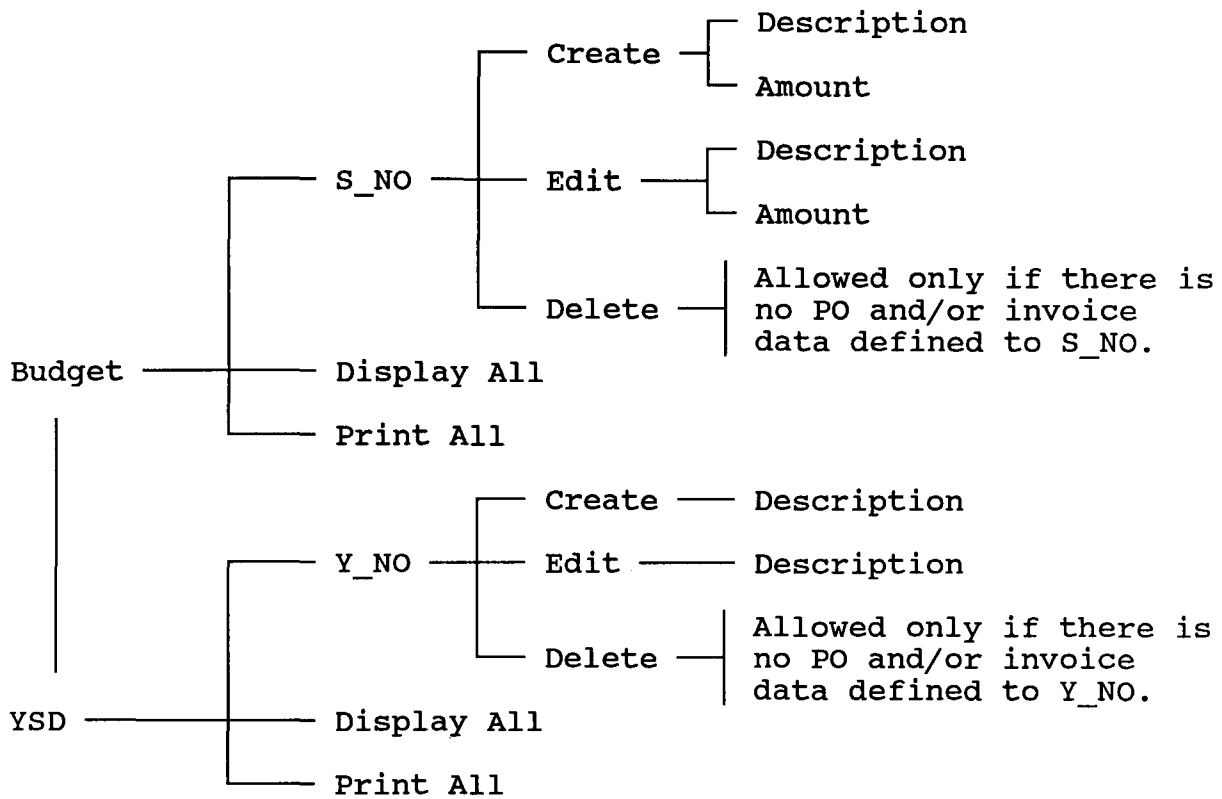
char char numeric numeric numeric numeric numeric
3 20 12.2 12.2 12.2 12.2 12.2

4 - Reconciliation - Tick Off POs



(Complete after reconciliation with central office report.)

5 - Create / Edit Distributions



BUDGET.DBF (budindex.ndx)

Record#	S_NO	Description	Allocate
1	1	Administration	4500.00
2	2	Business Ed.	5500.00

sys def data entry data entry data entry
 char 2 character 20 numeric
 12.2

YSDFILE.DBF (ysdindex.ndx)

Record#	Y_NO	Description
1	65610	Services

sys def data entry data entry
 char 2 character 20

6 - Utilities: Backup/Set up/Install/Restore

- (B)ackup current year
- (S)et up new fiscal year
- (I)nstall program updates
- (R)estore required fiscal year

Backup: Place a recently formatted disk, WITHOUT previous files in drive A. Press ENTER to continue.

(If you do not reformat the disk, dBASE will prompt to overwrite all files individually. Press the letter O to overwrite.)

Set up: You must complete a current backup before access is allowed. This option deletes all purchase order and invoice data, but your school accounts/budgets and Yellowhead accounts remain. Edit the school and Yellowhead accounts as required for the new year.

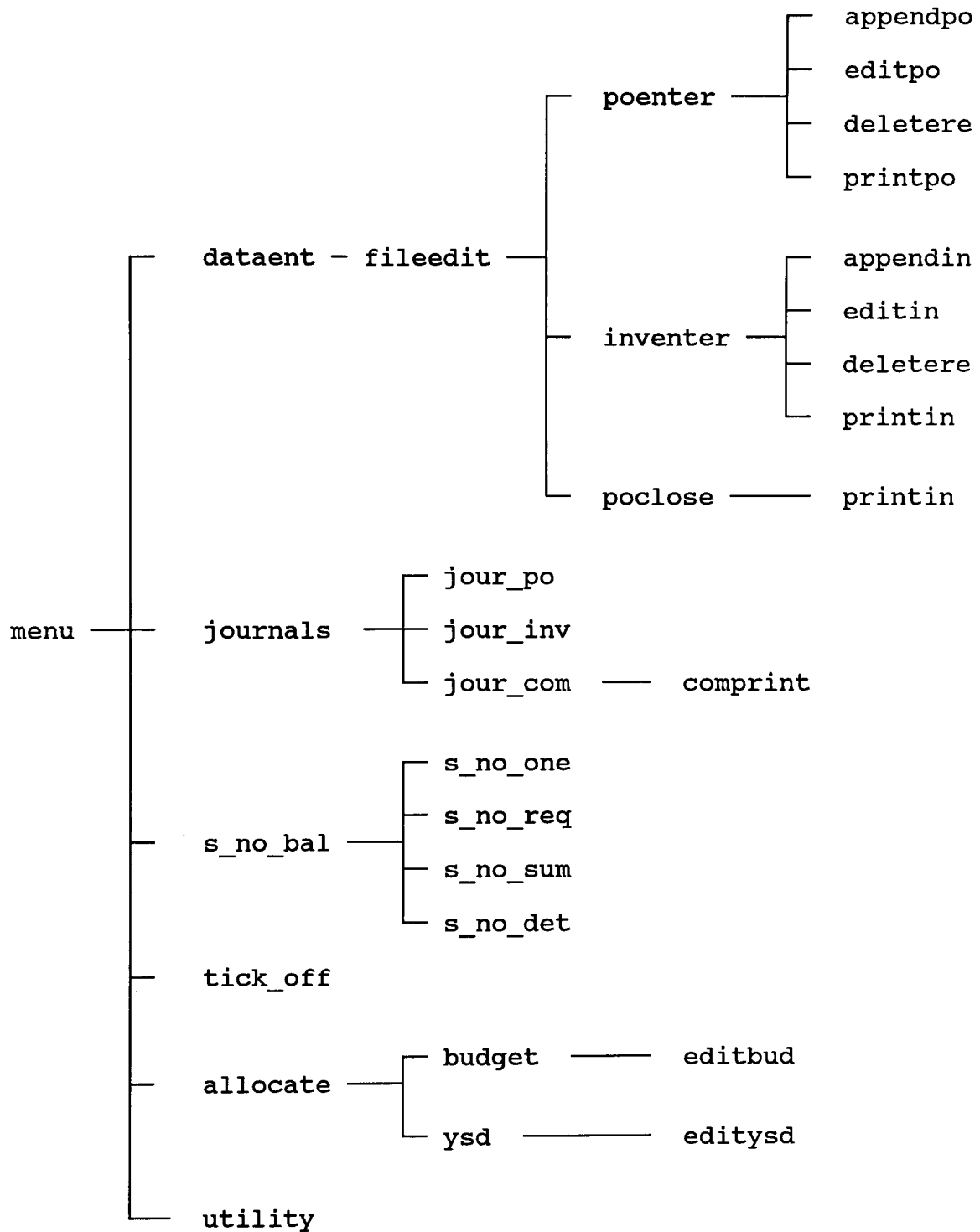
Install: Place the update program disk in drive A. Press ENTER to continue.

Four files are not updated by this process: CAPMENU.PRG, CAPMENU.DBO, UTILITY.PRG, and UTILITY.DBO. If updating these files, exit to DOS and copy the required files to the dBASE directory.

Restore: You must complete a current backup before access is allowed. Place the required fiscal year backup in drive A. Press ENTER to continue.

7 - Quit

Program Map



Directory Listings:

Data Base Files:

INVFILE.DBF	YSDFILE.DBF	ALLPO.DBF
POCLOSE.DBF	POFILE.DBF	BUDGET.DBF
SUMFILE.DBF	S_NO_REQ.DBF	ALLINV.DBF

Multiple Index Files:

ALLPO.MDX	ALLINV.MDX	POFILE.MDX
POCLOSE.MDX	INVFILE.MDX	S_NO_REQ.MDX

Single Index Files:

BUDINDEX.NDX	YSDINDEX.NDX
--------------	--------------

Program Files (source code *.PRG, object/compiled code *.DBO):

S_NO_DET.PRG	POENTER.PRG	APPENDIN.PRG
APPENDPO.PRG	POCLOSE.PRG	S_NO_SUM.PRG
BUDGET.PRG	EDITIN.PRG	EDITPO.PRG
CLEAROUT.PRG	YSD.PRG	MENU.PRG
ALLOCATE.PRG	EDITYSD.PRG	EDITBUD.PRG
JOUR_PO.PRG	JOURNALS.PRG	INVENTER.PRG
JOUR_INV.PRG	S_NO_BAL.PRG	S_NO_ONE.PRG
PRINTPO.PRG	UPDATEIN.PRG	JOUR_COM.PRG
COMPRINT.PRG	DELETERE.PRG	PRINTIN.PRG
UTILITY.PRG	DATAENT.PRG	FILEEDIT.PRG
S_NO_REQ.PRG	TICK_OFF	

Report Design Files (report generator *.FRM, object/compiled code *.FRO, source code *.FRG):

ALLPOS_.FRM	ALLOPEN.FRM	ALLCLO.FRM
ALLPOY_.FRM	ALLPOOPE.FRM	ALLPOCLO.FRM
ALLOP.FRM	ALLPOVOI.FRM	ALLPOSTA.FRM
ALLINV.FRM	ALLINVS_.FRM	ALLINVY_.FRM

Install/Upgrade Utilities:

CLEAROUT.PRG - deletes all records in all files
UPDATEIN - changes index files to tag multiple index files
TICK_FIX - writes N to new tick_field for all records

To do:

1. additional check: if no YSD and budget accounts, access denied to data entry, journals, and balances programs.
2. use poclose order cloindex
modify structure
tick_off character 1
do tick_fix

c:\wp\account1.jp

SECTION THREE - PROGRAM LISTING



```
close all
set device to screen
set decimals to 2
set talk off
set bell off
clear
@2,28 say 'AccountAbility 2.0'
@6,20 say 'Reindexing Data Base -- Please Wait'
select 5
use ysdfile index ysdindex
pack
use
select 4
use budget index budindex
pack
use
select 3
use invfile order invindex
pack
use
select 2
use poclose order cloindex
pack
use
select 1
use pofile order poindex
pack
use
store '7' to response
store .F. to exit
do while .not. exit
    clear
    @2,23 say 'AccountAbility 2.0 - Main Menu'
    @5,20 say ' 1 - Data Entry - PO / Invoice / Close'
    @7,20 say ' 2 - Journal Listings (PO, Invoice)'
    @9,20 say ' 3 - S_NO Balances (Account, All)'
    @11,20 say ' 4 - Reconciliation - Tick Off POs'
    @13,20 say ' 5 - Create / Edit Distributions'
    @15,20 say ' 6 - Utilities: Backup/Set up/Install/Restore'
    @17,20 say ' 7 - Quit'
    @20,25 say ' Enter Selection Number:' get response picture '9'
    read
    do case
        case response = '1'
            do dataent
        case response = '2'
            do journals
        case response = '3'
            do s_no_bal
        case response = '4'
            do tick_off
        case response = '5'
            do allocate
        case response = '6'
            do utility
        case response = '7'
            close all
            quit
    endcase
    store '7' to response
```



```

close all
select 3
use invfile order invindex
select 2
use poclose order cloindex
select 1
use pofile order poindex
store 'A' to another
do while another = 'A'
store 'R' to retry
  do while retry = 'R'
    select 2
    store 'Q' to response
    clear
    @2,10 say 'AccountAbility 2.0'
    @5,10 say '1 - Data Entry - PO / Invoice / Close (dataent.prg)'
    @10,10 say 'Enter PO_NO, (Q)uit' get response picture '999999'
    read
    store trim(response) to response
    store len(response) to length
    if length < 6 .and. right(response,1) <> 'Q'
      @12,10 say 'Invalid PO_NO, (R)etry, or any key to menu' get retry picture 'X'
      read
      if UPPER(retry) <> 'R'
        close all
        return
      endif
    endif
    if right(response,1) = 'Q'
      close all
      return
    endif
    if length = 6
      find &response
      if .not. found()
        store 'C' to create
        store 'Q' to mvendor
        @13,10 say 'PO_NO'
        @13,18 say response
        @13,25 say 'is not on file.'
        @16,10 say '(C)reate, or any key to menu' get create picture 'X'
        read
        if UPPER(create) <> 'C'
          exit
        endif
        @19,10 say 'Enter Vendor Name, (Q)uit' get mvendor picture 'XXXXXXXXXXXXXXXXX'
        read
        if UPPER(mvendor) = 'Q'
          exit
        else
          select 2
          append blank
          replace po_no with response
          replace vendor with mvendor
          replace date with date()
          replace close with 'Q'
          replace tick_off with 'N'
          do fileedit
          endif
        endif
      endif
    endif
  endif

```

endif

if found()
do fileedit
endif

enddo
enddo

```

store 'A' to openclose
do while openclose = 'A'
  if close = '0'
    store 'A' to enteragain
    do while enteragain = 'A'
      clear
      select 3
      count to invnum for po_no = response
      if invnum > 0
        select 1
        count to ponum for po_no = response
        if ponum = 0
          @7,10 say 'This PD_NO has invoice records but no purchase order records.'
          @9,10 say 'Be sure to close the PO if you wish the invoice records'
          @10,10 say 'to be included in the calculation for department balances.'
        else
          @7,10 say 'This PD_NO has invoice records and purchase order records.'
          @9,10 say 'Be sure to close the PO if you wish the invoice records'
          @10,10 say 'to be included in the calculation for department balances.'
        endif
      else
        select 1
        count to ponum for po_no = response
        if ponum = 0
          @7,10 say 'This PD_NO has no purchase order or invoice records.'
        else
          @7,10 say 'This PD_NO has purchase order records but no invoice records.'
        endif
      endif
    select 1
    store 'Q' to noentries
    @2,24 say 'AccountAbility 2.0'
    @13,10 say 'PD_NO'
    @13,18 say response
    @13,25 say 'is currently open.'
    @16,10 say '(P)D Entries, (I)nvoice Entries, (C)lose/Void PD, (Q)uit' get noentries picture 'X'
    read
    store UPPER(noentries) to noentries
    if noentries = 'Q' .and. b->close = '0'
      select 2
      store ' ' to enteragain
      store ' ' to openclose
    endif
    if noentries = 'P'
      do poenter
    endif
    if noentries = 'I'
      do inventer
      if b->close = 'C'
        exit
      endif
    endif
    if noentries = 'C'
      do poclose
      if b->close = 'C' .or. b->close = 'V'
        exit
      endif
    endif
  endif
enddo
endif

```

```
        if b->close = 'C' .or. b->close = 'V'
            store 'Q' to noopen
            @13,10 say 'PO_NO'
            @13,18 say response
            @13,25 say 'is currently'
            if b->close = 'C'
                @13,38 say 'closed.'
            endif
            if b->close = 'V'
                @13,38 say 'void.'
            endif
            @16,10 say 're(O)pen, (Q)uit' get noopen picture 'X'
            read
            store UPPER(noopen) to noopen
            if noopen = 'Q'
                exit
            endif
            if noopen = 'O'
                do poclose
            endif
        endif
    enddo
```

```
store 'E' to recordsedit
do while recordsedit = 'E'
  store 'Q' to editrecords
  select 2
  store 0 to responseval
  responseval = po_no
  store '      ' to mvendor
  store '      ' to mdate
  mdate = date
  mvendor = vendor
  select 1
  clear
  @1,0 say 'PD DATA ON FILE:'
  count to number for a->po_no = responseval
  if number > 6
    @10,10 say 'Records for PD exceed 6, select (P)rint for Details'
  else
    display for a->po_no = responseval
  endif
  sum to total for a->po_no = responseval
  @13,0 say 'Vendor:'
  @13,10 say mvendor
  @13,26 say 'PD Total:'
  @13,35 say total picture '99999999.99'
  @13,55 say 'PD Date:'
  @13,65 say mdate
@17,1 say '(A)ppend Record, (E)dit Record, (D)delete Record, (P)rint PD, (Q)uit' get editrecords picture 'X'
read
  store UPPER(editrecords) to editrecords
  if editrecords = 'Q'
    clear
    exit
  endif
  if editrecords = 'E'
    do editpo
  endif
  if editrecords = 'A'
    do appendpo
  endif
  if editrecords = 'D'
    do deletere
  endif
  if editrecords = 'P'
    do printpo
  endif
enddo
```

```

clear
store space(69) to clearline
store '      ' to mdesc
store 0 to mamt
@3,25 say 'Append Record to Purchase Order'
@6,5 say 'Vendor:'
@6,15 say mvendor
@6,35 say 'PO_NO: '
@6,43 say response
@8,5 say 'Description: ' get mdesc picture 'XXXXXXXXXXXXXXXXX'
read
store 'A' to others_no
do while others_no = 'A'
store ' ' to ms_no
@8,40 say 'S_NO: ' get ms_no picture 'XX'
read
if ms_no = ' '
    store ' ' to ms_no
else
select 6
use budget index budindex
find &ms_no
if found()
store '      ' to sdesc
sdesc = desc
@10,40 say sdesc
store 'A' to othersy_no
do while othersy_no = 'A'
store '      ' to my_no
@8,60 say 'Y_NO: ' get my_no picture 'XXXXXX'
read
if my_no = '      '
    store '      ' to my_no
else
select 6
use ysdfile index ysdindex
find &my_no
if found()
store '      ' to ydesc
ydesc = desc
@10,60 say ydesc
use
@12,5 say 'Amount:' get mamt picture '99999999.99'
read
store 'C' to confirm
@15,10 say clearline
@15,10 say '(C)onfirm or any key to menu' get confirm picture 'X'
read
if UPPER(confirm) <> 'C'
    use
    return
else
select 1
append blank
replace a->po_no with response
replace a->desc with mdesc
replace a->s_no with ms_no
replace a->y_no with my_no
replace a->amt with mamt
return

```

```
        endif
endif
    if .not. found()
        store 'R' to redo
        @15,10 say clearline
        @15,10 say 'Y_NO:'
        @15,18 say my_no
        @15,25 say 'is not on file.'
        @17,10 say '(R)edo, or any key to menu' get redo picture 'x'
        read
        if UPPER(redo) <> 'R'
            use
            return
        endif
        @15,10 say clearline
        @17,10 say clearline
    endif
endif
enddo
endif
if .not. found()
    store 'R' to redo
    @15,10 say clearline
    @15,10 say 'S_NO:'
    @15,18 say ms_no
    @15,25 say 'is not on file.'
    @17,10 say '(R)edo, or any key to menu' get redo picture 'X'
    read
    if UPPER(redo) <> 'R'
        use
        return
    endif
    @15,10 say clearline
    @17,10 say clearline
endif
endif
enddo
```

```

store 'A' to anothertry
do while anothertry = 'A'
store space(69) to clearline
store 0 to test
@20,10 say clearline
@20,20 say 'Enter Record#: ' get test picture '999999'
read
select 1
count to counttest
  if test > counttest .OR. test = 0
    store 'R' to retry
    @20,10 say clearline
@20,10 say 'Invalid Record# for PD File -- (R)etry, or any key to menu' get retry picture 'X'
    read
    if retry <> 'R'
      return
    endif
  endif
endif

if test <= counttest .AND. test <> 0

  goto test
  if po_no <> responseval
    store 'R' to retry
    @20,10 say clearline
@20,10 say 'Invalid Record# for PD_NO -- (R)etry, or any key to menu' get retry picture 'X'
    read
    if retry <> 'R'
      return
    endif
  endif
endif

if po_no = responseval
store 'E' to editingpo
do while editingpo = 'E'
store 'Q' to editor
mvendor = b->vendor
clear
@3,25 say 'Edit Record to Purchase Order'
@8,0 say 'Vendor:'
@8,10 say mvendor
@9,10 say ' '
display
@15,5 say '(V)endor, (D)escription, (S)_NO, (Y)_NO, (A)mount, (Q)uit' get editor picture 'X'
read
@15,5 say clearline
store UPPER(editor) to editor
if editor = 'Q'
  return
endif

if editor = 'V'
  store ' ' to newvendor
  @15,10 say 'New Vendor:' get newvendor picture 'XXXXXXXXXXXXXXXXX'
  read
  replace b->vendor with newvendor
  @15,10 say clearline
endif

if editor = 'D'

```



```

store '          ' to newdesc
@15,10 say 'New Description:' get newdesc picture 'XXXXXXXXXXXXXXXXX'
read
replace a->desc with newdesc
@15,10 say clearline
endif

```

```

if editor = 'S'
store 'A' to anothers_no
do while anothers_no = 'A'
store ' ' to news_no
@15,10 say 'New S_NO: ' get news_no picture 'XX'
read
if news_no = ' '
store 'A' to anothers_no
else
select 6
use budget index budindex
find &news_no
if found()
use
select 1
replace a->s_no with news_no
exit
endif
if .not. found()
store 'R' to redo
@15,10 say clearline
@15,10 say 'S_NO:'
@15,18 say news_no
@15,25 say 'is not on file.'
@17,10 say '(R)edo, or any key to menu' get redo picture 'X'
read
if UPPER(redo) <> 'R'
use
select 1
exit
endif
@15,10 say clearline
@17,10 say clearline
endif
endif
enddo
endif

```

```

if editor = 'Y'
store 'A' to anothersy_no
do while anothersy_no = 'A'
store ' ' to newsy_no
@15,10 say 'New Y_NO: ' get newsy_no picture 'XXXXXX'
read
if newsy_no = ' '
store ' ' to newsy_no
else
select 6
use ysdfile index ysdindex
find &newsy_no
if found()
use
select 1

```

```
        replace a->y_no with newy_no
        exit
    endif
    if .not. found()
        store 'R' to redo
        @15,10 say clearline
        @15,10 say 'Y_NO: '
        @15,18 say newy_no
        @15,25 say 'is not on file.'
        @17,10 say '(R)edo, or any key to menu' get redo picture 'X'
        read
        if UPPER(redo) <> 'R'
            use
            select 1
            exit
        endif
        @15,10 say clearline
        @17,10 say clearline
    endif
endif
enddo
endif

if editor = 'A'
    store 0 to newamt
    @15,10 say 'New Amount: ' get newamt picture '99999999.99'
    read
    replace a->amt with newamt
    @15,10 say clearline
endif
enddo
endif
endif
enddo
```

```

*deleter.prg delete specified records to defined po
*copyright Aug 1987, John E. Percevault, Box 2236 - Edson, AB - T0E 0P0
store 'A' to anothertry
do while anothertry = 'A'
store space(69) to clearline
store 0 to test
@21,10 say clearline
@21,20 say 'Enter Record#: ' get test picture '999999'
read
if noentries = 'P'
  select 1
endif
if noentries = 'I'
  select 3
endif
count to counttest
  if test > counttest .OR. test = 0
    store 'R' to retry
    @21,10 say clearline
@21,10 say 'Invalid Record# for File -- (R)etry, or any key to menu' get retry picture 'X'
    read
    if retry <> 'R'
      return
    endif
  endif
endif

if test <= counttest .AND. test <> 0

  goto test
  if po_no <> responseval
    store 'R' to retry
    @21,10 say clearline
@21,10 say 'Invalid Record# for PD_NO -- (R)etry, or any key to menu' get retry picture 'X'
    read
    if retry <> 'R'
      return
    endif
  endif
endif

if po_no = responseval
  store 'C' to deleteyes
  mvendor = b->vendor
  clear
  @3,25 say 'Delete Record'
  @8,0 say 'Vendor:'
  @8,10 say mvendor
  @9,10 say ' '
  display
  @15,5 say '(C)onfirm, or any key to menu' get deleteyes picture 'X'
  read
  @15,5 say clearline
  if UPPER(deleteyes) <> 'C'
    return
  else
    delete
    pack
    return
  endif
endif
endif
endif

```

*printpo.prg prints selected po activity from poenter/edit program

```
select 2
  set device to printer
  set margin to 5
  @6,15 say 'Individual Purchase Order'
  @6,55 say date()
  @9,10 say '  '
  store 0 to linecnt
  display off to printer
  linecnt = linecnt + 14
select 1
  find &responseval
  if found()
    @linecnt,1 say 'Purchase Order Details:'
    linecnt = linecnt + 1
    @linecnt,1 say '  '
    display DESC, S_NO, Y_NO, AMT to printer for po_no = responseval
    store 0 to mpototal
    sum to mpototal for po_no = responseval
    count to mporecs for po_no = responseval
    linecnt = linecnt + mporecs + 3
    @ linecnt,34 say 'Purchase Order Total:'
    @ linecnt,54 say mpototal picture '99999999.99'
    linecnt = linecnt + 3
  endif
  set margin to 0
  set device to screen
  eject
return
```

```

store 'A' to recordsedit
do while recordsedit = 'A'
  store 'Q' to editrecords
  select 2
  store 0 to responseval
  responseval = po_no
  store '      ' to mvendor
  store '      ' to mdate
  mvendor = vendor
  mdate = date
  select 3
  clear
  @1,0 say 'INVOICE DATA ON FILE:'
  count to invnumber for po_no = response
  if invnumber > 6
    @4,5 say 'Invoice Records for PO exceed 6, select (P)rint for details'
  else
    display for po_no = response
  endif
  sum to invtotal for c->po_no = response
  select 1
  @9,0 say 'PO DATA ON FILE:'
  count to ponumber for po_no = response
  if ponumber > 6
    @12,5 say 'Purchase Order Records for PO exceed 6, select (P)rint for details'
  else
    display off for po_no = response
  endif
  sum to pototal for a->po_no = response
  @17,0 say 'Vendor:'
  @17,10 say mvendor
  @17,25 say 'INV Total:'
  @17,35 say invtotal picture '99999999.99'
  @17,56 say 'PO Total:'
  @17,65 say pototal picture '99999999.99'
@19,1 say '(A)ppend Rec., (E)dit Rec., (D)elete Rec., (P)rint INV, (C)lose PO, (Q)uit' get editrecords picture 'X'
  read
  store UPPER(editrecords) to editrecords
  if editrecords = 'Q'
    clear
    exit
  endif
  if editrecords = 'C'
    store 'C' to confirm
    select 2
    replace b->close with 'C'
    store space(78) to clearline
    @19,1 say clearline
    @19,15 say '(C)onfirm to close PO, or any key' get confirm picture 'X'
    read
    if UPPER(confirm) <> 'C'
      replace b->close with 'O'
    else
      store 'P' to inprint
      @19,1 say clearline
      @19,15 say '(P)rint, or any key' get inprint picture 'X'
      read
      if UPPER(inprint) = 'P'
        do printin
      endif
    endif
  endif

```

```
                clear
                exit
            endif
        endif
        if editrecords = 'A'
            do appendin
        endif
        if editrecords = 'E'
            do editin
        endif
        if editrecords = 'D'
            do deletere
        endif

        if editrecords = 'P'
            do printin
        endif
    enddo
```

```

clear
store space(69) to clearline
store ' ' to minv_no
store ' ' to minv_date
minv_date = date()
store ' ' to mdesc
store 0 to mant
@3,25 say 'Append Record to Invoice'
@6,10 say 'Vendor:'
@6,20 say mvendor
@6,40 say 'PO_NO: '
@6,48 say responseval
@8,10 say 'INV_NO: ' get minv_no picture 'XXXXXXXX'
read
@8,40 say 'ENTRY DATE: '
@8,53 say minv_date
@10,5 say 'Description: ' get mdesc picture 'XXXXXXXXXXXXXXXXX'
read
store 'A' to others_no
do while others_no = 'A'
store ' ' to ms_no
@10,40 say 'S_NO: ' get ms_no picture 'XX'
read
if ms_no = ' '
store ' ' to ms_no
else
select 6
use budget index budindex
find &ms_no
if found()
store ' ' to sdesc
sdesc = desc
@12,40 say sdesc
store 'A' to another_no
do while another_no = 'A'
store ' ' to my_no
@10,60 say 'Y_NO: ' get my_no picture 'XXXXXX'
read
if my_no = ' '
store ' ' to my_no
else
select 6
use ysdfile index ysdindex
find &my_no
if found()
store ' ' to ydesc
ydesc = desc
@12,60 say ydesc
use
@14,10 say 'Amount: ' get mant picture '99999999.99'
read
store 'C' to confirm
@17,10 say clearline
@17,10 say '(C)confirm or any key to menu' get confirm picture 'X'
read
if UPPER(confirm) <> 'C'
use
return
else
select 3

```

```
        append blank
        replace c->po_no with response
        replace c->inv_no with minv_no
        replace c->inv_date with minv_date
        replace c->desc with mdesc
        replace c->s_no with ms_no
        replace c->y_no with my_no
        replace c->amt with mamt
        return
    endif
endif

if .not. found()
    store 'R' to redo
    @17,10 say clearline
    @17,10 say 'Y_NO:'
    @17,18 say my_no
    @17,25 say 'is not on file.'
    @19,10 say '(R)edo, or any key to menu' get redo picture 'x'
    read
        if UPPER(redo) <> 'R'
            use
            return
        endif
    @17,10 say clearline
    @19,10 say clearline
endif
endif
enddo

endif

if .not. found()
    store 'R' to redo
    @17,10 say clearline
    @17,10 say 'S_NO:'
    @17,18 say ms_no
    @17,25 say 'is not on file.'
    @19,10 say '(R)edo, or any key to menu' get redo picture 'X'
    read
        if UPPER(redo) <> 'R'
            use
            return
        endif
    @17,10 say clearline
    @19,10 say clearline
endif

endif
enddo
```



```

store 'A' to anothertry
do while anothertry = 'A'
store space(71) to clearline
store 0 to test
@20,10 say clearline
@20,20 say 'Enter Record#: ' get test picture '999999'
read
select 3
count to counttest
  if test > counttest .OR. test = 0
    store 'R' to retry
    @20,10 say clearline
@20,10 say 'Invalid Record# for INV File -- (R)etry, or any key to menu' get retry picture 'X'
    read
    if retry <> 'R'
      return
    endif
  endif

if test <= counttest .AND. test <> 0

  goto test
  if po_no <> responseval
    store 'R' to retry
    @20,10 say clearline
@20,10 say 'Invalid Record# for required PO_NO -- (R)etry, or any key to menu' get retry picture 'X'
    read
    if retry <> 'R'
      return
    endif
  endif

  if po_no = responseval
    store 'E' to editingpo
    do while editingpo = 'E'
      store 'Q' to editor
      mvendor = b->vendor
      clear
      @3,25 say 'Edit Record to Invoice'
      @8,0 say 'Vendor:'
      @8,10 say mvendor
      @9,10 say ' '
      display
@15,5 say 'Inv_(N)o, (I)nv_Date, (D)escription, (S)_NO, (Y)_NO, (A)mount, (Q)uit' get editor picture 'X'
      read
      @15,5 say clearline
      store UPPER(editor) to editor
      if editor = 'Q'
        return
      endif

      if editor = 'N'
        store ' ' to newinv_no
        @15,10 say 'New INV_NO:' get newinv_no picture 'XXXXXXXXXXXXXX'
        read
        replace c->inv_no with newinv_no
        @15,10 say clearline
      endif

      if editor = 'I'

```

```

store ' ' to newdate
@15,10 say 'New INV_Date:' get newdate picture '@d 99/99/99'
read
store CTOD(newdate) to newdate
replace c->inv_date with newdate
@15,10 say clearline
endif

if editor = 'D'
store ' ' to newdesc
@15,10 say 'New Description:' get newdesc picture 'XXXXXXXXXXXXXXXXX'
read
replace c->desc with newdesc
@15,10 say clearline
endif

if editor = 'S'
store 'A' to anothers_no
do while anothers_no = 'A'
store ' ' to news_no
@15,10 say 'New S_NO: ' get news_no picture 'XX'
read
if news_no = ' '
store ' ' to news_no
else
select 6
use budget index budindex
find &news_no
if found()
use
select 3
replace c->s_no with news_no
exit
endif
if .not. found()
store 'R' to redo
@15,10 say clearline
@15,10 say 'S_NO:'
@15,18 say news_no
@15,25 say 'is not on file.'
@17,10 say '(R)edo, or any key to menu' get redo picture 'X'
read
if UPPER(redo) <> 'R'
use
select 3
exit
endif
@15,10 say clearline
@17,10 say clearline
endif
endif
enddo
endif

if editor = 'Y'
store 'A' to anothers_no
do while anothers_no = 'A'
store ' ' to newy_no
@15,10 say 'New Y_NO: ' get newy_no picture 'XXXXXX'
read

```

```

if newy_no = '      '
  store '      ' to newy_no
else
  select 6
  use ysdfile index ysdindex
  find &newy_no
  if found()
    use
    select 3
    replace c->y_no with newy_no
    exit
  endif
  if .not. found()
    store 'R' to redo
    @15,10 say clearline
    @15,10 say 'Y_NO: '
    @15,18 say newy_no
    @15,25 say 'is not on file.'
    @17,10 say '(R)edo, or any key to menu' get redo picture 'X'
    read
    if UPPER(redo) <> 'R'
      use
      select 3
      exit
    endif
    @15,10 say clearline
    @17,10 say clearline
  endif
endif
enddo

if editor = 'A'
  store 0 to newamt
  @15,10 say 'New Amount: ' get newamt picture '99999999.99'
  read
  replace c->amt with newamt
  @15,10 say clearline
endif
enddo
endif
enddo

```

```

*deletere.prg delete specified records to defined po
*copyright Aug 1987, John E. Percevault, Box 2236 - Edson, AB - T0E 0P0
store 'A' to anothertry
do while anothertry = 'A'
store space(69) to clearline
store 0 to test
@21,10 say clearline
@21,20 say 'Enter Record#: ' get test picture '999999'
read
if noentries = 'P'
  select 1
endif
if noentries = 'I'
  select 3
endif
count to counttest
  if test > counttest .OR. test = 0
    store 'R' to retry
    @21,10 say clearline
@21,10 say 'Invalid Record# for File -- (R)etry, or any key to menu' get retry picture 'X'
    read
    if retry <> 'R'
      return
    endif
  endif
endif

if test <= counttest .AND. test <> 0

  goto test
  if po_no <> responseval
    store 'R' to retry
    @21,10 say clearline
@21,10 say 'Invalid Record# for PO_NO -- (R)etry, or any key to menu' get retry picture 'X'
    read
    if retry <> 'R'
      return
    endif
  endif

  if po_no = responseval
    store 'C' to deleteyes
    mvendor = b->vendor
    clear
    @3,25 say 'Delete Record'
    @8,0 say 'Vendor:'
    @8,10 say mvendor
    @9,10 say ' '
    display
    @15,5 say '(C)onfirm, or any key to menu' get deleteyes picture 'X'
    read
    @15,5 say clearline
    if UPPER(deleteyes) <> 'C'
      return
    else
      delete
      pack
      return
    endif
  endif
endif
endif

```

deleter.prg 05/24/89

2

enddo

```
*printin.prg prints po/inv activity by specified po_no
```

```
select 2
set device to printer
set margin to 5
@6,10 say 'Invoice / Purchase Order Report'
@6,55 say date()
@9,10 say ' '

store 0 to linecnt
display off to printer
linecnt = linecnt + 14

select 1
find &responseval
if found()
  @linecnt,1 say 'Purchase Order Details:'
  linecnt = linecnt + 1
  @linecnt,1 say ' '
  display off DESC, S_NO, Y_NO, AMT to printer for po_no = responseval
  store 0 to mpototal
  sum to mpototal for po_no = responseval
  count to mporecs for po_no = responseval
  linecnt = linecnt + mporecs + 3
  @ linecnt,34 say 'Purchase Order Total:'
  @ linecnt,54 say mpototal picture '99999999.99'
  linecnt = linecnt + 3
endif

select 3
find &responseval
if found()
  @linecnt,1 say 'Invoice Details:'
  linecnt = linecnt + 1
  @linecnt,1 say ' '
display INV_NO, INV_DATE, DESC, S_NO, Y_NO, AMT to printer for po_no = responseval
store 0 to minvtotal
sum to minvtotal for po_no = responseval
count to minvrecs for po_no = responseval
linecnt = linecnt + minvrecs + 3
@linecnt, 34 say 'Invoice Total:'
@linecnt, 54 say minvtotal picture '99999999.99'
linecnt = linecnt + 2
endif

set margin to 0
set device to screen
eject
return
```

```

store space(69) to clearline
responseval = response
store 'C' to closer
do while closer = 'C'
  store 'C' to confirm
  store 'Q' to closing
  clear
  select 2
  store ' ' to mvendor
  mvendor = vendor
  store ' ' to mdate
  mdate = date
  store ' ' to mstatus
  mstatus = close
  select 3
  @1,0 say 'INVOICE DATA ON FILE:'
  count to invnumber for c->po_no = response
  if invnumber > 6
    @4,10 say 'Invoice Records for PD exceed 6, select (P)rint for details'
  else
    display off for c->po_no = response
  endif
  sum to invtotal for c->po_no = response
  select 1
  @9,0 say 'PD DATA ON FILE:'
  count to ponumber for a->po_no = response
  if ponumber > 6
    @12,10 say 'Purchase Order Records for PD exceed 6, select (P)rint for details'
  else
    display off for a->po_no = response
  endif
  sum to pototal for a->po_no = response
  @17,0 say 'Vendor:'
  @17,10 say mvendor
  @17,25 say 'INV Total:'
  @17,35 say invtotal picture '99999999.99'
  @17,56 say 'PD Total:'
  @17,65 say pototal picture '99999999.99'
  @19,0 say 'PO Status:'
  @19,12 say mstatus
  @19,30 say 'PO_NO:'
  @19,40 say response
  @21,10 say clearline
@21,10 say '(C)lose, (V)oid, re(O)pen, (P)rint, (Q)uit' get closing picture 'X'
read
if UPPER(closing) = 'Q'
  clear
  select 2
  return
endif
if UPPER(closing) = 'C'
  select 2
  replace b->close with 'C'
  mystatus = close
  @19,12 say mstatus
  @21,10 say clearline
@21,10 say '(C)onfirm to close PO, or any key' get confirm picture 'X'
read
if UPPER(confirm) <> 'C'
  replace b->close with 'Q'

```

```

        endif
    endif
    if UPPER(closing) = 'V'
        select 3
        find &response
        if found()
            clear
            @13,1 say 'PO_NO'
            @13,9 say response
            @13,17 say 'has invoice data on file. Can NOT void.'
            @15,1 say ' '
            wait
        endif
        if .not. found()
            select 2
            replace b->close with 'V'
            mystatus = close
            @19,12 say mstatus
            @21,10 say clearline
@21,10 say '(C)onfirm deletes all PO data, or any key' get confirm picture 'X'
            read
                if UPPER(confirm) <> 'C'
                    replace b->close with 'O'
                else
                    select 1
                    delete for po_no = response
                    pack
                endif
            endif
        endif
        if UPPER(closing) = 'O'
            select 2
            replace b->close with 'O'
            mystatus = close
            @19,12 say mstatus
            @21,10 say clearline
@21,10 say '(C)onfirm to reopen, or any key' get confirm picture 'X'
            read
                if UPPER(confirm) <> 'C'
                    replace b->close with 'C'
                endif
            endif
        endif
        if UPPER(closing) = 'P'
            do printin
        endif
    endif
enddo

```



```
*printin.prg prints po/inv activity by specified po_no
```

```
select 2
set device to printer
set margin to 5
@6,10 say 'Invoice / Purchase Order Report'
@6,55 say date()
@9,10 say ' '

store 0 to linecnt
display off to printer
linecnt = linecnt + 14

select 1
find &responseval
if found()
  @linecnt,1 say 'Purchase Order Details:'
  linecnt = linecnt + 1
  @linecnt,1 say ' '
  display off DESC, S_NO, Y_NO, AMT to printer for po_no = responseval
  store 0 to mpototal
  sum to mpototal for po_no = responseval
  count to mporecs for po_no = responseval
  linecnt = linecnt + mporecs + 3
  @ linecnt,34 say 'Purchase Order Total:'
  @ linecnt,54 say mpototal picture '99999999.99'
  linecnt = linecnt + 3
endif

select 3
find &responseval
if found()
  @linecnt,1 say 'Invoice Details:'
  linecnt = linecnt + 1
  @linecnt,1 say ' '
display INV_NO, INV_DATE, DESC, S_NO, Y_NO, AMT to printer for po_no = responseval
store 0 to minvtotal
sum to minvtotal for po_no = responseval
count to minvrecs for po_no = responseval
linecnt = linecnt + minvrecs + 3
@linecnt, 34 say 'Invoice Total:'
@linecnt, 54 say minvtotal picture '99999999.99'
linecnt = linecnt + 2
endif

set margin to 0
set device to screen
eject
return
```

```

*journals.prg setups menus for printing journals by po, invoice, *or matches
store 'A' to another
do while another = 'A'
clear
store 'Q' to dates
@2,10 say '          Accountability 2.0'
@5,10 say '          2 - Journal Listings (journals.prg)'
@10,10 say '(A)ll dates, (R)ange dates, (Q)uit' get dates picture 'X'
  read
  store UPPER(dates) to dates
  if dates = 'Q'
    close all
    return
  endif
  if dates = 'A'
    store 'A' to alldates
    do while alldates = 'A'
      store 'Q' to response
      clear
      @5,15 say '          All dates - Journal Listings'
      @10,15 say '(P)Os, (I)nvoices, (C)ombined, (Q)uit' get response picture 'X'
      read
      store UPPER(response) to response
      if response = 'Q'
        exit
      endif
      if response = 'P'
        do jour_po.prg
        store 'A' to alldates
      endif
      if response = 'I'
        do jour_inv.prg
        store 'A' to alldates
      endif
      if response = 'C'
        do jour_com.prg
        store 'A' to alldates
      endif
    enddo
  endif
  if dates = 'R'
    store ' ' to begindate
    store ' ' to enddate
    @15,10 say 'Beginning Date: ' get begindate picture '@D 99/99/99'
    *see page 245 range with CTOD
    read
    @15,45 say 'Ending Date: ' get enddate picture '@D 99/99/99'
    read
    store CTOD(begindate) to begindate
    store CTOD(enddate) to enddate
    store 'D' to daterange
    do while daterange = 'D'
      clear
      store 'Q' to response
      @5,15 say begindate
      @5,25 say 'to'
      @5,29 say enddate
      @5,41 say '- Journal Listings'
      @10,15 say '(P)Os, (I)nvoices, (Q)uit' get response picture 'X'
      read

```

```
store UPPER(response) to response
if response = 'Q'
  exit
endif
if response = 'P'
  do jour_po
  store 'D' to daterange
endif
if response = 'I'
  store 'D' to datatype
  do while datatype = 'D'
  store 'Q' to bydate
  clear
  @5,15 say 'Range Dates -- Invoice Journal Report List By:'
@10,15 say '(P)O date, (I)nvoice date, (Q)uit ' get bydate picture 'X'
  read
  store UPPER(bydate) to bydate
  if bydate = 'Q'
    exit
  endif
  if bydate = 'P'
    do jour_inv
    store 'D' to daterange
  endif
  if bydate = 'I'
    do jour_inv
    store 'D' to daterange
  endif
  enddo
endif
enddo
endif
enddo
```

```

*jour_po.prg lists po records according to po_no, s_no, y_no, or
*close, open, void status
if response = 'P'
  close all
  clear
  @5,20 say 'Purchase Order Journal Reports -- Select:'
  @10,10 say 'Creating files . . . Do not interrupt.'
  close all
  select 5
    erase allpo.dbf
    erase allpo.mdx
  select 3
    use pofile order poindex
  select 2
    use poclose order cloindex
    if dates = 'A'
join with c to allpo.dbf for b->po_no = c->po_no fields c->po_no, b->vendor, b->date, c->desc, c->s_no, c->y_no, c->amt, b->close, b->
  else
join with c to allpo.dbf for b->po_no = c->po_no .AND. b->date >= begindate .AND. b->date <= enddate fields c->po_no, b->vendor, b->dat
  endif
  select 3
    use
  select 2
    use
  select 5
    use allpo.dbf
    index on po_no tag allpopo
    index on s_no tag allpos_
    index on y_no tag allpoy_
    index on close tag allpoclo

  store 'S' to sorter
  do while sorter = 'S'
  store 'Q' to polister
  clear
  if dates = 'A'
  @5,15 say 'All Dates -- Purchase Order Journal Reports -- Select:'
  else
  @5,15 say 'Range Dates -- Purchase Order Journal Reports -- Select:'
  endif
  @10, 20 say '(I)ndexed, (S)tatus, (Q)uit' get polister picture 'X'
  read
  store UPPER(polister) to polister
if polister = 'Q'
  exit
endif

if polister = 'I'
  store 'I' to isorter
  do while isorter = 'I'
  store 'Q' to indexlister
  clear
  if dates = 'A'
  @5,15 say 'All Dates - Purchase Order Journal Report Indexed By:'
  else
  @5,15 say 'Range Dates - Purchase Order Journal Report Indexed by:'
  endif
  @10,20 say '(P)D_NO, (S)_NO, (Y)_NO, (Q)uit' get indexlister picture 'X'
  read
  store UPPER(indexlister) to indexlister

```

```

if indexlister = 'Q'
  close all
  exit
endif
if indexlister = 'P'
  select 5
  set order to allpopo
  report form allpo.frm to printer
  store 'S' to sorter
endif
if indexlister = 'S'
  select 5
  set order to allpos_
  report form allpos_.frm to printer
  store 'S' to sorter
endif
if indexlister = 'Y'
  select 5
  store 'N' to tickresp
  @15,20 say '(A)ll, (R)econciled - Ticked, (N)ot Ticked' get tickresp picture 'X'
  read
  store UPPER(tickresp) to tickresp
  if tickresp = 'A'
    set order to allpoy_
    report form allpoy_.frm to printer
  endif
  if tickresp = 'R'
    set order to allpoy_
    report form allpoy_.frm to printer for tick_off = 'Y'
  endif
  if tickresp = 'N'
    set order to allpoy_
    report form allpoy_.frm to printer for tick_off = 'N'
  endif
  store 'S' to sorter
endif
enddo
endif

if polister = 'S'
  store 'S' to ssorter
  do while ssorter = 'S'
    store 'Q' to statuslist
    clear
    if dates = 'A'
      @5,15 say 'All Dates - Purchase Order Journal Report Status By:'
    else
      @5,15 say 'Range Dates - Purchase Order Journal Report Status by:'
    endif
@10,20 say '(O)pen, (C)losed, (V)oid, (A)ll, (D)uit' get statuslist picture 'X'
    read
    store UPPER(statuslist) to statuslist
    if statuslist = 'Q'
      exit
    endif
    if statuslist = 'O' .OR. statuslist = 'C'
      select 5
      use allpo.dbf
      set order to allpoclo
      if statuslist = 'O'

```

```
        report form allpoope.frm to printer for close = 'D'
        store 'S' to sorter
    endif
if statuslist = 'C'
    report form allpoclo.frm to printer for close = 'C'
    store 'S' to sorter
endif
endif
if statuslist = 'V' .OR. statuslist = 'A'
    select 2
    use poclose.dbf order closeind
    if statuslist = 'V'
        if dates = 'A'
            report form allpovoi.frm to printer for close = 'V'
        else
report form allpovoi.frm to printer for close = 'V' .AND. date >= begindate .AND. date <= enddate
        endif
        store 'S' to sorter
    endif
    if statuslist = 'A'
        if dates = 'A'
            report form allposta.frm to printer
        else
report form allposta.frm to printer for date >= begindate .AND. date <= enddate
        endif
        store 'S' to sorter
    endif
endif
endif
endif
endif
return
endif
```

```

*jour_inv.prg lists inv records according to po_no, s_no, y_no
if response = 'I'
  clear
  close all
  @10,10 say 'Creating files . . . Do not interrupt.'
  select 5
    erase allinv.dbf
    erase allinv.mdx
  select 3
    use invfile order invindex
  select 2
    use poclose order cloindex
    if dates = 'A'
join with c to allinv.dbf for b->po_no = c->po_no fields c->po_no, c->inv_no, b->vendor, c->inv_date, c->desc, c->s_no, c->y_no, c->a
  else
    if bydate = 'P'
join with c to allinv.dbf for b->po_no = c->po_no .AND. b->date >= begindate .AND. b->date <= enddate fields c->po_no, c->inv_no, b->ve
  else
join with c to allinv.dbf for b->po_no = c->po_no .AND. c->inv_date >= begindate .AND. c->inv_date <= enddate fields c->po_no, c->inv_n

    endif
  endif

select 5
  use allinv.dbf
  index on po_no tag allinvpo
  index on s_no tag allinvs_
  index on y_no tag allinvy_

  store 'I' to isorter
  do while isorter = 'I'
  store 'Q' to indexlister
  clear
  if dates = 'A'
  @5,20 say 'All Dates -- Invoice Journal Report Indexed By:'
  else
  @5,20 say 'Range Dates -- Invoice Journal Reports Indexed By:'
  endif
  endif
@10,10 say '(P)O_NO, (C)losed POs, (D)pen POs, (S)_NO, (Y)_NO, (Q)uit' get indexlister picture 'X'
  read
  store UPPER(indexlister) to indexlister
  if indexlister = 'Q'
    close all
    exit
  endif
  if indexlister = 'P'
    use allinv.dbf
    set order to allinvpo
    report form allinv.frm to printer
  endif
*look at changing heading based on all dates or range date see heading on page 162
  endif
  if indexlister = 'C'
    use allinv.dbf
    set order to allinvpo
    report form allclo.frm for close = 'C' to printer
  endif
  if indexlister = 'D'
    use allinv.dbf
    set order to allinvpo

```

```
        report form allopen.frm for close = '0' to printer
    endif
    if indexlister = 'S'
        use allinv.dbf
        set order to allinvs_
        report form allinvs_.frm to printer
    endif
    if indexlister = 'Y'
        store 'N' to tickresp
        @15,20 say '(A)ll, (R)econciled - Ticked, (N)ot Ticked' get tickresp picture 'X'
        read
        store UPPER(tickresp) to tickresp
        use allinv.dbf
        set order to allinvy_
        if tickresp = 'A'
            report form allinvy_.frm to printer
        endif
        if tickresp = 'R'
            report form allinvy_.frm to printer for tick_off = 'Y'
        endif
        if tickresp = 'N'
            report form allinvy_.frm to printer for tick_off = 'N'
        endif
    endif
enddo
endif
```


*jour_com prints a combined list of po/inv activity in sequence by po_no

```

select 3
use invfile order invindex
select 2
use poclose order cloindex
goto 1
select 1
use pofile order poindex
select 2
store 'A' to another
do while another = 'A'

    store 'Q' to combine
    store 0 to pagectn
    store '    ' to apo_no

    clear
    @5,10 say 'All Dates -- Combined Journal Report -- Select:'
    @10,10 say '(O)pen, (C)losed, (A)ll, (Q)uit ' get combine picture 'X'
    read
    store UPPER(combine) to combine

    if combine = 'Q'
        return
    endif

```

*modify to show if no open or closed on file

```

if combine = 'O'
    set order to closeind
    find 'O'
    if found()
        do while close = 'O'
            set device to printer
            set margin to 5
            @6,15 say 'Combined PO/INV Activity Report -- Open POs'
            @6,60 say date()
            @9,10 say ' '
            do comprint
            skip
        enddo
    endif
    store 'A' to another
endif

if combine = 'C'
    set order to closeind
    find 'C'
    if found()
        do while close = 'C'
            set device to printer
            set margin to 5
            @6,15 say 'Combined PO/INV Activity Report -- Closed POs'
            @6,60 say date()
            @9,10 say ' '
            do comprint
            skip
        enddo
    endif
    store 'A' to another
endif

```

```
endif

if combine = 'A'
  set order to cloindex
  goto top
  do while .not. eof()
    set device to printer
    set margin to 5
    @6,15 say 'Combined PO/INV Activity Report -- All POs'
    @6,60 say date()
    @9,10 say ' '
    do comprint
    skip
  enddo
  store 'A' to another
endif

enddo
return
```

```

*s_no_bal.prg projects ending balances per s_no or all
*copyright John E. Percevault, Box 2236 - Edson, AB - TOE OPO
store 'A' to another
do while another = 'A'
select 8
use s_no_req order s_no
select 7
use sumfile
erase posingle.dbf
select 4
use budget index budindex
select 3
use pofile order poindex
select 2
use poclose order cloindex
goto top
select 1
use invfile order invindex
store 'R' to retry
do while retry = 'R'
store 'R' to invalidtry
do while invalidtry = 'R'
    store 'P' to printing
    select 4
    store 'Q ' to response
clear
    @2,10 say '          AccountAbility 2.0'
    @5,10 say '          3 - S_NO Balances (s_no_bal.prg)'
@10,10 say 'Enter S_NO, (R)equested departments, (A)ll departments, (Q)uit' get response picture 'NW'
read
    responseval = response
    store UPPER(trim(response)) to response
    store len(response) to length
    if length = 2
        exit
    else
        if response = 'R' .or. response = 'A' .or. response = 'Q'
            exit
        else
            @12,10 say 'Invalid entry, (R)etry or any key to menu' get invalidtry picture 'X'
            read
            if UPPER(invalidtry) <> 'R'
                close all
                return
            endif
        endif
    endif
endif
enddo
if right(response,1) = 'Q'
    close all
    return
endif
if response = 'R'
    do s_no_req
    exit
endif
if response = 'A'
    store 'S' to reporttype
@15,10 say '(S)ummary, (D)etail, or any key to menu' get reporttype picture 'X'
read

```

```
if UPPER(reporttype) <> 'S' .and. UPPER(reporttype) <> 'D'
    exit
endif
if UPPER(reporttype) = 'S'
    do s_no_sum
    exit
endif
if UPPER(reporttype) = 'D'
    do s_no_det
    exit
endif

endif
if length = 2
    find &responseval
    if .not. found()
        @13,10 say 'S_NO'
        @13,18 say response
        @13,24 say 'is not on file.'
@16,10 say '(R)etry, or any key to main menu' get retry picture 'X'
        read
        if UPPER(retry) <> 'R'
            close all
            return
        endif
    endif
    if found()
        do s_no_one
    endif
endif
enddo
enddo
```

```

*s_no_one.prg projects ending balances for one s_no
*copyright John E. Percevault, Box 2236 - Edson, AB - TOE OPD
clear
@10,13 say 'Calculating S_NO Account Balance . . Do not interrupt.'
  select 3
  set relation to po_no into poclose
  sum to mpototal for c->s_no = response .and. b->close = '0'
  set relation to
  select 1
    set relation to po_no into poclose
    sum to minvttotal for a->s_no = response .and. b->close = 'C'
    set relation to
    store 0 to mcommitment
    mcommitment = mpototal + minvttotal
    store 0 to mallocation
    mallocation = d->allocation
    store '      ' to mdesc
    mdesc = d->desc
    store 0 to mbalance
    mbalance = mallocation - mcommitment

  clear
  @1,15 say 'Department Summary --'
  @1,45 say mdesc
  @1,65 say date()
  @4,15 say 'Budget allocation . . . . . '
  @4,50 say mallocation picture '99999999.99'
  @5,53 say '-----'
  @6,15 say 'Purchase Order Commitment . .'
  @6,50 say mpototal picture '99999999.99'
  @8,15 say 'Paid Invoice Total . . . . . '
  @8,50 say minvttotal picture '99999999.99'
  @9,53 say '-----'
  @11,15 say 'Total Budget Commitment . .'
  @11,50 say mcommitment picture '99999999.99'
  @12,53 say '-----'
  @14,15 say 'Projected Department Balance:'
  @14,50 say mbalance picture '99999999.99'
  @15,53 say '-----'

@20,10 say 'PRINT SCREEN, (P)rint all account activity, or any key to menu' get printing picture 'X'
read
  if UPPER(printing) = 'P'
    select 3
    set relation to po_no into poclose
    set device to printer
    @6,10 say 'Purchase Order Details --'
    @6,40 say mdesc
    @6,60 say date()
    @9,15 say 'Purchase Order Commitment . .'
    @9,53 say mpototal picture '99999999.99'
    @12,10 say '      '
    list off to printer for c->s_no = response .and. b->close = '0' c->PO_NO, b->VENDOR, b->DATE, c->DESC, c->S_NO, c->Y_NO,
    ? ' '
    set device to screen
    eject
    set relation to
    select 1
    set relation to po_no into poclose
    set device to printer

```

```
@6,10 say 'Paid Invoice Details --'
@6,40 say mdesc
@6,60 say date()
@9,15 say 'Paid Invoice Total . . .'
@9,53 say minvttotal picture '99999999.99'
@12,10 say ' '
list off to printer for a->s_no = response .and. b->close = 'C' a->PO_NO, b->VENDOR, a->INV_NO, a->DESC, a->S_NO, a->Y_NO, a->AMT, b->C
? ' '
set device to screen
eject
set relation to
set device to printer
@6,15 say 'Printed on:'
@6,35 say date()
@6,50 say mdesc
@9,15 say 'Total Budget Commitment . . '
@9,53 say mcommitment picture '99999999.99'
@12,15 say 'Budget Allocation . . . . . '
@12,53 say mallocation picture '99999999.99'
@17,15 say 'Projected Department Balance:'
@17,53 say mbalance picture '99999999.99'
@18,10 say ' '
set device to screen
eject
else
store 'A' to another
return
endif
```

```

store 'A' to anotherreq
do while anotherreq = 'A'
store 'R' to invalidtry
do while invalidtry = 'R'
store 0 to entryval
store 'Q ' to entry
select 4
clear
@10,10 say 'Enter S_NO, (D)isplay requested S_NOs, (P)rocess, (Q)uit' get entry picture 'NN'
read
entryval = entry
store UPPER(trim(entry)) to entry
store len(entry) to length
if length < 1
    @12,10 say 'Invalid entry, (R)etry, or any key to menu' get invalidtry picture 'X'
    read
    if UPPER(invalidtry) <> 'R'
        return
    endif
endif
if entry = 'Q'
    select 8
    delete all
    pack
    return
else
    exit
endif
enddo
if entry = 'D'
    select 8
    clear
    display all off
    wait
endif
if entry = 'P'
    clear
    @10,1 say 'This report goes directly to the printer.'
    @13,1 say 'Set printer to 17 pitch or use wide paper.'
    @15,13 say ' '
    wait
    clear
    @10,8 say 'Calculating and printing requested S_No account details.'
    @12,8 say 'Do not interrupt.'
    store ' ' to s_nopoint
    select 8
    goto top
    do while .not. eof()
        s_nopoint = h->s_no
        select 3
        set relation to po_no into poclose
        store ' ' to mdesc
        mdesc = h->desc
        sum to mpototal for c->s_no = s_nopoint .and. b->close = 'D'
        set device to printer
        @6,10 say 'Purchase Order Details - -'
        @6,40 say mdesc
        @6,60 say date()
        @9,15 say 'Purchase Order Commitment . .'
        @9,52 say mpototal picture '99999999.99'
    enddo

```

```

@12,10 say ' '
list off to printer for c->s_no = s_nopoint .and. b->close = '0' c->PO_NO, b->VENDOR, b->DATE, c->DESC,
? ' '
set device to screen
eject
set relation to

select 1
set relation to po_no into poclose
sum to minvtotal for a->s_no = s_nopoint .and. b->close = 'C'
store 0 to mcommitment
mcommitment = mpototal + minvtotal
store 0 to mallocation
mallocation = d->allocation
store 0 to mbalance
mbalance = mallocation - mcommitment
set device to printer
@6,10 say 'Paid Invoice Details - -'
@6,40 say mdesc
@6,60 say date()
@9,15 say 'Paid Invoice Total . . .'
@9,53 say minvtotal picture '99999999.99'
@12,10 say ' '
list off to printer for a->s_no = s_nopoint .and. b->close = 'C' a->PO_NO, b->VENDOR, a->INV_NO, a->DES
? ' '
set device to screen
eject
set device to printer
@6,15 say 'Printed on:'
@6,35 say date()
@6,50 say mdesc
@9,15 say 'Total Budget Commitment . .'
@9,53 say mcommitment picture '99999999.99'
@12,15 say 'Budget Allocation . .'
@12,53 say mallocation picture '99999999.99'
@17,15 say 'Projected Department Balance:'
@17,53 say mbalance picture '99999999.99'
@18,10 say ' '
set device to screen
eject
set relation to

select 8
skip
if eof()
  delete all
  pack
  return
endif
enddo

endif

if length = 2
select 4
find &entryval
if found()
  select 8
  find &entryval
  if found()
    clear

```



```
display for s_no = entry
store 'A' to anotherdel
do while anotherdel = 'A'
store 'Q' to dels_no
@10,10 say '(D)delete requested S_NO, (Q)uit' get dels_no picture 'X'
read
store UPPER(dels_no) to dels_no
if dels_no = 'Q'
    exit
endif
if dels_no = 'D'
    delete
    pack
    exit
endif
enddo
else
store 'C' to confirm
@15,10 say '(C)onfirm, or any key' get confirm picture 'X'
read
if UPPER(confirm) = 'C'
    append blank
    replace h->s_no with entry
    replace h->desc with d->desc
endif
endif
else
@13,0 say 'S_NO'
@13,8 say entry
@13,14 say 'is not on file.'
@14,0 say ' '
wait
endif
endif
enddo
```

*s_no_sum.prg projects ending summary balances for all s_no
 *copyright John E. Percevault, Box 2236 - Edson, AB - TQE OFO

```

clear
@10,13 say 'Calculating S_NO Account Balances . . Do not interrupt.'
store 0 to schallocate, schpototal, schintotal, schgrand, schbalance
select 4
store 0 to s_no_number
count to s_no_number
store 1 to scounter
goto top
do while .not. eof()
  @15,5 say scounter
  @15,17 say 'of'
  @15,19 say s_no_number
  @15,32 say 'department balances calculating.'
  store ' ' to response
  response = d->s_no

  select 2

  select 3
  set relation to po_no into poclose
  store 0 to mpototal
  sum to mpototal for c->s_no = response .and. b->close = 'D'
  set relation to

  select 1
  set relation to po_no into poclose
  store 0 to minvtotal
  sum to minvtotal for a->s_no = response .and. b->close = 'C'
  set relation to

  schpototal = schpototal + mpototal

  schintotal = schintotal + minvtotal
  store 0 to mcommitment
  mcommitment = mpototal + minvtotal

  schgrand = schgrand + mcommitment
  store 0 to mallocation
  mallocation = d->allocation
  schallocate = schallocate + mallocation
  store 0 to mbalance

  mbalance = mallocation - mcommitment
  schbalance = schbalance + mbalance

  select 7
  use sumfile
  append blank
  replace s_no with response
  replace desc with d->desc
  replace allocation with mallocation
  replace pototal with mpototal
  replace invtotal with minvtotal
  replace commitment with mcommitment
  replace balance with mbalance

  store 'Q ' to retry

```

```

select 4
  skip
    scounter = scounter + 1
  if eof()
    select 7
    store 'P' to lister
    store 'R' to relister
    do while relister = 'R'
    clear
    @10,15 say 'S_NO Account Balances Calculated'
@20,15 say '(P)rint, (S)creen listing, or any key to main menu' get lister picture 'X'
    read
    if UPPER(lister) <> 'P' .and. UPPER(lister) <> 'S'
      select 7
      use sumfile
      delete all
      pack
      exit
    endif
    if UPPER(lister) = 'S'
      clear
      @1,25 say 'S_NO Account Balances -- Summary Form'
      list all off S_NO, ALLOCATION, POTOTAL, INVTOTAL, COMMITMENT, BALANCE
      ? ' '
      ? ' '
      ? 'School Totals:'
? ' Budget PD Total INV Total Commitment Balance'
? ' ', schallocate picture '99999999.99', schpototal picture '99999999.99' AT 20, schintotal picture '99999999.99' AT 33, schgrand
      ? ' '
      store 'R' to relister
      wait
    endif

    if UPPER(lister) = 'P'
      clear
      @10,0 say 'Set printer to 17 pitch or use wide paper.'
      wait
      set device to printer
      @1,15 say 'S_NO Account Balances -- Summary Form'
      @1,60 say date()
      @2,25 say ' '
      list all off to printer
      @56,1 say 'School Totals:'
@58,1 say ' Budget PD Total INV Total Commitment Balance'
      @59,28 say schallocate picture '99999999.99'
      @59,41 say schpototal picture '99999999.99'
      @59,54 say schintotal picture '99999999.99'
      @59,67 say schgrand picture '99999999.99'
      @59,80 say schbalance picture '99999999.99'
      @60,1 say ' '
      set device to screen
      eject
      store 'R' to relister
    endif
  enddo
return
endif
enddo

```

*s_no_det.prg prints account details per s_no and projects all ending balances

*copyright John E. Percevault, Box 2236 - Edson, AB - TOE OPO

clear

@10,1 say 'This report goes directly to the printer.'

@13,1 say 'Set printer to 17 pitch or use wide paper.'

@15,13 say ' '

wait

clear

@10,8 say 'Calculating & Printing S_NO Account Details . . Do not interrupt.'

store 0 to schallocate, schpototal, schintotal, schgrand, schbalance

select 4

do while .not. eof()

 response = d->s_no

 select 3

 set relation to po_no into poclose

 store ' ' to mdesc

 mdesc = d->desc

 sum to mpototal for c->s_no = response .and. b->close = '0'

 schpototal = schpototal + mpototal

 set device to printer

 @6,10 say 'Purchase Order Details -- '

 @6,40 say mdesc

 @6,60 say date()

 @9,15 say 'Purchase Order Commitment . .'

 @9,53 say mpototal picture '99999999.99'

 @12,10 say ' '

 list off to printer for c->s_no = response .and. b->close = '0' c->PO_NO, b->VENDOR, b->DATE, c->DESC, c->S_NO, c->Y_NO, c->AMT

 ? ' '

 set device to screen

 eject

 set relation to

select 1

 set relation to po_no into poclose

 sum to minvtotal for a->s_no = response .and. b->close = 'C'

 schintotal = schintotal + minvtotal

 store 0 to mcommitment

 mcommitment = mpototal + minvtotal

 schgrand = schgrand + mcommitment

 store 0 to mallocation

 mallocation = d->allocation

 schallocate = schallocate + mallocation

 store 0 to mbalance

 mbalance = mallocation - mcommitment

 schbalance = schbalance + mbalance

 set device to printer

 @6,10 say 'Paid Invoice Details --'

 @6,40 say mdesc

 @6,60 say date()

 @9,15 say 'Paid Invoice Total . . .'

 @9,53 say minvtotal picture '99999999.99'

 @12,10 say ' '

 list off to printer for a->s_no = response .and. b->close = 'C' a->PO_NO, b->VENDOR, a->INV_NO, a->DESC, a->S_NO, a->Y_NO, a->AMT, b->C

 ? ' '

 set device to screen

 eject

 set device to printer

 @6,15 say 'Printed on:'

```

    @6,35 say date()
    @6,50 say mdesc
    @9,15 say 'Total Budget Commitment . . .'
    @9,53 say mcommitment picture '99999999.99'
    @12,15 say 'Budget Allocation . . . . .'
    @12,53 say mallocation picture '99999999.99'
    @17,15 say 'Projected Department Balance:'
    @17,53 say mbalance picture '99999999.99'
    @18,10 say ' '
    set device to screen
    set relation to

select 7
  use suafile
  append blank
  replace s_no with response
  replace desc with d->desc
  replace allocation with mallocation
  replace pototal with mpototal
  replace invtotal with minvtotal
  replace commitment with mcommitment
  replace balance with mbalance

  store 'Q ' to retry

select 4
  skip

  if eof()
    select 7
      clear
      set device to printer
      @1,25 say 'S_NO Account Balances -- Summary Form'
      @2,25 say ' '
      list all off to printer
      @56,1 say 'School Totals:'
      @58,1 say '
                Budget      PO Total      INV Total      Commitment      Balance'
      @59,28 say schallocate picture '99999999.99'
      @59,41 say schpototal picture '99999999.99'
      @59,54 say schintotal picture '99999999.99'
      @59,67 say schgrand picture '99999999.99'
      @59,80 say schbalance picture '99999999.99'
      @60,1 say ' '
      set device to screen
      eject
      store 'R' to relister
      delete all
      pack
    endif
  enddo

```

```

close all
select 2
use poclose order cloindex
store 'Y' to tick
store 'A' to another tick
do while another tick = 'A'
  store 'R' to retry
  do while retry = 'R'
    clear
    store 'Q' to response
    @2,10 say 'AccountAbility 2.0 '
    @5,15 say ' 4 - Reconciliation - Tick off PDs'
    @10,20 say 'Enter PD_NO, (Q)uit' get response picture '999999'
    read
    store trim(response) to response
    store len(response) to length
    if length < 6 .and. right(response,1) <> 'Q'
      @12,10 say 'Invalid PD_NO, (R)etry, or any key to menu' get retry picture 'X'
      read
      if UPPER(retry) <> 'R'
        close all
        return
      endif
    endif
    if right(response,1) = 'Q'
      close all
      return
    endif
    if length = 6
      find &response
      if .not. found()
        @12,10 say 'Invalid PD_NO, (R)etry, or any key to menu' get retry picture 'X'
        read
        if UPPER(retry) <> 'R'
          close all
          return
        endif
      else
        store 'A' to another resp
        store 'Y' to tick resp
        do while another resp = 'A'
          @13,30 say 'Tick_Off = '
          @13,43 say b->tick_off
          @17,15 say '(Y)es to Tick_Off, (N)o, or any key to menu' get tick resp picture 'X'
          read
          store UPPER(tick resp) to tick resp
          if tick resp = 'Y' .or. tick resp = 'N'
            replace b->tick_off with tick resp
          else
            exit
          endif
        enddo
      endif
    endif
  enddo
enddo

```

```
*allocate.prg
store 'A' to again
do while again = 'A'
  close all
  store 'Q' to response1
  clear
  @2,10 say '          AccountAbility 2.0'
  @5,10 say '    5 - Create / Edit Distributions (allocate.prg)'
  @10,10 say '(B)udget, (Y)ellowhead, (Q)uit' get response1 picture 'X'
  read
  if UPPER(response1) = 'Q'
    return
  endif
  if UPPER(response1) = 'B'
    do budget.prg
  endif
  if UPPER(response1) = 'Y'
    do ysd.prg
  endif
enddo
```

```

*budget.prg creates school departments and allocations
*copyright John E. Percevault, Box 2236 - Edson, AB, July 87
select 4
use budget index budindex
select 3
use pofile order pos_no
select 1
use invfile order invs_no
store 'R' to retry
do while 'R' = retry
store 'R' to invalidtry
do while invalidtry = 'R'
  select 4
  store ' ' to editrecords
  store 0 to responseval
  store 'Q' to response
  clear
  @2,10 say '          AccountAbility 2.0'
  @5,10 say '    Create / Edit Budget Allocations (budget.prg)'
  @10,10 say 'Enter S_NO, (D)isplay All, (P)rint All, (Q)uit' get response picture 'NN'
  read
  responseval = response
  store UPPER(trim(response)) to response
  store len(response) to length
  if length < 1
    @12,10 say 'Invalid entry, (R)etry, or any key to menu' get invalidtry picture 'X'
    read
    if UPPER(invalidtry) <> 'R'
      close all
      return
    endif
  endif
  endif
  if right(response,1) = 'Q'
    close all
    return
  else
    exit
  endif
enddo
if response = 'D'
  select 4
  sum all to budtotal
  clear
  @1,10 say 'School Allocation Total:'
  @1,40 say budtotal picture '99999999.99'
  @3,10 say ' '
  wait
  display all off
  wait
endif
if response = 'P'
  select 4
  sum all to budtotal
  set device to printer
  @5,10 say 'School Allocation Total:'
  @5,35 say budtotal picture '99999999.99'
  @5,55 say date()
  @6,10 say ' '
  list all off to printer
  set device to screen

```



```

    eject
  endif
  if length = 2
    find &responseval
    if found()
      store 'E' to editrecords
      @13,10 say 'S_NO'
      @13,18 say responseval
      @13,23 say 'is currently on file.'
    @16,10 say '(E)dit S_NO, or any key to menu' get editrecords
      read
      if UPPER(editrecords) = 'E'
        store 'E' to editrecords
      endif
    endif
    if .not. found() .and. response <> 'P' .and. response <> 'D'
      clear
      store ' ' to mdesc
      store 0 to mallocation
      store 'C' to confirm
      @10,10 say 'S_NO'
      @10,20 say responseval
      @13,10 say 'Enter department description:' get mdesc
      read
      @16,10 say 'Enter department allocation' get mallocation
      read
    @20,10 say '(C)onfirm department, or any key to menu' get confirm picture 'X'
      read
      if UPPER(confirm) = 'C'
        append blank
        replace s_no with response
        replace desc with mdesc
        replace allocation with mallocation
        store 'E' to editrecords
      endif
    endif
  endif
  endif

do while editrecords = 'E'
  clear
  sum all to budtotal
  @1,0 say 'SCHODL BUDGET ALLOCATION TOTAL:'
  @1,50 say budtotal
  @5,0 say 'DEPARTMENT BUDGET ON FILE:'
  display off for s_no = responseval
  @20,10 say '(E)dit S_NO, (D)etele S_NO, or any key to menu' get editrecords picture 'X'
  read
  if UPPER(editrecords) = 'E'
    do editbud
  endif
  if UPPER(editrecords) = 'D'
    select 3
    find &response
    if found()
      clear
      @17,10 say 'S_NO'
      @17,20 say responseval
    @17,23 say 'is on file with purchase order data. Can not delete!'
    @20,10 say '(R)etry, or any key to menu; then select 1 - Edit PO' get retry picture 'X'
      read

```

```
        if UPPER(retry) = 'R'
            exit
        endif
        if UPPER(retry) <> 'R'
            close all
            return
        endif
    endif
    select 1
    find &responseval
    if found()
        clear
        @17,10 say 'S_NO'
        @17,20 say responseval
    @17,23 say 'is on file with invoice data. Can not delete!'
    @20,10 say '(R)etry, or any key to menu; then select 2 - Edit Invoice' get retry picture 'X'
        read
            if UPPER(retry) = 'R'
                exit
            endif
            if UPPER(retry) <> 'R'
                close all
                return
            endif
        endif
        if .not. found()
            select 4
            delete for d->s_no = responseval
            pack
            clear
            sum all to budtotal
            @1,10 say 'School Allocation Total:'
            @1,40 say budtotal
            @3,10 say ' '
            display all off
            wait
        endif
    endif
    store 'R' to retry
enddo
enddo
```

```
*editbud allows editing of desc and allocation fields in budget.dbf
store space(69) to clearline
store 'E' to editingbud
do while editingbud = 'E'
  store 'Q' to editor
  clear
  @3,25 say 'Edit Record to Budget'
  @5,10 say ' '
  display off for s_no = responseval
  find &responseval
  @15,5 say '(D)escription, (A)llocation, (Q)uit' get editor picture 'X'
  read
  @15,5 say clearline
  if UPPER(editor) = 'Q'
    return
  endif
  if UPPER(editor) = 'D'
    store ' ' to newdesc
    @15,10 say 'New Description: ' get newdesc picture 'XXXXXXXXXXXXXXXXXXXXX'
    read
    replace desc with newdesc
    @15,5 say clearline
  endif
  if UPPER(editor) = 'A'
    store 0 to newvalue
    @15,10 say 'New Allocation: ' get newvalue picture '99999999999.99'
    read
    replace allocation with newvalue
    @15,1 say clearline
  endif
enddo
```

```

*ysd.prg creates yellowhead departments
*copyright John E. Percevault, Box 2236 - Edson, AB, July 87
select 4
use ysdfile index ysdindex
select 3
use pofile order poy_no
select 1
use invfile order invy_no
store 'R' to retry
do while 'R' = retry
store 'R' to invalidtry
do while invalidtry = 'R'
  select 4
  store ' ' to editrecords
  store 0 to responseval
  store 'Q' to response
  clear
  @2,10 say '          AccountAbility 2.0'
  @5,10 say '      Create / Edit Divison Accounts (ysd.prg)'
  @10,10 say 'Enter Y_NO, (D)isplay All, (P)rint All, (Q)uit' get response picture 'NNNNNN'
  read
  responseval = response
  store UPPER(trim(response)) to response
  store len(response) to length
  if length >= 5
    exit
  else
    if response = 'P' .or. response = 'D' .or. response = 'Q'
      exit
    else
      @12,10 say 'Invalid entry, (R)etry, or any key to menu' get invalidtry picture 'X'
      read
      if UPPER(invalidtry) <> 'R'
        close all
        return
      endif
    endif
  endif
endif
enddo
if right(response,1) = 'Q'
  close all
  return
endif
if response = 'D'
  select 4
  clear
  display all off Y_NO, DESC
  wait
endif
if response = 'P'
  select 4
  set device to printer
  @5,15 say 'Division Accounts on File:'
  @5,55 say date()
  @7,10 say ' '
  list Y_NO, DESC all off to printer
  set device to screen
  eject
endif
if length = 5 .or. length = 6

```

```

find &responseval
  if found()
    store 'E' to editrecords
    @13,10 say 'Y_NO'
    @13,18 say responseval
    @13,29 say 'is currently on file.'
@16,10 say '(E)dit Y_NO, or any key to menu' get editrecords
  read
  if UPPER(editrecords) = 'E'
    store 'E' to editrecords
  endif
endif
if .not. found() .and. response <> 'D' .and. response <> 'D'
  clear
  store ' ' to mdesc
  store 'C' to confirm
  @10,10 say 'Y_NO'
  @10,20 say responseval
  @13,10 say 'Enter department description:' get mdesc
  read
@20,10 say '(C)onfirm department, or any key to menu' get confirm picture 'X'
  read
  if UPPER(confirm) = 'C'
    append blank
    replace y_no with response
    replace desc with mdesc
    store 'E' to editrecords
  endif
endif
endif
do while editrecords = 'E'
  clear
  @1,0 say 'DIVISION DISTRIBUTION ON FILE:'
  @5,0 say ' '
  display Y_NO, DESC off for y_no = responseval
@20,10 say '(E)dit Y_NO, (D)delete Y_NO, or any key to menu' get editrecords picture 'X'
  read
  if UPPER(editrecords) = 'E'
    do editysd
  endif
  if UPPER(editrecords) = 'D'
    select 3
    find &response
    if found()
      clear
      @17,10 say 'Y_NO'
      @17,16 say responseval
@17,23 say 'is on file with purchase order data. Can not delete!'
@20,10 say '(R)etry, or any key to menu; then select 1 - Edit PO' get retry picture 'X'
    read
    if UPPER(retry) = 'R'
      exit
    endif
    if UPPER(retry) <> 'R'
      close all
      return
    endif
  endif
  select 1
  find &responseval

```

```
        if found()
            clear
            @17,10 say 'Y_NO'
            @17,16 say responseval
@17,23 say 'is on file with invoice data. Can not delete!'
@20,10 say '(R)etry, or any key to menu; then select 2 - Edit Invoice' get retry picture 'X'
            read
                if UPPER(retry) = 'R'
                    exit
                endif
                if UPPER(retry) <> 'R'
                    close all
                    return
                endif
            endif
            if .not. found()
                select 4
                delete for d->y_no = responseval
                pack
            endif
            endif
            store 'R' to retry
        enddo
    enddo
```

```
*editbud allows editing of desc and allocation fields in budget.dbf
store space(69) to clearline
store 'E' to editingbud
do while editingbud = 'E'
  store 'Q' to editor
  clear
  @3,25 say 'Edit Record to Division Distribution'
  @6,10 say ' '
  display Y_NO, DESC off for y_no = responseval
  find &responseval
  @15,5 say '(D)escription, (Q)uit' get editor picture 'X'
  read
  @15,5 say clearline
  if UPPER(editor) = 'Q'
    return
  endif
  if UPPER(editor) = 'D'
    store ' ' to newdesc
    @15,10 say 'New Description: ' get newdesc picture 'XXXXXXXXXXXXXXXXXXXXX'
    read
    replace desc with newdesc
    @15,5 say clearline
  endif
enddo
```

```

close all
store ' ' to backdata
store 'A' to anothercon
do while anothercon = 'A'
store 'Q' to response
clear
@2,10 say '          AccountAbility 2.0'
@5,10 say '          6 - Utilities (utility.dbo)'
@8,20 say 'Select: (B)ackup current year'
@10,20 say '          (S)et up new fiscal year'
@12,20 say '          (I)nstall program updates'
@14,20 say '          (R)estore required fiscal year'
@16,20 say '          (Q)uit'
@19,20 say 'Enter selection: ' get response picture 'X'
read
store UPPER(response) to response
if response = 'Q'
    return
endif
if response = 'B'
    clear
    store 'A' to anothercon
    do while anothercon = 'A'
        store 'C' to confirm
        @10,10 say 'Place a formatted data disk in drive A.'
        @15,10 say '(C)onfirm, or any key to menu.' get confirm picture 'X'
        read
        if UPPER(confirm) <> 'C'
            exit
        else
            copy file invfile.dbf to a:\invfile.dbf/v
            copy file ysdfile.dbf to a:\ysdfile.dbf/v
            copy file poclose.dbf to a:\poclose.dbf/v
            copy file pofile.dbf to a:\pofile.dbf/v
            copy file budget.dbf to a:\budget.dbf/v
            copy file sumfile.dbf to a:\sumfile.dbf/v
            copy file s_no_req.dbf to a:\s_no_req.dbf/v
            copy file budindex.ndx to a:\budindex.ndx/v
            copy file ysdindex.ndx to a:\ysdindex.ndx/v
            copy file pofile.mdx to a:\pofile.mdx/v
            copy file poclose.mdx to a:\poclose.mdx/v
            copy file invfile.mdx to a:\invfile.mdx/v
            copy file s_no_req.mdx to a:\s_no_req.mdx/v
            store 'B' to backdata
            exit
        endif
    enddo
endif
if response = 'S'
    clear
    if backdata <> 'B'
        @10,0 say 'Complete a current backup of data before'
        @12,0 say 'attempting this option.'
        @13,0 say ' '
        wait
    else
        store 'A' to anothercon
        do while anothercon = 'A'
            store 'C' to confirm
            @5,20 say 'Set up for New Fiscal Year'

```



```

@8,10 say 'This routine erases ALL purchase order and'
@9,10 say 'invoice data. On completion, edit as required'
@10,10 say 'the budget and ysd files to reflect new allocations'
@11,10 say 'or account codes for the new fiscal year.'
@14,10 say '(C)onfirm, or any key to menu.' get confirm picture 'X'
read
if UPPER(confirm) <> 'C'
    exit
else
    select 1
    use poclose
    delete all
    pack
    use pofile
    delete all
    pack
    use invfile
    delete all
    pack
    close all
    clear
    @10,0 say 'Purchase order and data files erased.'
    @11,0 say 'To start the new year, edit the budget'
    @12,0 say 'and ysd files as required.'
    @13,0 say ' '
    wait
    exit
endif
enddo
endif
endif
if response = 'I'
    clear
    store 'A' to anothercon
    do while anothercon = 'A'
        store 'C' to confirm
        @10,10 say 'Place the update program disk in drive A.'
        @15,10 say '(C)onfirm, or any key to menu.' get confirm picture 'X'
        read
        if UPPER(confirm) <> 'C'
            exit
        else
            erase s_no_det.prg
            erase poenter.prg
            erase appendin.prg
            erase appendpo.prg
            erase poclose.prg
            erase s_no_sum.prg
            erase budget.prg
            erase editin.prg
            erase editpo.prg
            erase ysd.prg
            erase menu.prg
            erase utility.prg
            erase allocate.prg
            erase editysd.prg
            erase editbud.prg
            erase jour_po.prg
            erase journals.prg
            erase inventer.prg

```

erase jour_inv.prg
 erase s_no_bal.prg
 erase s_no_one.prg
 erase printpo.prg
 erase updatein.prg
 erase jour_com.prg
 erase comprint.prg
 erase deletere.prg
 erase printin.prg
 erase dataent.prg
 erase fileedit.prg
 erase s_no_req.prg

erase s_no_det.dbo
 erase poenter.dbo
 erase appendin.dbo
 erase appendpo.dbo
 erase poclose.dbo
 erase s_no_sum.dbo
 erase budget.dbo
 erase editin.dbo
 erase editpo.dbo
 erase ysd.dbo
 erase allocate.dbo
 erase editysd.dbo
 erase editbud.dbo
 erase jour_po.dbo
 erase journals.dbo
 erase inventer.dbo
 erase jour_inv.dbo
 erase s_no_bal.dbo
 erase s_no_one.dbo
 erase printpo.dbo
 erase updatein.dbo
 erase jour_com.dbo
 erase comprint.dbo
 erase deletere.dbo
 erase printin.dbo
 erase dataent.dbo
 erase fileedit.dbo
 erase s_no_req.dbo

erase allpos_.frm
 erase allopen.frm
 erase allclo.frm
 erase allpoy_.frm
 erase allpoope.frm
 erase allpoclo.frm
 erase allpo.frm
 erase allpovoi.frm
 erase allposta.frm
 erase allinv.frm
 erase allinvs_.frm
 erase allinvy_.frm

erase allpos_.fro
 erase allopen.fro
 erase allclo.fro
 erase allpoy_.fro
 erase allpoope.fro

```

erase allpoclo.fro
erase allpo.fro
erase allpovoi.fro
erase allposta.fro
erase allinv.fro
erase allinvs_.fro
erase allinvy_.fro

```

```

erase allpos_.frg
erase allopen.frg
erase allclo.frg
erase allpoy_.frg
erase allpoope.frg
erase allpoclo.frg
erase allpo.frg
erase allpovoi.frg
erase allposta.frg
erase allinv.frg
erase allinvs_.frg
erase allinvy_.frg

```

```

copy file a:s_no_det.prg to c:\dbase\s_no_det.prg/v
copy file a:poenter.prg to c:\dbase\poenter.prg/v
copy file a:\appendin.prg to c:\dbase\appendin.prg/v
copy file a:\appendpo.prg to c:\dbase\appendpo.prg/v
copy file a:\poclose.prg to c:\dbase\poclose.prg/v
copy file a:\s_no_sum.prg to c:\dbase\s_no_sum.prg/v
copy file a:\budget.prg to c:\dbase\budget.prg/v
copy file a:\editin.prg to c:\dbase\editin.prg/v
copy file a:\editpo.prg to c:\dbase\editpo.prg/v
copy file a:\ysd.prg to c:\dbase\ysd.prg/v
copy file a:\menu.prg to c:\dbase\menu.prg/v
copy file a:\allocate.prg to c:\dbase\allocate.prg/v
copy file a:\editysd.prg to c:\dbase\editysd.prg/v
copy file a:\editbud.prg to c:\dbase\editbud.prg/v
copy file a:\jour_po.prg to c:\dbase\jour_po.prg/v
copy file a:\journals.prg to c:\dbase\journals.prg/v
copy file a:\inventer.prg to c:\dbase\inventer.prg/v
copy file a:\jour_inv.prg to c:\dbase\jour_inv.prg/v
copy file a:\s_no_bal.prg to c:\dbase\s_no_bal.prg/v
copy file a:\s_no_one.prg to c:\dbase\s_no_one.prg/v
copy file a:\printpo.prg to c:\dbase\printpo.prg/v
copy file a:\updatein.prg to c:\dbase\updatein.prg/v
copy file a:\jour_com.prg to c:\dbase\jour_com.prg/v
copy file a:\comprint.prg to c:\dbase\comprint.prg/v
copy file a:\deletere.prg to c:\dbase\deletere.prg/v
copy file a:\printin.prg to c:\dbase\printin.prg/v
copy file a:\utility.prg to c:\dbase\utility.prg/v
copy file a:\dataent.prg to c:\dbase\dataent.prg/v
copy file a:\fileedit.prg to c:\dbase\fileedit.prg/v
copy file a:\s_no_req.prg to c:\dbase\s_no_req.prg/v

```

```

copy file a:s_no_det.dbo to c:\dbase\s_no_det.dbo/v
copy file a:poenter.dbo to c:\dbase\poenter.dbo/v
copy file a:\appendin.dbo to c:\dbase\appendin.dbo/v
copy file a:\appendpo.dbo to c:\dbase\appendpo.dbo/v
copy file a:\poclose.dbo to c:\dbase\poclose.dbo/v
copy file a:\s_no_sum.dbo to c:\dbase\s_no_sum.dbo/v
copy file a:\budget.dbo to c:\dbase\budget.dbo/v
copy file a:\editin.dbo to c:\dbase\editin.dbo/v

```

```
copy file a:\editpo.dbo to c:\dbase\editpo.dbo/v
copy file a:\ysd.dbo to c:\dbase\ysd.dbo/v
copy file a:\allocate.dbo to c:\dbase\allocate.dbo/v
copy file a:\editysd.dbo to c:\dbase\editysd.dbo/v
copy file a:\editbud.dbo to c:\dbase\editbud.dbo/v
copy file a:\jour_po.dbo to c:\dbase\jour_po.dbo/v
copy file a:\journals.dbo to c:\dbase\journals.dbo/v
copy file a:\inventer.dbo to c:\dbase\inventer.dbo/v
copy file a:\jour_inv.dbo to c:\dbase\jour_inv.dbo/v
copy file a:\s_no_bal.dbo to c:\dbase\s_no_bal.dbo/v
copy file a:\s_no_one.dbo to c:\dbase\s_no_one.dbo/v
copy file a:\printpo.dbo to c:\dbase\printpo.dbo/v
copy file a:\updatein.dbo to c:\dbase\updatein.dbo/v
copy file a:\jour_com.dbo to c:\dbase\jour_com.dbo/v
copy file a:\comprint.dbo to c:\dbase\comprint.dbo/v
copy file a:\deletere.dbo to c:\dbase\deletere.dbo/v
copy file a:\printin.dbo to c:\dbase\printin.dbo/v
copy file a:\dataent.dbo to c:\dbase\dataent.dbo/v
copy file a:\fileedit.dbo to c:\dbase\fileedit.dbo/v
copy file a:\s_no_req.dbo to c:\dbase\s_no_req.dbo/v
```

```
copy file a:\allpos_.fro to c:\dbase\allpos_.fro/v
copy file a:\allopen.fro to c:\dbase\allopen.fro/v
copy file a:\allclo.fro to c:\dbase\allclo.fro/v
copy file a:\allpoy_.fro to c:\dbase\allpoy_.fro/v
copy file a:\allpoope.fro to c:\dbase\allpoope.fro/v
copy file a:\allpoclo.fro to c:\dbase\allpoclo.fro/v
copy file a:\allpo.fro to c:\dbase\allpo.fro.v
copy file a:\allpovoi.fro to c:\dbase\allpovoi.fro/v
copy file a:\allposta.fro to c:\dbase\allposta.fro/v
copy file a:\allinv.fro to c:\dbase\allinv.fro/v
copy file a:\allinvs_.fro to c:\dbase\allinvs_.fro/v
copy file a:\allinvy_.fro to c:\dbase\allinvy_.fro/v
```

```
copy file a:\allpos_.frm to c:\dbase\allpos_.frm/v
copy file a:\allopen.frm to c:\dbase\allopen.frm/v
copy file a:\allclo.frm to c:\dbase\allclo.frm/v
copy file a:\allpoy_.frm to c:\dbase\allpoy_.frm/v
copy file a:\allpoope.frm to c:\dbase\allpoope.frm/v
copy file a:\allpoclo.frm to c:\dbase\allpoclo.frm/v
copy file a:\allpo.frm to c:\dbase\allpo.frm.v
copy file a:\allpovoi.frm to c:\dbase\allpovoi.frm/v
copy file a:\allposta.frm to c:\dbase\allposta.frm/v
copy file a:\allinv.frm to c:\dbase\allinv.frm/v
copy file a:\allinvs_.frm to c:\dbase\allinvs_.frm/v
copy file a:\allinvy_.frm to c:\dbase\allinvy_.frm/v
```

```
copy file a:\allpos_.frg to c:\dbase\allpos_.frg/v
copy file a:\allopen.frg to c:\dbase\allopen.frg/v
copy file a:\allclo.frg to c:\dbase\allclo.frg/v
copy file a:\allpoy_.frg to c:\dbase\allpoy_.frg/v
copy file a:\allpoope.frg to c:\dbase\allpoope.frg/v
copy file a:\allpoclo.frg to c:\dbase\allpoclo.frg/v
copy file a:\allpo.frg to c:\dbase\allpo.frg.v
copy file a:\allpovoi.frg to c:\dbase\allpovoi.frg/v
copy file a:\allposta.frg to c:\dbase\allposta.frg/v
copy file a:\allinv.frg to c:\dbase\allinv.frg/v
copy file a:\allinvs_.frg to c:\dbase\allinvs_.frg/v
copy file a:\allinvy_.frg to c:\dbase\allinvy_.frg/v
```

```

                exit
            endif
        enddo
    endif

    if response = 'R'
        clear
        if backdata <> 'B'
            @10,0 say 'Complete a current backup of data before'
            @12,0 say 'attempting this option.'
            @13,0 say ' '
            wait
        else
            store 'A' to anothercon
            do while anothercon = 'A'
                store 'C' to confirm
                @10,10 say 'Place the required fiscal year data disk in drive A.'
                @15,10 say '(C)onfirm, or any key to menu.' get confirm picture 'X'
                read
                if UPPER(confirm) <> 'C'
                    exit
                else
                    erase invfile.dbf
                    erase ysdfile.dbf
                    erase poclose.dbf
                    erase pofile.dbf
                    erase budget.dbf
                    erase sumfile.dbf
                    erase s_no_req.dbf
                    erase budindex.ndx
                    erase ysdindex.ndx
                    erase pofile.mdx
                    erase poclose.mdx
                    erase invfile.mdx
                    erase s_no_req.mdx

                    copy file a:\invfile.dbf to c:\dbase\invfile.dbf/v
                    copy file a:\ysdfile.dbf to c:\dbase\ysdfile.dbf/v
                    copy file a:\poclose.dbf to c:\dbase\poclose.dbf/v
                    copy file a:\pofile.dbf to c:\dbase\pofile.dbf/v
                    copy file a:\budget.dbf to c:\dbase\budget.dbf/v
                    copy file a:\sumfile.dbf to c:\dbase\sumfile.dbf/v
                    copy file a:\s_no_req.dbf to c:\dbase\s_no_req.dbf/v
                    copy file a:\budindex.ndx to c:\dbase\budindex.ndx/v
                    copy file a:\ysdindex.ndx to c:\dbase\ysdindex.ndx/v
                    copy file a:\pofile.mdx to c:\dbase\pofile.mdx/v
                    copy file a:\poclose.mdx to c:\dbase\poclose.mdx/v
                    copy file a:\invfile.mdx to c:\dbase\invfile.mdx/v
                    copy file a:\s_no_req.mdx to c:\dbase\s_no_req.mdx/v
                    exit
                endif
            enddo
        endif
    endif
enddo
endif
enddo

```

SECTION FOUR - SAMPLE REPORTS



School Allocation Total: 82133.00 06/11/89

S_NO	DESC	ALLOCATION
10	Language Arts	1500.00
11	Mathematics	2200.00
12	Science	3500.00
13	Social Studies	1750.00
14	Physical Ed.	1500.00
15	Health	2500.00
16	French	2250.00
20	Art	550.00
21	Computers	4500.00
22	Drama	2200.00
23	Band	3300.00
24	Hunter Ed.	560.00
25	Living Skills	1200.00
26	Home Ec.	4000.00
27	IA 1	2400.00
28	IA 2	2400.00
29	IA 3	2400.00
30	Success Center	4500.00
31	Library	5500.00
40	Administration	5000.00
41	Counselling	1000.00
50	General Supply	350.00
51	Copy Costs	10000.00
52	Telephone	2500.00
53	Postage	1750.00
54	Freight	0.00
55	Public Relation	750.00
60	Capital	2157.00
61	YAC	2100.00
62	SUBS	0.00
63	Band Repairs	1250.00
64	SEICAG (Grant)	6566.00
65	French Im	0.00
66	CAPITAL REPLACE	0.00

Division Accounts on File:

06/11/89

Y_NO	DESC
03100	FRENCH IMM
03110	CAPITAL REPLACEMEN
03410	Services C & P
03440	Post & Phone
03520	Rentals
03610	S & M Instruction
03641	Texts
03651	Media
03669	Freight
05610	Spec. Ed. Inst.
05641	Spec. Ed. Texts
11410	Supp. Trans.
11420	YAC
12750	Capital
12751	Capital(-300)
33460	Other Mileage
99992	SUBS

Journal Listing -- All Invoices

Sorted by Y_NO

PG_NO	INV_NO	Vendor	INV_Date	Description	S_NO	Y_NO	Amount	C	T
** Y_NO	03410								
803285	467556	National Music	05/12/89	Repairs	63	03410	81.97	C	N
803285	468452	National Music	05/12/89	Repairs	63	03410	181.47	C	N
** Subtotal **							263.44		
** Y_NO	03440								
803294	7237482	A.G.T.	05/18/89	Telephone	52	03440	17.60	C	N
803294	7235515	A.G.T.	05/18/89	Telephone	52	03440	398.29	C	N
** Subtotal **							415.89		
** Y_NO	03520								
803290	0508449	Lion Business	05/17/89	Copy Costs	51	03520	68.64	D	N
** Subtotal **							68.64		
** Y_NO	03610								
803101	344400	L.R.D.Q	05/15/89	Books	15	03610	3.90	D	N
803175	34421	L.R.D.C.	05/15/89	Books	65	03610	42.52	D	N
803177	825329	School Service	05/17/89	Cassette	20	03610	52.45	C	N
803179	836525	B & B Supplies	05/18/89	Art Supplies	20	03610	18.82	D	N
803191	723361	Holt & Rinehart	05/18/89	D.A.T. Testing	41	03610	518.00	C	N
803217	19581	Opus Framing	05/24/89	Art Supplies	20	03610	224.62	D	N
803218	497257	Fitzhenry	05/05/89	Art Supplies	20	03610	76.10	C	N
803221	T0005096	Xerox Canada	05/17/89	Repair Typewrit	50	03610	96.00	D	N
803257	220768	Tandy	05/09/89	Supplies	27	03610	49.34	C	N
803263	50911480	Xerox Canada	05/15/89	French DaisyWhe	65	03610	29.77	C	N
803265	03127	Pacific Ed. Pre	05/09/89	book	65	03610	13.20	C	N
803266	E-2029	Carrefour	05/30/89	Books	65	03610	-79.44	D	N
803270	R001	Robinson Stores	05/10/89	Supplies	26	03610	10.53	C	N
803274	18947	Northwest Lab.	05/09/89	Frogs	12	03610	165.87	C	N
803275	86	LInford Foods	05/09/89	Home Ec Supply	26	03610	377.16	C	N
803280	E 20943	Burlington Art	05/24/89	Art Supplies	20	03610	188.69	D	N
803280	E 20973	Burlington Art	05/24/89	Art Supplies	20	03610	121.49	D	N

Journal Listing -- All Invoices

Sorted by Y_NO

PO_NO	INV_NO	Vendor	INV_Date	Description	S_NO	Y_NO	Amount	C	T
803281	68162	Mountain wood	05/24/89	Paint Set	20	03610	180.00	C	N
803291	75388	Timberjack	05/17/89	Trackmeet	14	03610	25.77	D	N
803292	70108	Timberjack	05/17/89	Suppl paint	20	03610	32.94	D	N
803295	803295	Western Data	05/30/89	Computer labels	50	03610	54.66	C	N
803297	030356	Edson	05/30/89	Rent Theatre	15	03610	200.00	D	N
903001	346283	L.R.D.C.	05/15/89	Culturel Books	65	03610	204.66	D	N
** Subtotal **							2723.43		
** Y_NO 03651									
803175	344400	L.R.D.C.	05/17/89	French Imm	65	03651	3.90	D	N
803175	346283	L.R.D.C.	05/17/89	French Imm	65	03651	204.66	D	N
803236	83685	Periodica	05/08/89	Magazines	65	03651	334.45	C	N
803237	246265HI	Les	05/09/89	books	65	03651	73.24	D	N
803237	242856HI	Publicatio	05/09/89	books	65	03651	635.45	D	N
803238	210801	Les	05/09/89	books	65	03651	635.45	D	N
803238	210801	Publicatio	05/09/89	books	65	03651	635.45	D	N
803238	210801	Sanders Book Co	05/08/89	French Imm Book	65	03651	346.70	C	N
803238	5934	Sanders Book Co	05/24/89	Card Kits	65	03651	25.00	C	N
803266	1897	Carrefour	05/10/89	books	65	03651	169.62	D	N
803389	01458	Formac Publishi	05/17/89	library books	31	03651	141.43	D	N
** Subtotal **							1934.45		
** Y_NO 05610									
803269	119284	School Book Fai	05/10/89	Books	30	05610	137.85	D	N
** Subtotal **							137.85		
** Y_NO 05641									
803267	A27B54	Dominie Press	05/05/89	Books	30	05641	38.55	C	N
803269	119559	School Book Fai	05/18/89	Books	30	05641	21.05	D	N
** Subtotal **							59.60		
** Y_NO 11410									
803277	218	Beaupre Bus	05/09/89	Bus Rental	23	11410	25.00	C	N
** Subtotal **							25.00		
** Y_NO 11420									
803276	217	Beaupre Bus	05/09/89	Bus Rental	61	11420	125.00	C	N

** Subtotal **

Page No. 3
06/11/89

Journal Listing -- All Invoices

Sorted by Y_NO

PO_NO	INV_NO	Vendor	INV_Date	Description	S_NO	Y_NO	Amount	C	T
							125.00		
** Y_NO	12751								
803273	1685	Ron's Outdoor	05/09/89	Cooper Goals	14	12751	252.00	C	N
** Subtotal **							252.00		
*** Total ***							6005.30		

S_NO	DESC	ALLOCATION	POTOTAL	INVTOTAL	COMMITMENT	BALANCE
10	Language Arts	1500.00	1214.66	31.70	1246.36	253.64
11	Mathematics	2200.00	46.78	0.00	46.78	2153.22
12	Science	3500.00	55.92	2241.46	2297.38	1202.62
13	Social Studies	1750.00	1455.54	0.00	1455.54	294.46
14	Physical Ed.	1500.00	2878.98	252.00	3130.98	-1630.98
15	Health	2500.00	381.75	0.00	381.75	2118.25
16	French	2250.00	176.60	0.00	176.60	2073.40
20	Art	550.00	2512.71	550.97	3063.68	-2513.68
21	Computers	4500.00	0.00	905.40	905.40	3594.60
22	Drama	2200.00	100.09	708.36	808.45	1391.55
23	Band	3300.00	200.00	631.00	831.00	2469.00
24	Hunter Ed.	550.00	161.50	0.00	161.50	398.50
25	Living Skills	1200.00	0.00	0.00	0.00	1200.00
26	Home Ec.	4000.00	2218.03	969.21	3187.24	812.76
27	IA 1	2400.00	755.90	1780.12	2536.02	-136.02
28	IA 2	2400.00	354.02	1805.75	2159.77	240.23
29	IA 3	2400.00	0.00	2323.21	2323.21	76.79
30	Success Center	4500.00	305.45	149.96	455.41	4044.59
31	Library	5500.00	161.43	4151.78	4313.21	1186.79
40	Administration	5000.00	16.95	10.95	27.90	4972.10
41	Counselling	1000.00	1103.38	647.68	1751.06	-751.06
50	General Supply	350.00	15369.24	1607.81	16977.05	-16627.05
51	Copy Costs	10000.00	1452.64	1210.30	2662.94	7337.06
52	Telephone	2500.00	1510.06	1828.83	3338.89	-838.89
53	Postage	1750.00	652.05	384.06	1036.11	713.89
54	Freight	0.00	171.20	121.37	292.57	-292.57
55	Public Relation	750.00	183.00	379.98	562.98	187.02
60	Capital	2157.00	329.00	0.00	329.00	1828.00
61	YAC	2100.00	1438.66	250.00	1688.66	411.34
62	SUBS	0.00	0.00	0.00	0.00	0.00
63	Band Repairs	1250.00	0.00	974.09	974.09	275.91
64	SEICAS (Grant)	6566.00	0.00	1123.13	1123.13	5442.87
65	French Im	0.00	5982.58	1000.32	6982.90	-6982.90
66	CAPITAL REPLACE	0.00	0.00	0.00	0.00	0.00

School Totals:

Budget	PD Total	INV Total	Commitment	Balance
82133.00	41188.12	26039.44	67227.56	14905.44

Purchase Order Details -- Physical Ed. 06/11/89

Purchase Order Commitment . . 2878.98

c->PO_NO	b->VENDOR	b->DATE	c->DESC	c->S_NO	c->Y_NO	c->AMT	b->CLOSE
803107	North Star@	12/14/88	Phys Ed Supply	14	03610	1645.50	0
803118	North Star@	12/14/88	Volleyball Equi	14	12750	608.75	0
803186	Edu Media	02/09/89	Films	14	03610	11.26	0
803190	Alta Basketbal	02/21/89	Rule Books	14	03610	16.85	0
803230	Beaupre Busing	03/29/89	Phys Ed Trips	14	11410	500.00	0
803291	Timberjack	05/17/89	Trackmeet Suppl	14	03610	25.77	0
803301	Sup. Trans	12/14/88	Curling	14	11410	25.25	0
803301	Sup. Trans	12/14/88	Sup Trans	14	11410	22.80	0
803301	Sup. Trans	12/14/88	Sup Trans	14	11410	22.80	0

Paid Invoice Details -- Physical Ed. 06/11/89

Paid Invoice Total . . . 252.00

a->PO_NO	b->VENDOR	a->INV_NO	a->DESC	a->S_NO	a->Y_NO	a->AMT	b->CLOSE
803273	Ron's Outdoor	1685	Cooper Goals	14	12751	252.00	C

Printed on: 06/11/89 Physical Ed.

Total Budget Commitment . . 2120.98

Budget Allocation 1500.00

Projected Department Balance: -1630.98