Department of Applied Analysis
and Computer Science

Technical Report CSTR-1015

September 1972

SACL

Student Assisted Class List

by

Leroy J. Dickey
Department of Pure Mathematics

SACL

Student Assisted

Class List

by

Leroy J. Dickey

This work was supported by the National Research Council of Canada.

SACL

Student Assisted

Class List

by

Leroy J. Dickey

This work was supported by the National Research Council of Canada.

I. Introduction

The *SACL* system is designed to accompany the *CLASSPAK* system for handling classroom records [1]. The functions in *SACL* permit the student to enter his own name and section numbers into a file which the teacher can then move into his own workspace containing the *CLASSPAK DATA*.

It is expected that *SACL* will be of the most use to those who teach large first year classes. In this context, *SACL* has two advantages. It gives the new student an exposure to a computer terminal system, and it relieves the teacher of a great deal of work involved in entering the student names into *CLASSPAK*.

II. Description of *START*, *SIGNUP* and *FETCH*.

The function

START

is used after *INITIALIZE* and *ADD S* (in *CLASSPAK*) have been used. *START* asks for a file name, creates two files, and asks for course name and lecturer's name. These names and the section information (if any was entered via *ADD S*) are stored in the first of the two files and are passed to the student when he uses the function

SIGNUP

which is in a workspace that has been prepared for the students by the teacher. This function asks the student 1) if he wants to enroll for the course and lecturer listed, and, if he says yes, 2) for his name, and 3) which of the listed sections he is in. His responses are appended to the second of the two files created by *START*. The function

FETCH

is used by the teacher to transfer the names from the name file and place

them in his workspace.

Example 1: The first steps with *INITIALIZE* (from *CLASSPAK*) and *START* (from *SACL*).

```
        )LOAD 3 CLASSPAK2
SAVED    9.28.27 07/L8/72
        INITIALIZE
TYPE IN MAXIMUM NUMBERS FOR NAMES, SECTIONS, AND MARK COLUMN TITLES:
□:
        200 7 20
--NUMBER OF CHARACTERS IN A NAME OR COLUMN TITLE:
□:
        23
INITIALIZATION COMPLETE.
PLEASE )ERASE INITIALIZE SETCONST DESCRIBE


        )ERASE INITIALIZE SETCONST DESCRIBE
        ADD S
--SECTION NAMES TO BE ADDED:
BEGINNERS;INTERMEDIATES;MASTERS
BEGINNERS ↔ SECTION 1
INTERMEDIATES ↔ SECTION 2
MASTERS ↔ SECTION 3
        )COPY 555 SACL STARTER
SAVED  10.42.12 31/08/72
        START

THIS FUNCTION SHOULD BE RUN ONLY AFTER THE INITIALIZE FUNCTION ( 3 CLASSPAK )
HAS BEEN RUN.  ALSO, ANY SECTIONS YOU MAY WISH TO SPECIFY SHOULD BE ENTERED
VIA THE COMMAND ADD S.  THESE SECTIONS SHOULD NOT BE ALTERED UNTIL AFTER
ALL USES OF THE FETCH FUNCTION ARE COMPLETED.

ARE ALL SECTIONS ENTERED?  YES

NAME FOR FILE RECORDS:  111 CHESS
111 CHESS← NAME.  OK?  YES

WHAT IS THE COURSE NAME?  CHESS FOR ALL
CHESS FOR ALL← COURSE NAME.  OK?  NO

WHAT IS THE COURSE NAME?  SUPER CHESS
SUPER CHESS← COURSE NAME.  OK?  Y

WHO IS THE LECTURER?  B. FISCHER
B.FISCHER← LECTURER.  OK?  YES
THE NAMES OF THE CREATED FILES ARE:
        111 CHESS1
        111 CHESS2

PLEASE EXECUTE THIS SEQUENCE OF COMMANDS:
```

```
)ERASE START IO          (←NO LONGER NEEDED)
)SAVE NN:LL              (←PICK A NAME, NN, AND A LOCK, LL,  THAT YOU
                           CAN REMEMBER)
)CLEAR                   (←MAKE A CLEAN WS)
)COPY 555 SACL STU       (←GETS SIGNUP FUNCTION)
)COPY NN:LL  FN          (←USE  NN  AND  LL  AS BEFORE AND THE LETTERS  'FN')

)SEAL                    (←FOR SECURITY)
)SAVE XX   NAME          (←PICK A LIBRARY NUMBER, XX, AND NAME, NAME, FOR
                           WORKSPACE THAT WILL BE USED BY STUDENTS.)O

        )ERASE START IO
        )WSID KING:CHECK
WAS 3 CLASSPAK2
        )SAVE
    16.41.55 01/09/72 KING
        )CLEAR
CLEAR WS
        )COPY 555 SACL STU
SAVED  10.42.12 31/08/72
        )COPY KING:CHECK FN
SAVED  16.41.55 01/09/72
        )SEAL
        )SAVE 555 QUEEN
    16.42.58 01/09/72
        )OFF
```

Example 2: The *SIGNUP* function used by the student.

```
        )LOAD 555 QUEEN
SAVED  16.42.58 01/09/72
        SIGNUP
DO YOU WANT TO SIGN UP FOR SUPER CHESS WITH B. FISCHER?  YES

PLEASE:  YOUR FAMILY NAME, A COMMA, AND YOUR INITIALS:
DONALDSON, D.D.
DONALDSON, D.D.← YOUR NAME. OK? YES

WHICH OF THESE SECTIONS ARE YOU IN?
        1← BEGINNERS
        2← INTERMEDIATES
        3← MASTERS
ENTER NUMBERS (←) PLEASE: 2

2--INTERMEDIATES
↑ SECTIONS. OK?  Y
THANK YOU.

        SIGNUP
DO YOU WANT TO SIGN UP FOR SUPER CHESS WITH B. FISCHER?  Y

PLEASE:  YOUR FAMILY NAME, A COMMA, AND YOUR INITIALS:
DUCK, DONALD
DUCK, DONALD ← YOUR NAME. OK?  NO
```

```
PLEASE:  YOUR FAMILY NAME, A COMMA, AND YOUR INITIALS:
DUCK, D.
DUCK, D. ← YOUR NAME. OK?  Y

WHICH OF THESE SECTIONS ARE YOU IN?
        1← BEGINNERS
        2← INTERMEDIATES
        3← MASTERS
ENTER NUMBERS (←) PLEASE:  1

1--BEGINNERS
↑ SECTIONS. OK?  Y
THANK YOU.


        SIGNUP
DO YOU WANT TO SIGN UP FOR SUPER CHESS WITH B. FISCHER?  Y

PLEASE:  YOUR FAMILY NAME, A COMMA, AND YOUR INITIALS:
WEDDERBIRM, JOHN HENRY MILLFORD
WEDDERBIRM, JOHN HENRY ← YOUR NAME, (IT WAS TRUNCATED.)  OK?  NO

PLEASE:  YOUR FAMILY NAME, A COMMA, AND YOUR INITIALS:
WEDDERBURN, J.H.M.
WEDDERBURN, J.H.M. ← YOUR NAME.  OK?  Y

WHICH OF THESE SECTIONS ARE YCU IN?
        1← BEGINNERS
        2← INTERMEDIATES
        3← MASTERS
ENTER NUMBERS (←) PLEASE:  3

3-- MASTERS
↑ SECTIONS.  OK?  Y
THANK YOU
```

Example 3: The *FETCH* function used by the teacher.

```
        )LOAD KING:CHECK
SAVED 16.41.55 01/09/72
        FETCH
3 NAMES ADDED TO CLASS LIST.
DO NOT FORGET TO  )SAVE
        LIST ALL  N

NO. STUDENT NAMES (3)

001  DONALDSON, D.D.
002  DUCK, D.
003  WEDDERBURN, J.H.M.
---END---

        )SAVE
      9.20.58. 05/09/72 KING
```

II . Description of the auxiliary functions, *CLEANUP*, *SCANFILE*, and *SCRUBFILES*.

The names in the name file are not removed by the function *FETCH* when they are transferred to the workspace containing *DATA*. Instead the lowest numbered component of the file contains an integer which is the number of the next component that will be transferred. After you are convinced that the data has been successfully transferred from the file to the workspace, the function

CLEANUP

can be used to delete from the file those names which have been transferred. This should be done from time to time in order to keep the size of the file no larger than necessary.

You can display the contents of either of the two files at any time by typing the command

SCANFILE i (i is either 1 or 2).

This could be useful to you if there are more entries in the file than can be put into your workspace containing *DATA*. If you wish to see only a few of the components of file i, then the command

SCANFILE i, j, k

will cause the components with numbers between j and k to be listed.

After all names have been entered into *CLASSPAK*, there will be no further need for the files. The files should be deleted from the *APL* system. The function

SCRUBFILES

will remove the files. At this time you will also want to remove the workspace containing *SIGNUP* that you designed for the student. This can

be done by the system command

)*DROP* *XX* *NAME*

Of course, you must use the same number, *XX*, and name, *NAME*, that you used when you stored the workspace.

Example 4: The *CLEANUP*, *SCANFILE*, and *SCRUBFILES* functions.

```
        )LOAD 555 SACL
SAVED 10.42.12  31/08/72
        )COPY KING:CHECK FN
SAVED  9.20.58  05/09/72
        SCANFILE 1
        111 CHESS1
(1) 10609  09/01/72  16:39:45:41  HOW TO PLAY CHESS
(2) 10609  09/01/72  16:39:45:52  B.  FISCHER
(3) 10609  09/01/72  16:39:45:59
BEGINNERS
INTERMEDIATES
MASTERS
        SCANFILE 2
        111 CHESS2
(1) 10609  09/05/72  09:20:36:06  5
(2) 30442  09/01/72  16:47:45:30  DONDALDSON, D.D.          010
(3) 30442  09/01/72  16:49:01:13  DUCK, D.                  100
(4) 30442  09/01/72  16:51:49:48  WEDDERBURN, J.H.M.        001
        CLEANUP
3 ENTRIES DROPPED FROM FILE.
        SCANFILE 2
        111 CHESS2
(4) 10609  09/05/72  09:22:55:18  5

        SCRUBFILES
ERASED:           111 CHESS1
ERASED:           111 CHESS2
        )DROP 555 QUEEN
      10:55:37  05/09/72
```

III. Security

It is possible for the student to learn the name of the file where his name temporarily resides. In example 1, this is 111 *CHESS2*. It is always given by the second line of the variable *FN*. However, his

use of the file is restricted to appending only, and this, only with the use of a lock number. The lock number for the file is contained inside the function and cannot be displayed because it has been sealed. Even if the lock number becomes known, the worst that can happen to the file is that unusable data could be appended.

Your workspace with *DATA* from *CLASSPAK* is secure as long as you keep the workspace name and lock a secret. (This was *KING:CHECK* in example 1).

You can create files that have the same library number as your own account number or with any public library number (that is, a number between 1 and 999). It is recommended that you specify a public account number for your files. In this way, your account number is not a part of the file name. Likewise, the workspace that the student loads should be put into a public library rather than into your own.

All access to the two files is controlled by means of an access matrix. The lock numbers specified are different for the creator of the file than for all other users. For this reason, you must use the same account number every time you do any of the functions *FETCH*, *CLEANUP*, *SCANFILE*, or *SCRUBFILES* that you used when you executed the function *START*. If you wish to try the function *SIGNUP*, you will be able to do so with any account number other than your own.

IV. Space problems.

In order to run the function *START* it is necessary to have only the group of variables called *DATA* from your workspace (called *KING:CHECK* in example 1). To run the function *FETCH* both *DATA* and *FN* are necessary. *FN* is a matrix giving the names of the files containing your class information

and names of students.

For example, if you try to execute *START* and receive the message *WSFULL*, this sequence of steps may help.

```
→
)ERASE  STARTER
)SAVE  NN:LL
)CLEAR
)COPY  NN:LL  DATA
)COPY 555 SACL STARTER
START
)WSID CONTINUE
)SAVE CONTINUE
)LOAD NN:LL
)COPY CONTINUE FN
)SAVE
)DROP CONTINUE
```

Or, if problems occur while using *FETCH*, this sequence may help.

```
→
)SAVE  NN:LL
)CLEAR
)COPY  NN:LL  DATA FN
)COPY 555 SACL STARTER
FETCH
)WSID CONTINUE
)SAVE CONTINUE
)LOAD  NN:LL
)COPY CONTINUE DATA
)SAVE
)DROP CONTINUE
```

The functions *CLEANUP*, *SCANFILE*, *SCRUBFILES* need only the matrix *FN*, so this sequence will work.

```
)LOAD 555 SACL
)COPY  NN:LL  FN
```

CLEANUP, *SCANFILE*, or *SCRUBFILES*

No saving is necessary or useful after the execution of these functions, since they only affect the files and not the contents of a workspace.

V. Some suggested sequences.

The underlined expressions are substitutions for names, lock, and library number which you will specify.

1. The starting sequence

```
)sign on
)LOAD 3 CLASSPAK2
INITIALIZE
ADD S
)COPY 555 SACL STARTER
START
)WSID NN:LL
)SAVE
```

2. Preparing the workspace for students

```
)CLEAR
)COPY 555 SACL STU
)COPY NN:LL  FN
)SEAL
)SAVE XX  NAME
```

3. The student sign up

```
)number          (sign on)
)LOAD XX  NAME
SIGNUP
)OFF
```

4. Transferring names from file to workspace

```
)LOAD NN:LL
FETCH
)SAVE
```

5. Removing names from the file

```
)LOAD 555 SACL
)COPY NN:LL  FN
CLEANUP
```

6. Ending student signup

```
)LOAD NN:LL
FETCH
)SAVE

)LOAD 555 SACL
)COPY NN:LL  FN
SCRUBFILES

)LOAD NN:LL
)ERASE STARTER
)SAVE

)DROP XX  NAME
```

VI . Bibliography.

[1] The classpak system, APL routines for the automatic manipulation of classroom data, Patrick C. Fischer, University of Waterloo, CSTR 1008 (1970).

[2] APL PLUS File Subsystem Instruction Manual, I.P. Sharp Associates, Limited, Toronto.

[3] APL 360 User's Manual (IBM Manual H20-0683).