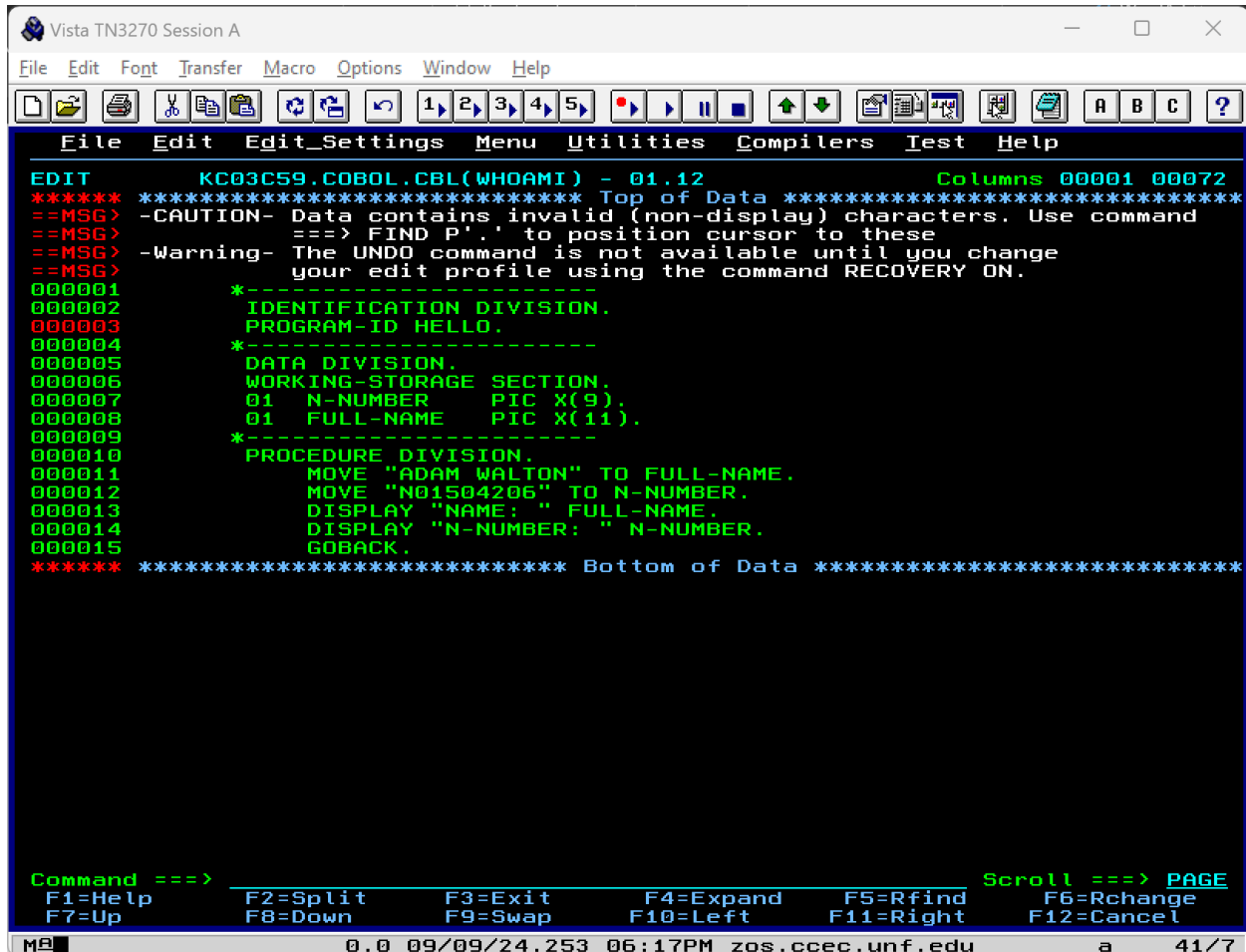


Step 1: Modifying the COBOL program.

After logging in the mainframe and accessing my datasets through ISPF, I opened the COBOL dataset in edit mode and selected "WHOAMI". I then edited the provided code to include data and move statements as shown below.



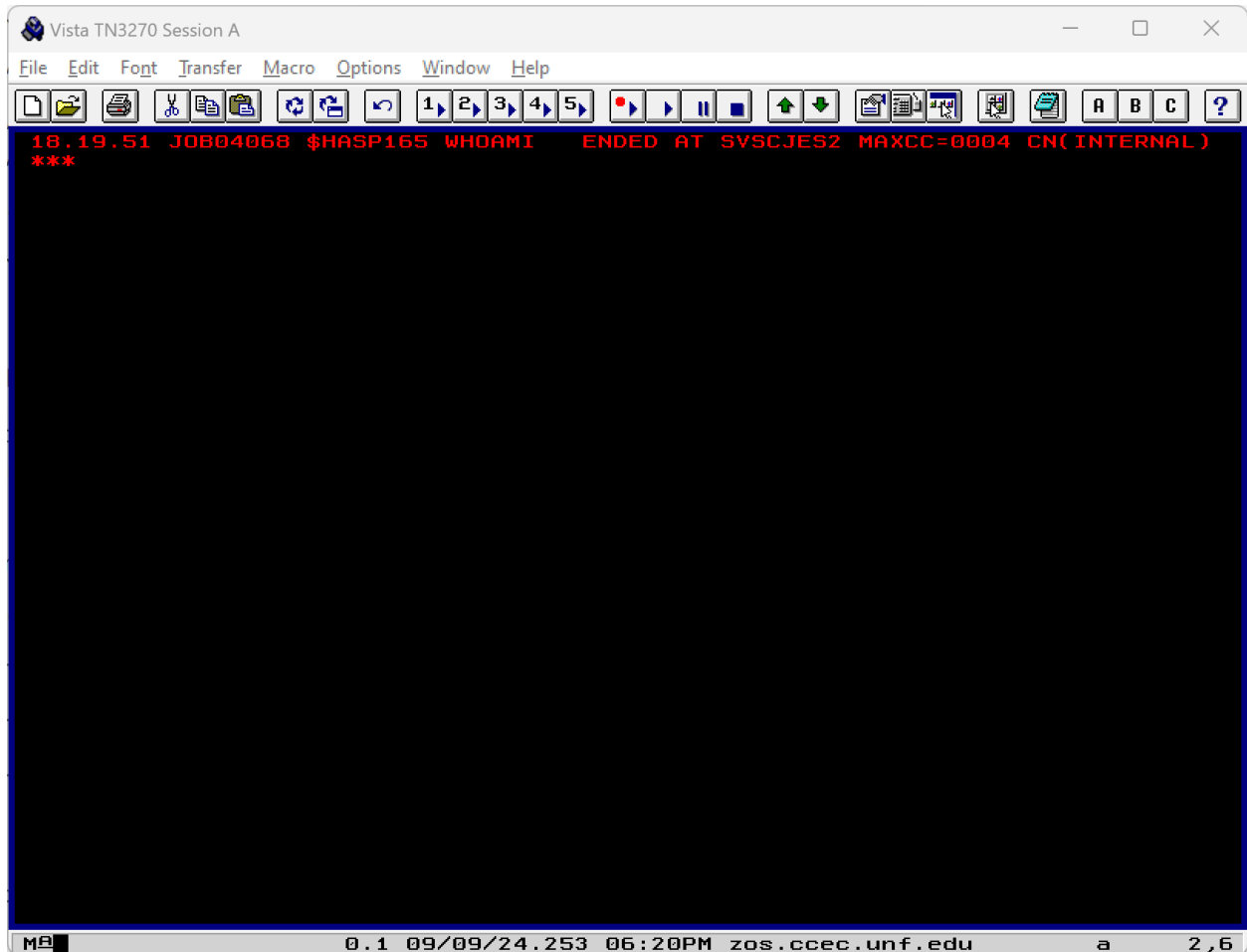
The screenshot shows a mainframe ISPF editor window titled "Vista TN3270 Session A". The window has a menu bar with "File", "Edit", "Font", "Transfer", "Macro", "Options", "Window", and "Help". Below the menu bar is a toolbar with various icons for file operations, editing, and navigation. The main editing area displays a COBOL program being edited. The program is identified as "KC03C59.COBOL.CBL(WHOAMI) - 01.12" and has columns 00001 to 00072. The program content is as follows:

```
EDIT          KC03C59.COBOL.CBL(WHOAMI) - 01.12          Columns 00001 00072
***** ***** Top of Data *****
==MSG> -CAUTION- Data contains invalid (non-display) characters. Use command
==MSG> ==> FIND P'.' to position cursor to these
==MSG> -Warning- The UNDO command is not available until you change
==MSG> your edit profile using the command RECOVERY ON.
000001      *-----*
000002      IDENTIFICATION DIVISION.
000003      PROGRAM-ID HELLO.
000004      *-----*
000005      DATA DIVISION.
000006      WORKING-STORAGE SECTION.
000007      01 N-NUMBER      PIC X(9).
000008      01 FULL-NAME     PIC X(11).
000009      *-----*
000010      PROCEDURE DIVISION.
000011      MOVE "ADAM WALTON" TO FULL-NAME.
000012      MOVE "N01504206" TO N-NUMBER.
000013      DISPLAY "NAME: " FULL-NAME.
000014      DISPLAY "N-NUMBER: " N-NUMBER.
000015      GOBACK.
***** ***** Bottom of Data *****
```

At the bottom of the window, there is a command line and a status bar. The command line shows "Command ==>" followed by a list of function keys and their actions: F1=Help, F2=Split, F3=Exit, F4=Expand, F5=Rfind, F6=Rchange, F7=Up, F8=Down, F9=Swap, F10=Left, F11=Right, F12=Cancel. The status bar shows "0.0 09/09/24.253 06:17PM zos.ccec.unf.edu a 41/7".

Step 2: JCL compilation job results.

Afterwards, I saved my code and opened the JCL dataset in view mode. I then selected “WHOAMI” and typed in the commands “sub; =sd; st” to begin compiling and switch to the job list. After waiting a few seconds and pressing enter, the screen below popped up showing the compilation results.



Step 3: Detailed compiler output.

Once the program was compiled, I selected my job on the job list screen to have it show the compiler output in full, as shown below.

The screenshot shows a terminal window titled "Vista TN3270 Session A" with a menu bar (File, Edit, Font, Transfer, Macro, Options, Window, Help) and a toolbar. The main display area shows the following text:

```

-----
Display Filter View Print Options Search Help
-----
SDSF OUTPUT DISPLAY WHOAMI JOB04068 DSID 2 LINE 0 COLUMNS 02- 81
COMMAND INPUT ==> SCROLL ==> PAGE
*****
JES2 JOB LOG -- SYSTEM 8561 -- NODE
*****
18.19.45 JOB04068 ---- MONDAY, 09 SEP 2024 ----
18.19.45 JOB04068 IRR010I USERID KC03C59 IS ASSIGNED TO THIS JOB.
18.19.45 JOB04068 IEF677I WARNING MESSAGE(S) FOR JOB WHOAMI ISSUED
18.19.47 JOB04068 ICH70001I KC03C59 LAST ACCESS AT 18:08:14 ON MONDAY, SEPTEMB
18.19.47 JOB04068 $HASP373 WHOAMI STARTED - INIT 1 - CLASS A - SYS
18.19.51 JOB04068 -----TIMINGS (MINS.)--
18.19.51 JOB04068 -STEPNAME PROCSTEP RC EXCP CONN TCB SRB C
18.19.51 JOB04068 -COBRUN COBOL 04 14432 178 .00 .00
18.19.51 JOB04068 -COBRUN LKED 00 377 38 .00 .00
18.19.51 JOB04068 -GO 00 36 2 .00 .00
18.19.51 JOB04068 -WHOAMI ENDED. NAME- TOTAL TCB CPU TIM
18.19.51 JOB04068 $HASP395 WHOAMI ENDED - RC=0004
----- JES2 JOB STATISTICS -----
09 SEP 2024 JOB EXECUTION DATE
6 CARDS READ
525 SYSOUT PRINT RECORDS
0 SYSOUT PUNCH RECORDS
34 SYSOUT SPOOL KBYTES
0.09 MINUTES EXECUTION TIME
1 //WHOAMI JOB 1,NOTIFY=&SYSUID
IEFC653I SUBSTITUTION JCL - 1,NOTIFY=KC03C59
2 //COBRUN EXEC IGYWCL
3 XXIGYWCL PROC LNGPRFX='IGY630',
XX LIBPRFX='CEE',
XX PGMLIB='&&GOS&T',GOPGM=GO
XX*
XX*
XX* IBM Enterprise COBOL for z/OS
XX* Version 6 Release 3 Modification 0
XX*
XX* LICENSED MATERIALS - PROPERTY OF IBM.
XX*
XX* 5655-EC6 COPYRIGHT IBM CORP. 1983, 2019
PF 1=HELP 2=SPLIT 3=END 4=RETURN 5=RFIND 6=RCHANGE
PF 7=UP 8=DOWN 9=SWAP 10=LEFT 11=RIGHT 12=RETRIEVE
M0 0.0 09/09/24.253 06:24PM zos.ccec.unf.edu a 4,21

```

Step 4: Batch run output.

I then went back to the job list and typed “?” in front of my job to see a list of sections from the compiler output. Upon selecting the “sysout” option, the following screen was displayed showing the program output from when the job was ran.

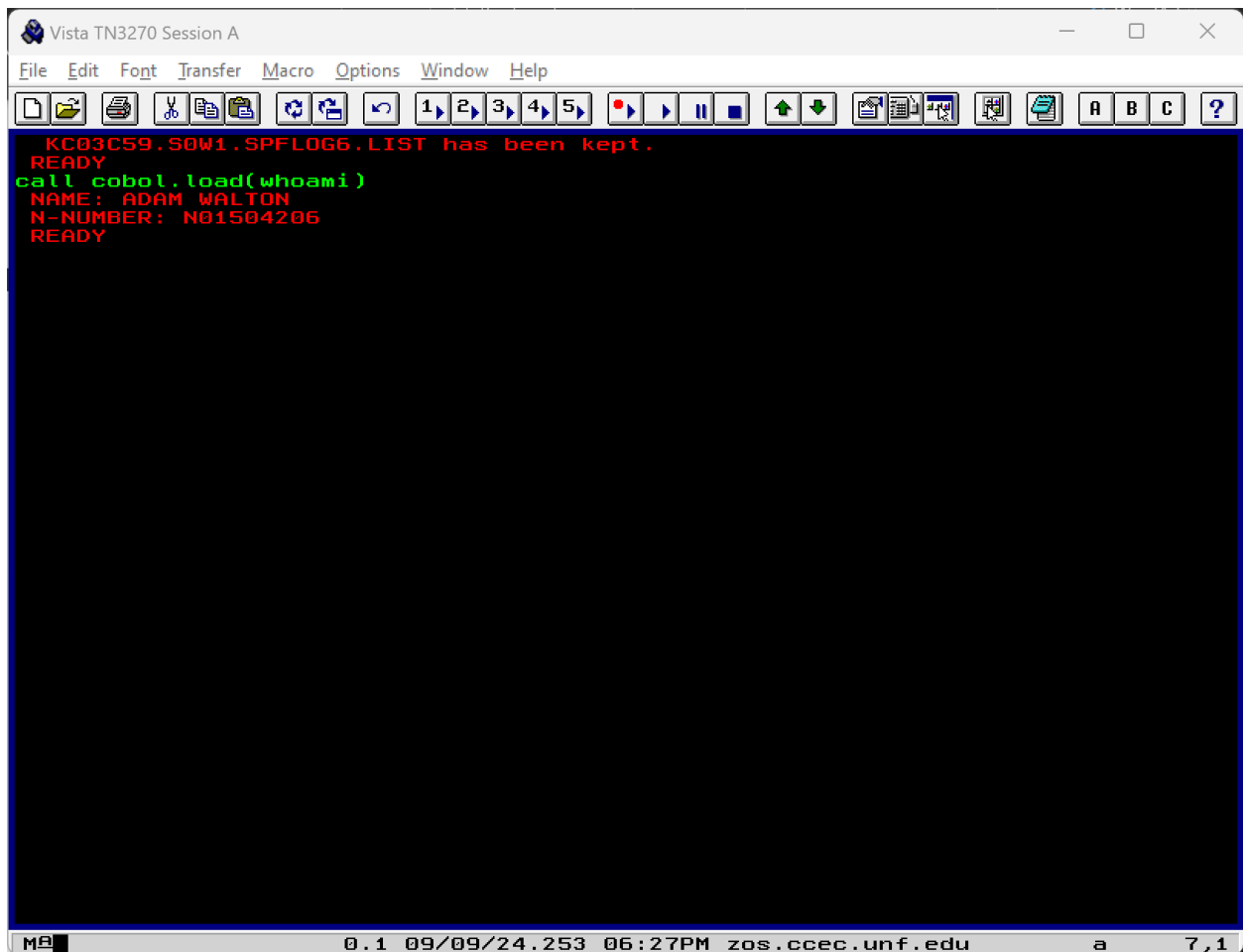
```
-----
Display Filter View Print Options Search Help
-----
SDSF OUTPUT DISPLAY WHOAMI  JOB04068  DSID   103 LINE 0          COLUMNS 02- 81
COMMAND INPUT ==>          SCROLL ==>  PAGE
***** TOP OF DATA *****
NAME: ADAM WALTON
N-NUMBER: N01504206
***** BOTTOM OF DATA *****

PF 1=HELP      2=SPLIT      3=END      4=RETURN      5=RFIND      6=RCHANGE
PF 7=UP        8=DOWN      9=SWAP     10=LEFT     11=RIGHT     12=RETRIEVE

M01 0.0 09/09/24.253 06:25PM zos.ccec.unf.edu a 4,21
```

Step 5: Program output from tso command line (ready prompt).

Finally, I exited out of ISPF to the command line, and entered the command shown below to run my program once again.



The screenshot shows a TSO command line session window titled "Vista TN3270 Session A". The window has a menu bar with "File", "Edit", "Font", "Transfer", "Macro", "Options", "Window", and "Help". Below the menu bar is a toolbar with various icons for file operations, editing, and navigation. The main area of the window displays the following text:

```
KC03C59.S0W1.SPFL0G6.LIST has been kept.  
READY  
call cobol.load(whoami)  
NAME: ADAM WALTON  
N-NUMBER: N01504206  
READY
```

The status bar at the bottom of the window displays the following information: "0.1 09/09/24.253 06:27PM zos.ccec.unf.edu a 7,1".

Comments:

Overall, I didn't struggle too much with this project. I was able to figure out how to create my code, compile, and run it using some examples in the textbook and watching the zoom recording. One thing I did get confused by was thinking the line numbers in the editor represented columns 1-6 of my COBOL program since they are typically for line numbers, which caused errors when compiling since I was starting my lines on column 1 instead of column 7. I was able to correct this by looking at some other examples in my COBOL dataset without asking for assistance. This project has helped me learn how to use the data division in a COBOL program as well as run my code, and better understand how I need to structure my code to ensure I'm putting the right things in the right columns on each line of code.