Department of Computer Science and Engineering

19CSE213 – Operating Systems

Case Study: XOS

Batch: 2021 – 2025

Semester: 4

Team Members:

S. No	Roll No.	Name
1.	CB.EN.U4CSE21401	Abhi Ram
2.	CB.EN.U4CSE21414	Vaibhav
3.	CB.EN.U4CSE21430	K M Sami
4.	CB.EN.U4CSE21439	M Rahul
5.	CB.EN.U4CSE21458	Manjunadh

Introduction:

Project XOS, also known as eXperimental Operating System, is a platform designed for the development of a toy operating system. Its primary purpose is to serve as an instructional tool for students, enabling them to learn and implement OS data structures and functionalities on a simulated machine called XSM (eXperimental String Machine).

The operating system itself is programmed using a custom language called SPL (System Programmer's Language), while application programs that run on the OS are developed using APL (Application Programmer's Language).

Contents of XOS:

XOS consists of four main components:

1. **APL Compiler**: The APL (Application Programmers Language) compiler is used to compile user programs into XSM machine instructions.
2. **SPL Compiler**: The SPL (System Programmers Language) compiler is used to compile system programs, which include operating system routines, into XSM machine instructions.
3. **XFS-INTERFACE**: This directory contains the XFS-INTERFACE (eXperimental File System). It provides an interface that allows files from a UNIX machine to be loaded into the File System of XSM. The interface also offers options for disk formatting, listing loaded files, removing files, copying blocks to a UNIX file, and displaying files within the XFS disk.
4. **XSM Machine Simulator**: The "xsm" directory contains the XSM (eXperimental String Machine) simulator. This simulator emulates the behavior of the XSM machine, providing a simulated environment for executing programs and testing the functionality of the operating system.

RoadMap:

Step1: Setting up the System

We have downloaded the XOS from the official page:

https://xosnitc.github.io/downloads.html

Then we extracted the content and stored it in the home directory.

```
rocky@LAPTOP-8ISMJOM2:~/Downloads$ ls
myxos-1.0.1.tar.gz  xos.gz
rocky@LAPTOP-8ISMJOM2:~/Downloads$ tar -xvf xos.gz
myxos-1/
myxos-1/block.txt
myxos-1/doc/
myxos-1/prog.txt
myxos-1/data.txt
myxos-1/code.txt
myxos-1/spl/
myxos-1/xfs-interface/xfs-interface
myxos-1/xfs-interface/AUTHORS
myxos-1/xfs-interface/createDisk.c
myxos-1/1.txt
rocky@LAPTOP-8ISMJOM2:~/Downloads$ ls
myxos-1  myxos-1.0.1.tar.gz  xos.gz
rocky@LAPTOP-8ISMJOM2:~/Downloads$
```

We also need to install flex and bison for XOS to work properly.

```
rocky@LAPTOP-8ISMJOM2:~/Downloads$ cd -r myxos-1 ~/myxos
rocky@LAPTOP-8ISMJOM2:~/Downloads$ sudo apt install flex bison
[sudo] password for rocky:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
bison is already the newest version (2:3.8.2+dfsg-1build1).
flex is already the newest version (2.6.4-8build2).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
rocky@LAPTOP-8ISMJOM2:~/Downloads$
```

```
rocky@LAPTOP-8ISMJOM2:~$ ls
Downloads  myexpos  myxos
rocky@LAPTOP-8ISMJOM2:~$ cd myxos/
rocky@LAPTOP-8ISMJOM2:~/myxos$ ls
1.txt     apl        data.txt    doc       sample.dat   xfs-interface
Makefile  block.txt  data11.txt  fat1.txt  sample2.dat  xsm
README    code.txt   disk2.txt   prog.txt  spl
rocky@LAPTOP-8ISMJOM2:~/myxos$
```

Step2: Understanding the File System

Commands Used:

./xfs-interface

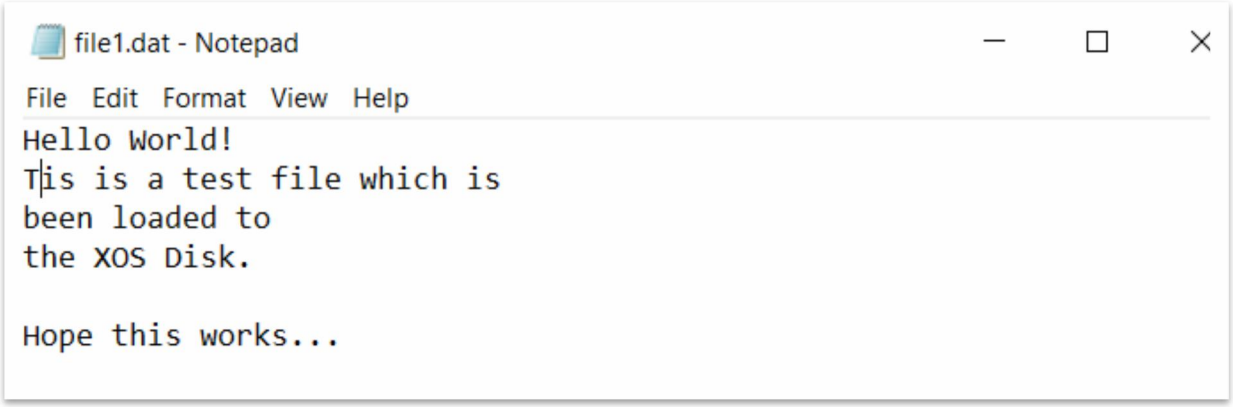
fdisk

df

```
rocky@LAPTOP-8ISMJOM2:~/myxos$ cd xfs-interface/
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$ ./xfs-interface
Unix-XFS Interace Version 1.0.
Type "help" for  getting a list of commands.
# fdisk
Formatting Complete. "disk.xfs" created.
# df
0       -       1
1       -       1
2       -       1
3       -       1
4       -       1
5       -       1
```

```
508     -       0
509     -       0
510     -       0
511     -       0

No of Free Blocks = 488
Total no of Blocks = 512#
```

nano file1.dat




```
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$ nano file1.dat
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$
```

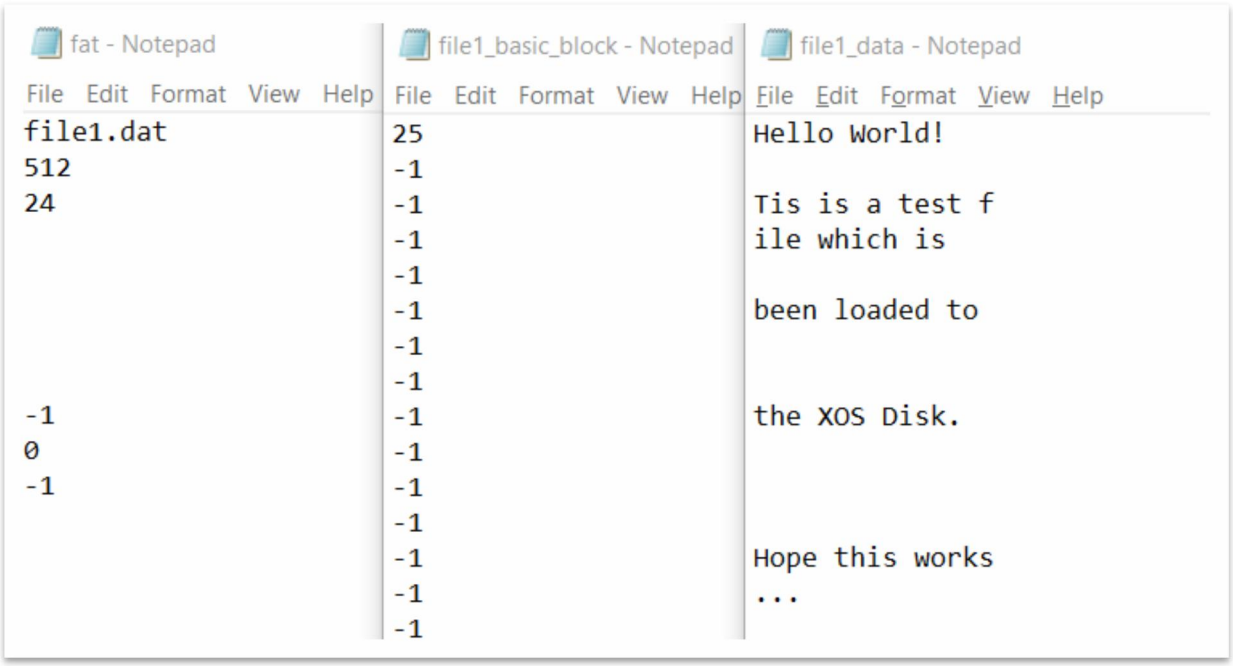
load --data file1.dat

copy 19 19 fat.txt

copy 24 24 file1_basic_block.txt

copy 25 25 file1_data.txt

```
# load --data file1.dat
# df
0        -       1
1        -       1
2        -       1
3        -       1
508      -       0
509      -       0
510      -       0
511      -       0

No of Free Blocks = 486
Total no of Blocks = 512#
#
# copy 19 19 fat.txt
# copy 24 24 file1_basic_block.txt
# copy 25 25 file1_data.txt
#
```



fat - Notepad

File Edit Format View Help

```
file1.dat
512
24




-1
0
-1
```

file1_basic_block - Notepad

File Edit Format View Help

```
25
-1
-1
-1
-1
-1
-1
-1
-1
-1
-1
-1
-1
-1
-1
```

file1_data - Notepad

File Edit Format View Help

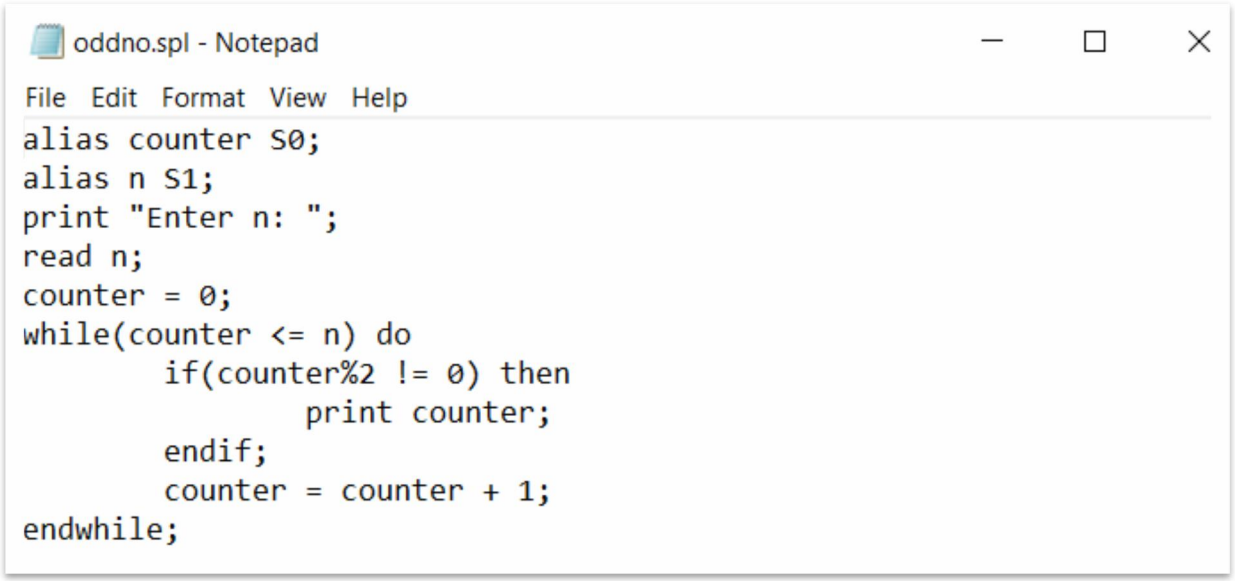
```
Hello World!

Tis is a test f
ile which is

been loaded to


the XOS Disk.




Hope this works
...
```

Step3: Starting the Machine

Commands Used:

nano oddno.spl

./spl --os oddno.spl




```
rocky@LAPTOP-8ISMJOM2:~/myxos$ cd spl/
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ nano oddno.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ ./spl --os oddno.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$
```

./xfs-interface

load --os ../spl/os_startup.xsm

```
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$ ./xfs-interface
Unix-XFS Interace Version 1.0.
Type "help" for  getting a list of commands.
# load --os ../spl/os_startup.xsm
#
```

./xsm

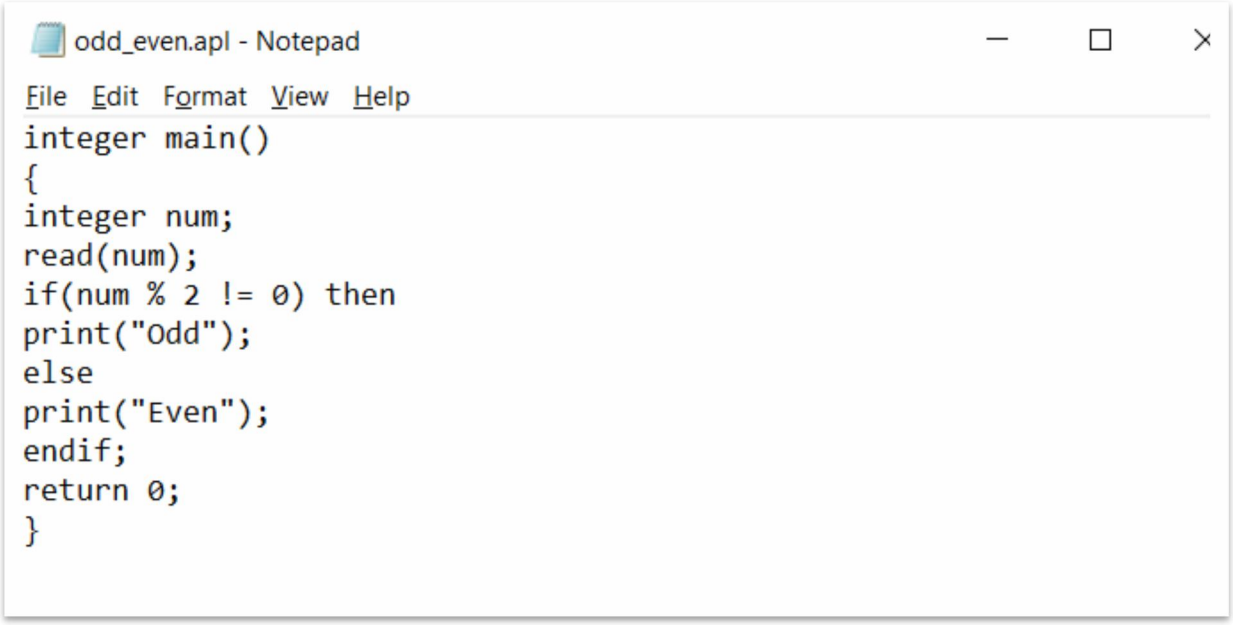
```
rocky@LAPTOP-8ISMJOM2:~/myxos$ cd xsm/
rocky@LAPTOP-8ISMJOM2:~/myxos/xsm$ ./xsm
Enter n:
10
1
3
5
7
9
Machine is halting
rocky@LAPTOP-8ISMJOM2:~/myxos/xsm$
```

Step4: Running a User Program

Commands Used:

nano odd_even.apl

./apl odd_even.apl




```
rocky@LAPTOP-8ISMJOM2:~/myxos$ cd apl/
rocky@LAPTOP-8ISMJOM2:~/myxos/apl$ nano odd_even.apl
rocky@LAPTOP-8ISMJOM2:~/myxos/apl$ ./apl odd_even.apl
rocky@LAPTOP-8ISMJOM2:~/myxos/apl$
```

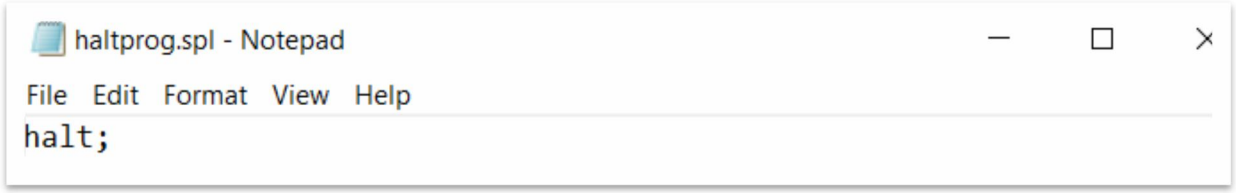
./xfs-interface

load --init ../apl/odd_even.xsm

```
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$ ./xfs-interface
Unix-XFS Interace Version 1.0.
Type "help" for  getting a list of commands.
# fdisk
Formatting Complete. "disk.xfs" created.
# load --init ../apl/odd_even.xsm
#
```

nano haltprog.spl

./spl –int=7 haltprog.spl




```
rocky@LAPTOP-8ISMJOM2:~/myxos$ cd spl/
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ nano haltprog.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ ./spl --int=7 haltprog.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$
```

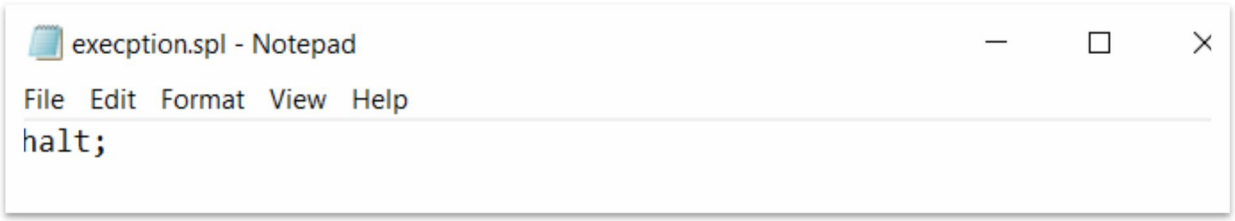
./xfs-interface

load –int=7 ../spl/int7.xsm

```
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$ ./xfs-interface
Unix-XFS Interace Version 1.0.
Type "help" for  getting a list of commands.
# load --int=7 ../spl/int7.xsm
#
```

nano exception.spl

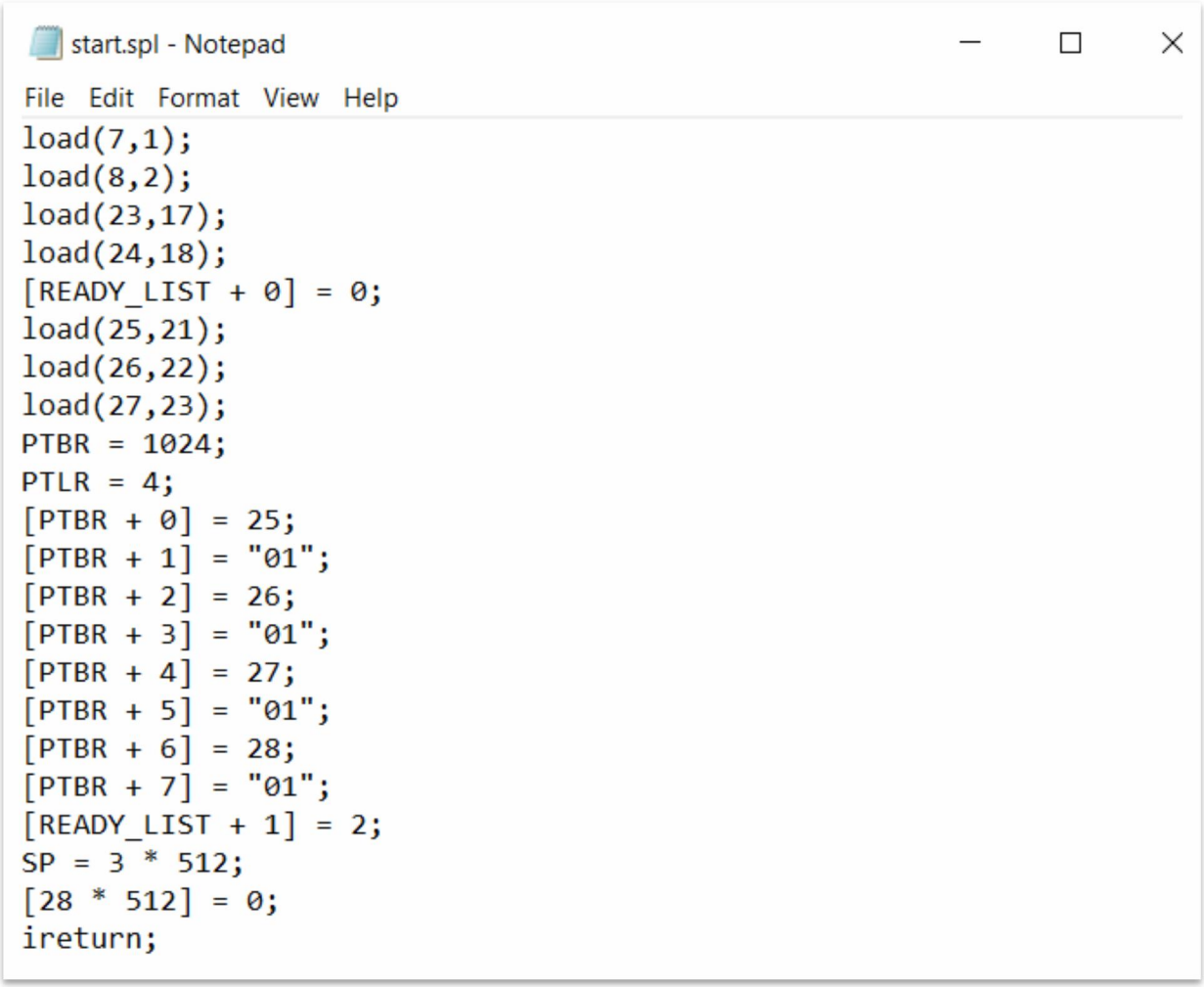
./spl --exhandler exception.spl




```
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ nano execption.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ ./spl --exhandler execption.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$
```

nano start.spl

./spl --os start.spl




```
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ nano start.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ ./spl --os start.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$
```

./xfs-interface

load --os ../spl/os_startup.xsm

```
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$ ./xfs-interface
Unix-XFS Interace Version 1.0.
Type "help" for  getting a list of commands.
# load --os ../spl/os_startup.xsm
# exit
```

./xsm --timer=0

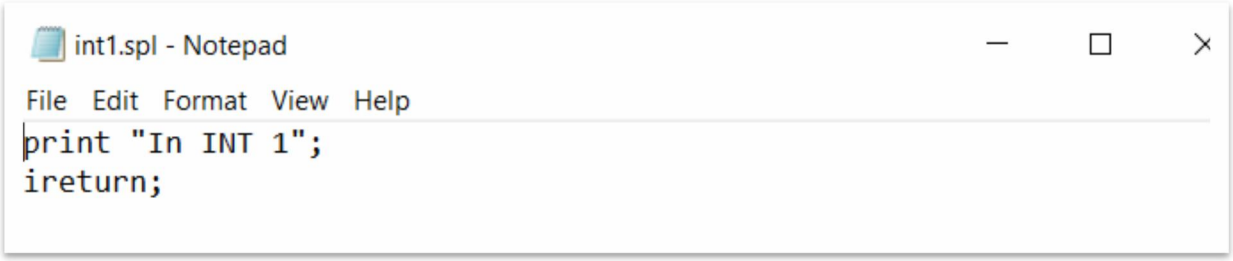
```
rocky@LAPTOP-8ISMJOM2:~/myxos$ cd xsm/
rocky@LAPTOP-8ISMJOM2:~/myxos/xsm$ ./xsm --timer=0
5
Odd
Machine is halting
rocky@LAPTOP-8ISMJOM2:~/myxos/xsm$
```

Step5: Interrupt Routines

Commands Used:

nano int1.spl

./spl --int=1 int1.spl



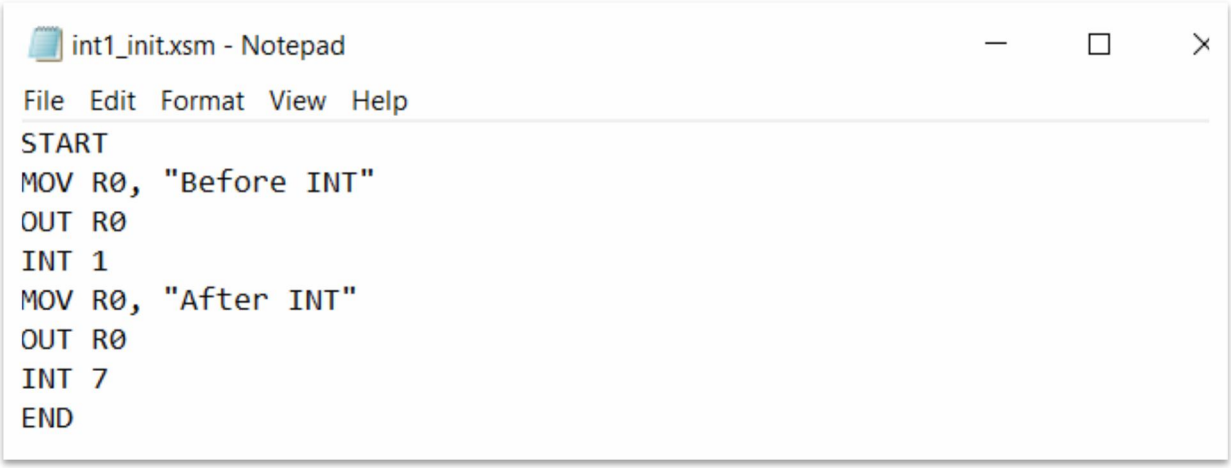

```
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ nano int1.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ ./spl --int=1 int1.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$
```

./xfs-interface

load --int=1 ../spl/int1.xsm

```
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$ ./xfs-interface
Unix-XFS Interace Version 1.0.
Type "help" for  getting a list of commands.
# load --int=1 ../spl/int1.xsm
# exit
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$
```

nano int1_init.xsm




```
rocky@LAPTOP-8ISMJOM2:~/myxos$ cd apl/
rocky@LAPTOP-8ISMJOM2:~/myxos/apl$ nano int1_init.xsm
rocky@LAPTOP-8ISMJOM2:~/myxos/apl$
```

./xfs-interface

load --init ../apl/int1_init.xsm

```
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$ ./xfs-interface
Unix-XFS Interace Version 1.0.
Type "help" for  getting a list of commands.
# load --init ../apl/int1_init.xsm
# exit
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$
```

nano start1.spl

./spl --os start1.spl

start1.spl - Notepad

File Edit Format View Help

```
load(7,1);
load(8,2);
load(23,17);
load(24,18);
[READY_LIST + 0] = 0;
load(25,21);
load(26,22);
load(27,23);
PTBR = 1024;
PTLR = 4;
[PTBR + 0] = 25;
[PTBR + 1] = "01";
[PTBR + 2] = 26;
[PTBR + 3] = "01";
[PTBR + 4] = 27;
[PTBR + 5] = "01";
[PTBR + 6] = 28;
[PTBR + 7] = "01";
[READY_LIST + 1] = 2;
SP = 3 * 512;
[28 * 512] = 0;

load(11,5);
load(12,6);

ireturn;
```

```
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ nano start1.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ ./spl --os start1.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$
```

./xfs-interface

load --os ../spl/os_startup.xsm

```
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$ ./xfs-interface
Unix-XFS Interace Version 1.0.
Type "help" for  getting a list of commands.
# load --os ../spl/os_startup.xsm
# exit
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$
```

./xsm –timer=0

```
rocky@LAPTOP-8ISMJOM2:~/myxos/xsm$ ./xsm --timer=0
Before INT
In INT 1
After INT
Machine is halting
rocky@LAPTOP-8ISMJOM2:~/myxos/xsm$
```

Step6: Getting Started with Multiprogramming

Commands Used:

nano odd.apl

nano even.apl

./apl odd.apl

./apl even.apl

odd.apl - Notepad

File Edit Format View Help

```
integer main()
{
print("1");
print("3");
print("5");
print("7");
print("9");
print("11");
print("13");
print("15");
print("17");
print("19");
return 0;
}
```

even.apl - Notepad

File Edit Format View Help

```
integer main()
{
print("2");
print("4");
print("6");
print("8");
print("10");
print("12");
print("14");
print("16");
print("18");
print("20");
return 0;
}
```

```
rocky@LAPTOP-8ISMJOM2:~/myxos/apl$ nano odd.apl
rocky@LAPTOP-8ISMJOM2:~/myxos/apl$ nano even.apl
rocky@LAPTOP-8ISMJOM2:~/myxos/apl$ ./apl odd.apl
rocky@LAPTOP-8ISMJOM2:~/myxos/apl$ ./apl even.apl
rocky@LAPTOP-8ISMJOM2:~/myxos/apl$
```

./xfs-interface

fdisk

load --init ../apl/odd.xsm

load --exec ../apl/even.xsm

```
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$ ./xfs-interface
Unix-XFS Interace Version 1.0.
Type "help" for  getting a list of commands.
# fdisk
Formatting Complete. "disk.xfs" created.
# load --init ../apl/odd.xsm
# load --exec ../apl/even.xsm
#
```

nano start2.spl

nano multi_timer.spl

./spl --os start2.spl

./spl --int=timer multi_timer.spl

start2 - Notepad

File Edit Format View Help

```
load(7,1);
load(8,2);
load(23,17);
load(24,18);
[READY_LIST + 0] = 0;
load(25,21);
load(26,22);
load(27,23);
PTBR = 1024;
PTLR = 4;
[PTBR + 0] = 25;
[PTBR + 1] = "01";
[PTBR + 2] = 26;
[PTBR + 3] = "01";
[PTBR + 4] = 27;
[PTBR + 5] = "01";
[PTBR + 6] = 28;
[PTBR + 7] = "01";
[READY_LIST + 1] = 2;
SP = 3 * 512;
[28 * 512] = 0;

[READY_LIST+32]=1;

load(29,25);
alias PTBR_Process1 S0;
PTBR_Process1 = 1024 + 1 * 8;
[PTBR_Process1+0]=29;
[PTBR_Process1+1]="01";
[PTBR_Process1+2]=-1;
[PTBR_Process1+3]="00";
[PTBR_Process1+4]=-1;
[PTBR_Process1+5]="00";
[PTBR_Process1+6]=30;
[PTBR_Process1+7]="01";

[READY_LIST+33]=1;
[READY_LIST+37]=PTBR_Process1;
[READY_LIST+38]=4;
[READY_LIST+34]=3*512;
[READY_LIST+35]=3*512;
```

multi_timer - Notepad

File Edit Format View Help

```
alias currentPID S0;
alias newPID S10;
alias newPCB S11;
alias status S12;
currentPID = (PTBR - 1024) / 8;

alias currentPCB S1;
currentPCB = READY_LIST + 32 * currentPCB;
[currentPCB + 1] = 1;
[currentPCB + 2] = BP;
[currentPCB + 3] = SP - 1;
alias physicalSP S2;
physicalSP = ([PTBR + 2 * (SP / 512)] * 512) + (SP % 512);
[ currentPCB + 4 ] = [ physicalSP ];
[ currentPCB + 5 ] = PTBR;
[ currentPCB + 6 ] = PTLR;
[ currentPCB + 7 ] = R0;
[ currentPCB + 8 ] = R1;
[ currentPCB + 9 ] = R2;
[ currentPCB + 10 ] = R3;
[ currentPCB + 11 ] = R4;
[ currentPCB + 12 ] = R5;
[ currentPCB + 13 ] = R6;
[ currentPCB + 14 ] = R7;

newPID=currentPID;
while( READY_LIST + 32*newPID <=2560) do
        newPID=newPID+1;
        if (newPID==32) then
                newPID=0;
        endif;
        newPCB=READY_LIST+32*newPID;
        status=[ newPCB+1];
        if(status ==1 ) then
                R0 = [ newPCB + 7 ];
                R1 = [ newPCB + 8 ];
                R2 = [ newPCB + 9 ];
                R3 = [ newPCB + 10 ];
                R4 = [ newPCB + 11 ];
                R5 = [ newPCB + 12 ];
                R6 = [ newPCB + 13 ];
```

```
[READY_LIST+36]=0;
load(9,3);
load(10,4);

ireturn;
```

```
                R7 = [ newPCB + 14 ];
                BP = [ newPCB + 2 ];
                SP = [ newPCB + 3 ];
                PTBR = [ newPCB + 5 ];
                PTLR = [ newPCB + 6 ];
                SP = SP + 1;
                alias newphysicalSP S2;
                newphysicalSP = ([PTBR + 2 * (SP / 512)] * 512) + (SP % 512);
                breakpoint;
                [ newphysicalSP ] = [ newPCB + 4 ];
                [ newPCB + 1 ] = 2;
                ireturn;
        endif;
endwhile;

ireturn;
```

```
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ nano start2.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ nano multi_timer.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ ./spl --os start2.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$ ./spl --int=timer multi_timer.spl
rocky@LAPTOP-8ISMJOM2:~/myxos/spl$
```

./xfs-interface

load --os ../spl/os_startup.xsm

load --int=timer ../spl/timer.xsm

```
rocky@LAPTOP-8ISMJOM2:~/myxos/xfs-interface$ ./xfs-interface
Unix-XFS Interace Version 1.0.
Type "help" for  getting a list of commands.
# load --os ../spl/os_startup.xsm
# load --int=timer ../spl/timer.xsm
#
```

./xsm

```
rocky@LAPTOP-8ISMJOM2:~/myxos/xsm$ ./xsm
1
3
2
4
5
6
7
9
8
11
10
13
12
14
15
16
18
17
20
19
```

The New Version of XOS – eXpOS:

Project eXpOS, is the latest and improved version XOS. eXpOS and the previous version XOS are projects created by students to understand the working of an Operating System. The primary purpose of eXpOS is to provide students with an educational tool through which they can grasp and implement essential OS data structures and functionalities. This is achieved by utilizing a simulated machine known as XSM (eXperimental String Machine).

The operating system is developed using a specialized language called SPL (System Programmer's Language), while application programs that operate on the OS are built using APL (Application Programmer's Language).

Installation of eXpOS:

First, we need to download the eXpOS files. We can find them in their official page:

http://exposnitc.github.io/support_tools-files/setting-up.html

Before installing eXpOS we need to have libreadline-dev, flex, bison, make and gcc.

```
rocky@LAPTOP-8ISMJOM2:~/Downloads$ sudo apt-get install libreadline-dev flex bison make gcc wget curl
[sudo] password for rocky:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
bison is already the newest version (2:3.8.2+dfsg-1build1).
flex is already the newest version (2.6.4-8build2).
gcc is already the newest version (4:11.2.0-1ubuntu1).
libreadline-dev is already the newest version (8.1.2-1).
make is already the newest version (4.3-4.1build1).
wget is already the newest version (1.21.2-2ubuntu1).
curl is already the newest version (7.81.0-1ubuntu1.10).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
rocky@LAPTOP-8ISMJOM2:~/Downloads$
```

```
rocky@LAPTOP-8ISMJOM2:~/Downloads$ curl -sSf https://raw.githubusercontent.com/eXpO
SNitc/expos-bootstrap/main/download.sh | sh
rocky@LAPTOP-8ISMJOM2:~$ cd myexpos/
rocky@LAPTOP-8ISMJOM2:~/myexpos$ ls
Makefile  download.sh  expl  spl  xfs-interface  xsm
rocky@LAPTOP-8ISMJOM2:~/myexpos$
```