READ ME FILE FOR HEAD

Head.c code description:

The program starts by including necessary header files, including "types.h", "stat.h", "user.h", and "fcntl.h", which likely contain various system call definitions and constants.

emptyString(char *str) is a utility function to empty a character array by filling it with null characters ('\0'). This function is used to clear temporary string buffers.

zerocase(char *argv[]) is a function that handles the case when no arguments are provided. It reads 14 lines from standard input (stdin) and prints them using the printf function. It then exits.

firstcase(int y, char *argv[]) handles the case when there are two arguments. It assumes the first argument is a filename, opens that file, and reads y lines from it (where y is initially set to 14) and prints them. It uses a temporary buffer temp to accuwe mulate lines until a newline character is encountered, at which point it prints the line and clears the buffer. This process continues until y lines have been printed, or the end of the file is reached.

secondcase(char *argv[]) handles the case when there are four arguments. It assumes the first argument is -n, the second argument is a number l, and the third argument is a filename. It opens the file specified in the third argument, reads l lines from it, and prints them in the same way as the first case.

thirdcase(int g, char *argv[], int a1, int a2) handles the case when there are three or five arguments. It assumes that g is the number of lines to read, and a1 and a2 are indices of the filenames in the argv array. It opens the two files specified in the arguments, reads g lines from each file, and prints them. It uses separate temporary buffers (temp and temp2) for each file to accumulate lines and print them when newline characters are encountered.

In the main function, the program first checks the number of command-line arguments (argc) to determine which case to execute.

If argc is 1, it calls zerocase.

If argc is 2, it assumes the second argument is a filename and calls firstcase.

If argc is 4, it checks if the first argument is -n and calls secondcase.

If argc is 3 or 5, it calls thirdcase with appropriate arguments.

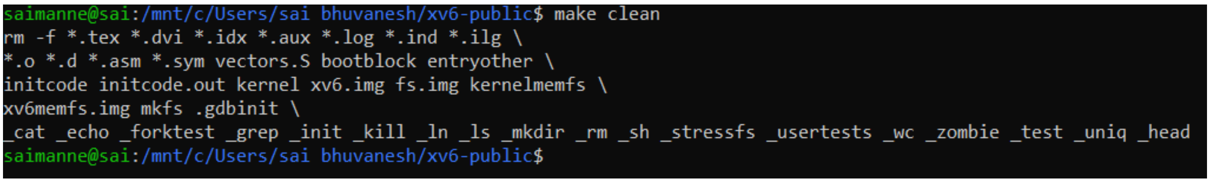
Each case processes and prints lines from one or two files based on the input parameters. Finally, the program exits.

Steps for execution:

1.we use wsl command to go to ubuntu.

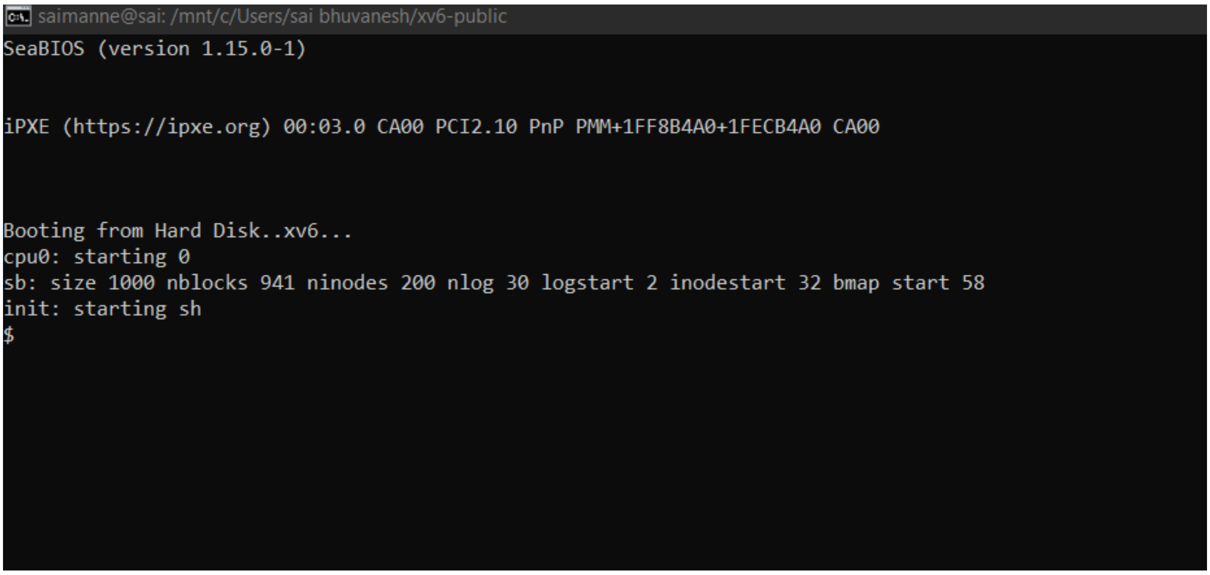
2.we change the directory to cd xv6-public

3.we use the command make clean



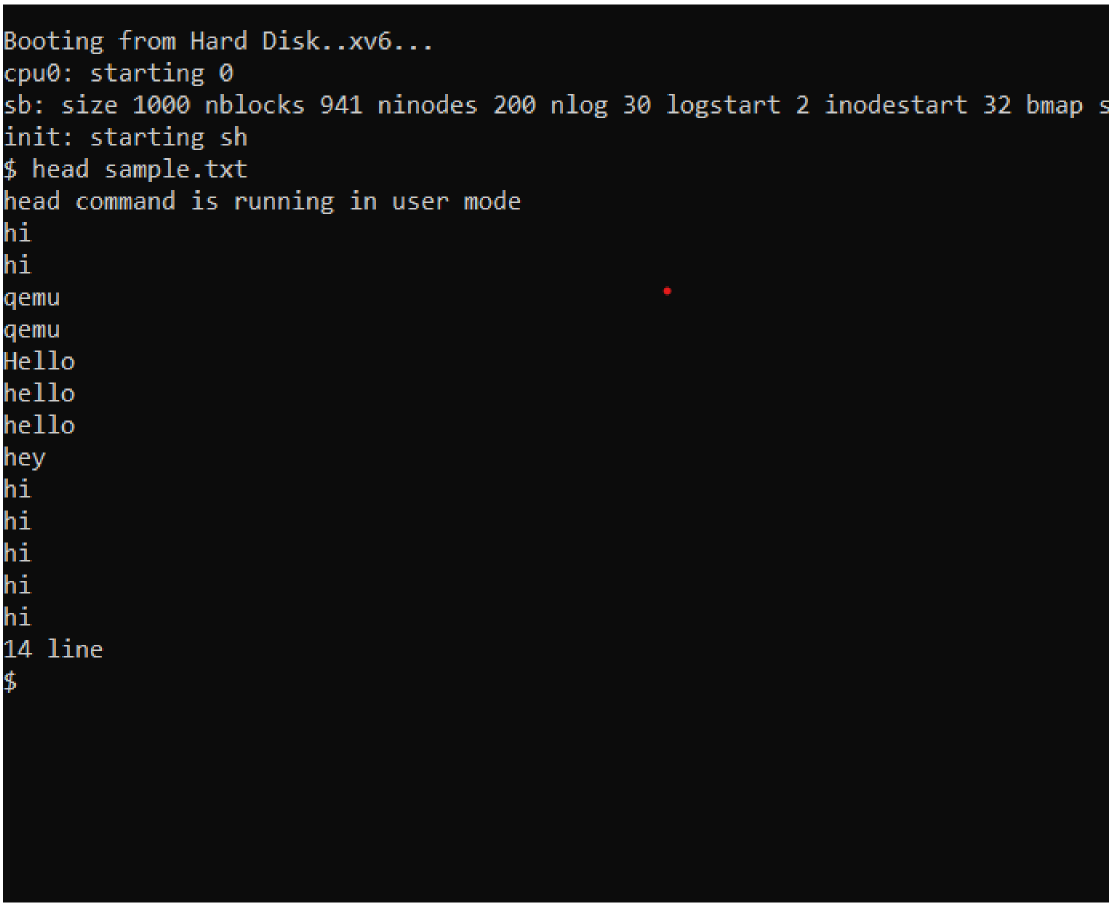
```
saimanne@sai:/mnt/c/Users/sai bhuvanesh/xv6-public$ make clean
rm -f *.tex *.dvi *.idx *.aux *.log *.ind *.ilg \
*.o *.d *.asm *.sym vectors.S bootblock entryother \
initcode initcode.out kernel xv6.img fs.img kernelmemfs \
xv6memfs.img mkfs .gdbinit \
_cat _echo _forktest _grep _init _kill _ln _ls _mkdir _rm _sh _stressfs _usertests _wc _zombie _test _uniq _head
saimanne@sai:/mnt/c/Users/sai bhuvanesh/xv6-public$
```

4.we use the command make qemu-nox



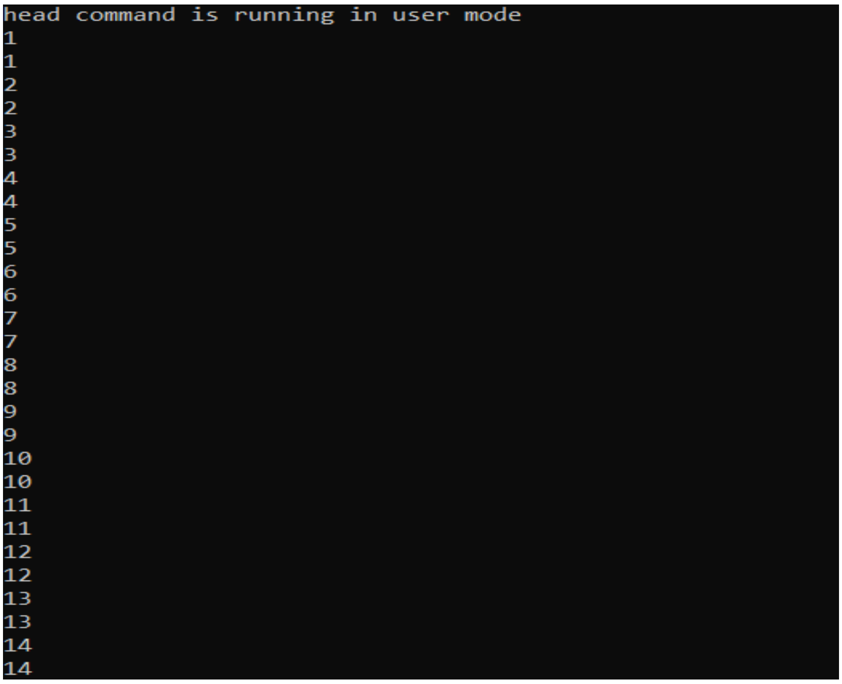
```
SeaBIOS (version 1.15.0-1)


iPXE (https://ipxe.org) 00:03.0 CA00 PCI2.10 PnP PMM+1FF8B4A0+1FECB4A0 CA00



Booting from Hard Disk..xv6...
cpu0: starting 0
sb: size 1000 nblocks 941 ninodes 200 nlog 30 logstart 2 inodestart 32 bmap start 58
init: starting sh
$
```

5.we can use head sample.txt which prints 14 lines of the file if presant



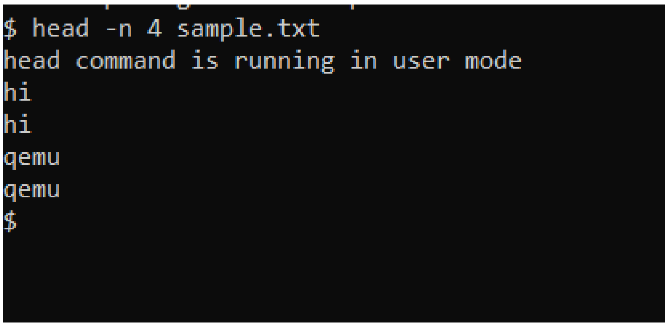
```
Booting from Hard Disk..xv6...
cpu0: starting 0
sb: size 1000 nblocks 941 ninodes 200 nlog 30 logstart 2 inodestart 32 bmap s
init: starting sh
$ head sample.txt
head command is running in user mode
hi
hi
qemu
qemu
Hello
hello
hello
hey
hi
hi
hi
hi
hi
14 line
$
```

6.inorder to take lines from standard input we use head as command and enter the inputs.



```
head command is running in user mode
1
1
2
2
3
3
4
4
5
5
6
6
7
7
8
8
9
9
10
10
11
11
12
12
13
13
14
14
```

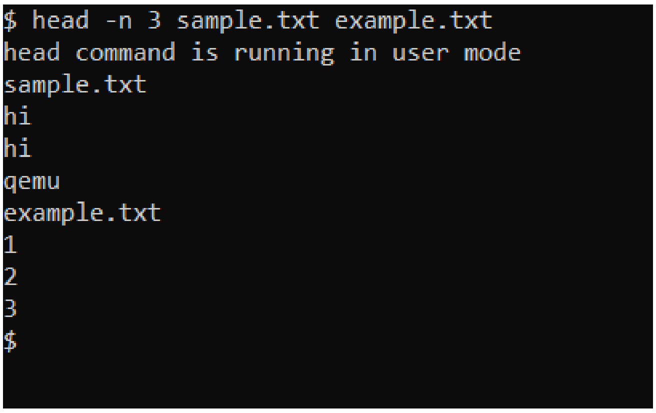
7.we can specify the number of lines to be read from the file as head -n “number” “file name”



```
$ head -n 4 sample.txt
head command is running in user mode
hi
hi
qemu
qemu
$
```

8.we can specify the number of lines to be read from two different files as follows

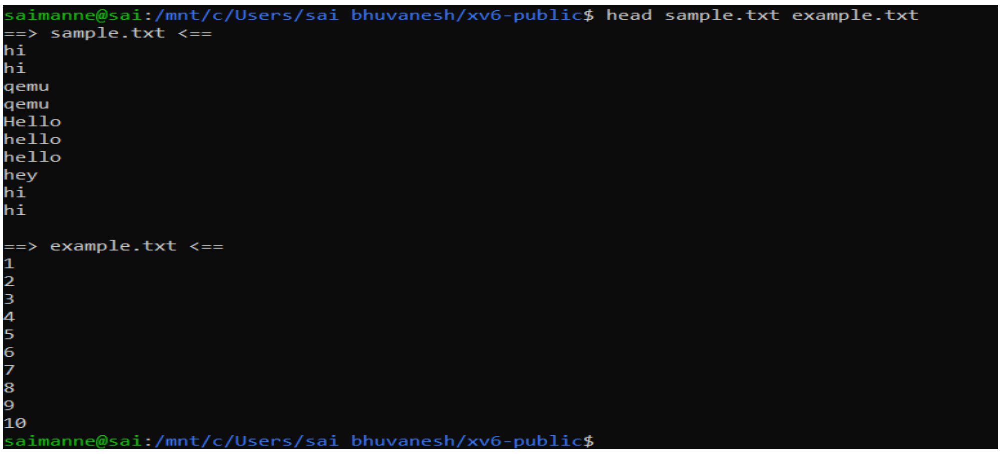
Head -n “number of lines to be read” “file 1” “file2”



```
$ head -n 3 sample.txt example.txt
head command is running in user mode
sample.txt
hi
hi
qemu
example.txt
1
2
3
$
```

9.we can read default 14 lines from two files as follows

Head “file1” “file2”



```
saimanne@sai:/mnt/c/Users/sai bhuvanesh/xv6-public$ head sample.txt example.txt
==> sample.txt <==
hi
hi
qemu
qemu
Hello
hello
hello
hey
hi
hi

==> example.txt <==
1
2
3
4
5
6
7
8
9
10
saimanne@sai:/mnt/c/Users/sai bhuvanesh/xv6-public$
```