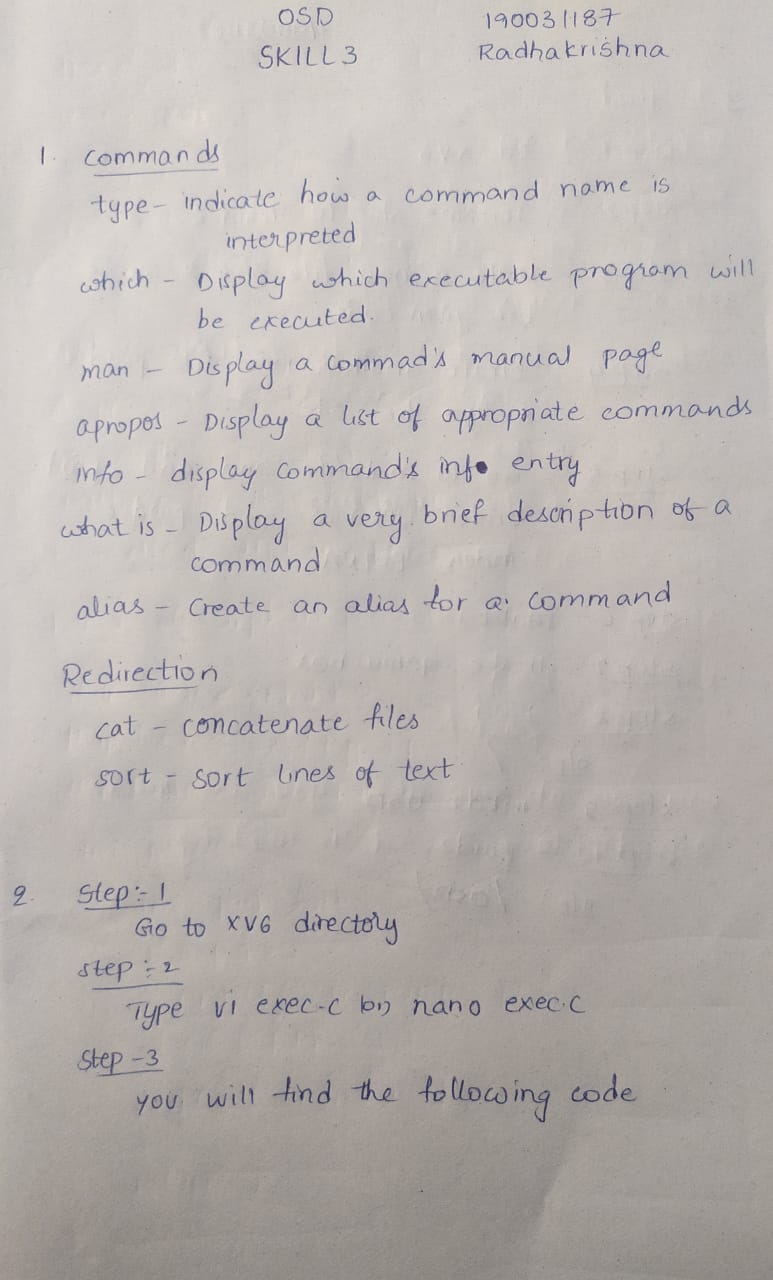
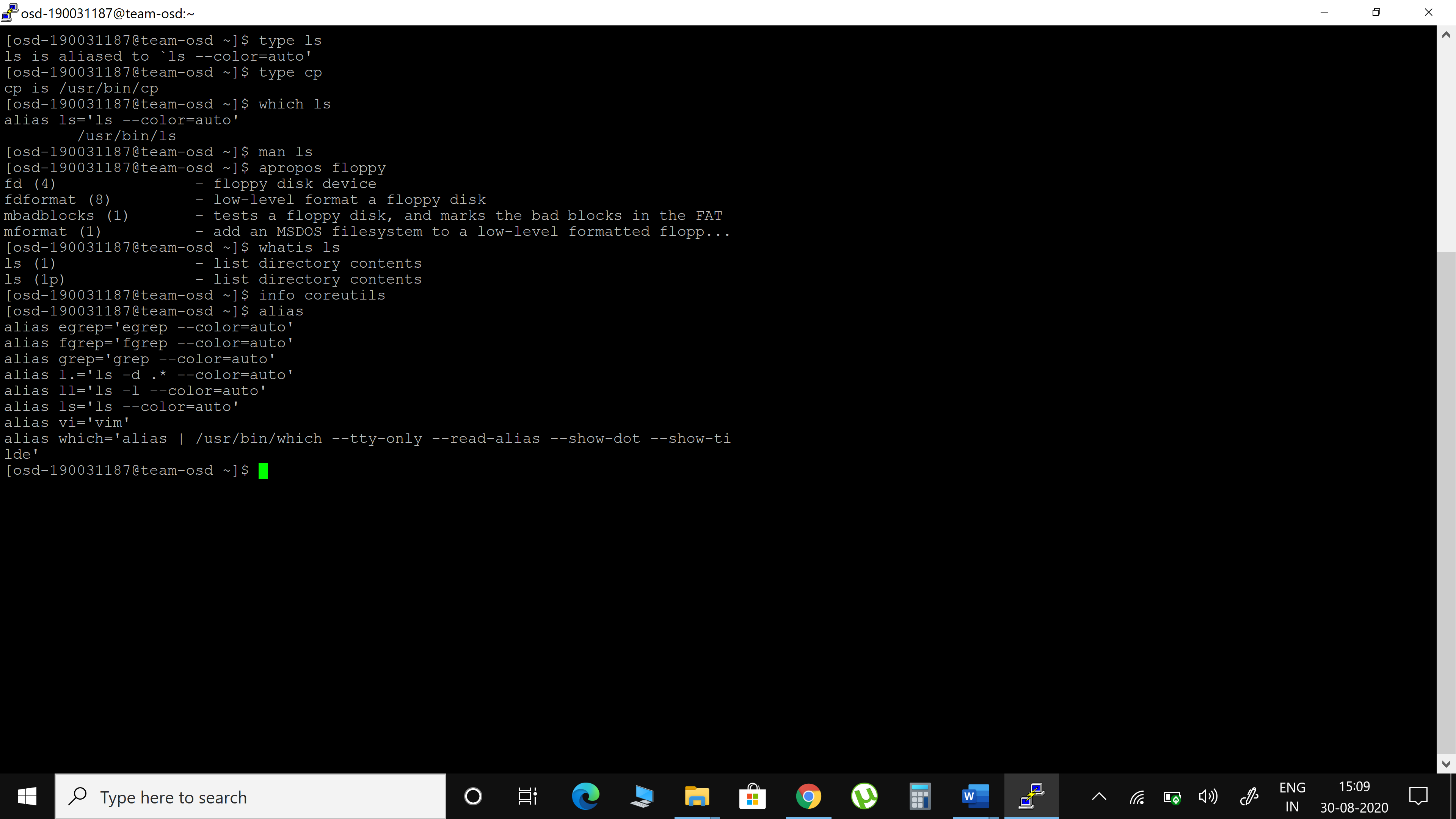
Operating System Design – 19CS2106S

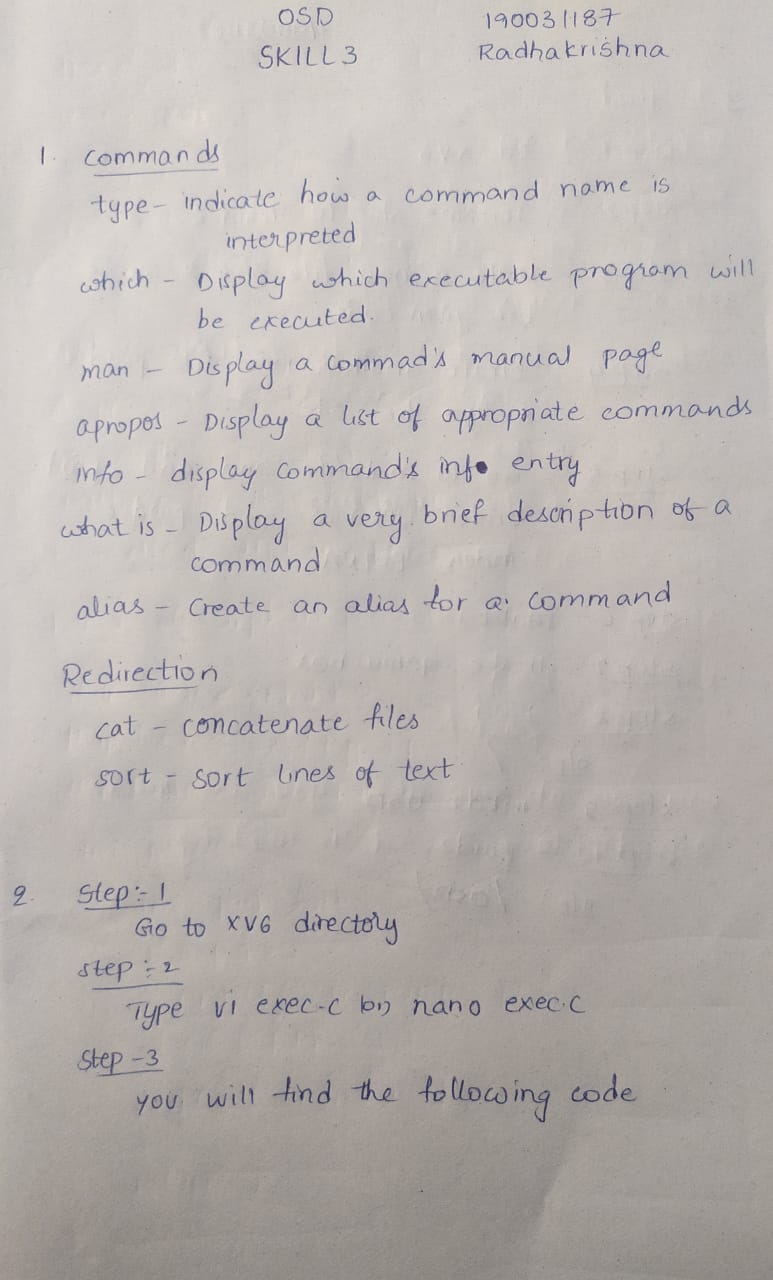
Skill Experiment – 3

1. **Working with Commands, Redirection**





1. **exec.c (Xv6 design & implementation)**



#include "types.h"

#include "param.h"

#include "memlayout.h"

#include "mmu.h"

#include "proc.h"

#include "defs.h"

#include "x86.h"

#include "elf.h"

int exec(char \*path, char \*\*argv)

{

char \*s, \*last; int i, off;

uint argc, sz, sp, ustack[3+MAXARG+1];

struct elfhdr elf;

struct inode \*ip;

struct proghdr ph;

pde\_t \*pgdir, \*oldpgdir;

struct proc \*curproc = myproc();

begin\_op();

if((ip = namei(path)) == 0)

{

end\_op();

cprintf("exec: fail\n");

return -1;

}

ilock(ip);

pgdir = 0;

// Check ELF header

if(readi(ip, (char\*)&elf, 0, sizeof(elf)) != sizeof(elf))

goto bad;

if(elf.magic != ELF\_MAGIC)

goto bad;

if((pgdir = setupkvm()) == 0)

goto bad;

// Load program into memory.

sz = 0;

for(i=0, off=elf.phoff; i< ph.filesz)

goto bad;

sz = 0;

for(i=0, off=elf.phoff; i< ph.filesz)

goto bad;

if(ph.vaddr + ph.memsz < ph.vaddr)

goto bad;

if((sz = allocuvm(pgdir, sz, ph.vaddr + ph.memsz)) == 0)

goto bad;

if(ph.vaddr % PGSIZE != 0)

goto bad;

if(loaduvm(pgdir, (char\*)ph.vaddr, ip, ph.off, ph.filesz) < 0)

goto bad;

iunlockput(ip);

end\_op();

ip = 0;

// Allocate two pages at the next page boundary.

// Make the first inaccessible. Use the second as the user stack.

sz = PGROUNDUP(sz);

if((sz = allocuvm(pgdir, sz, sz + 2\*PGSIZE)) == 0)

goto bad;

clearpteu(pgdir, (char\*)(sz - 2\*PGSIZE));

sp = sz;

// Push argument strings, prepare rest of stack in ustack.

for(argc = 0; argv[argc]; argc++)

{

if(argc >= MAXARG)

goto bad;

sp = (sp - (strlen(argv[argc]) + 1)) & ~3;

if(copyout(pgdir, sp, argv[argc], strlen(argv[argc]) + 1) < 0)

goto bad;

ustack[3+argc] = sp;

}

ustack[3+argc] = 0;

ustack[0] = 0xffffffff;

// fake return PC ustack[1] = argc;

ustack[2] = sp - (argc+1)\*4;

// argv pointer sp -= (3+argc+1) \* 4;

if(copyout(pgdir, sp, ustack, (3+argc+1)\*4) < 0)

goto bad;

// Save program name for debugging.

for(last=s=path; \*s; s++)

if(\*s == '/')

last = s+1;

safestrcpy(curproc->name, last, sizeof(curproc->name));

// Commit to the user image.

oldpgdir = curproc->pgdir; curproc->pgdir = pgdir;

curproc->sz = sz;

curproc->tf->eip = elf.entry;

// main

curproc->tf->esp = sp; switchuvm(curproc);

freevm(oldpgdir);

return 0;

bad:

if(pgdir) freevm(pgdir);

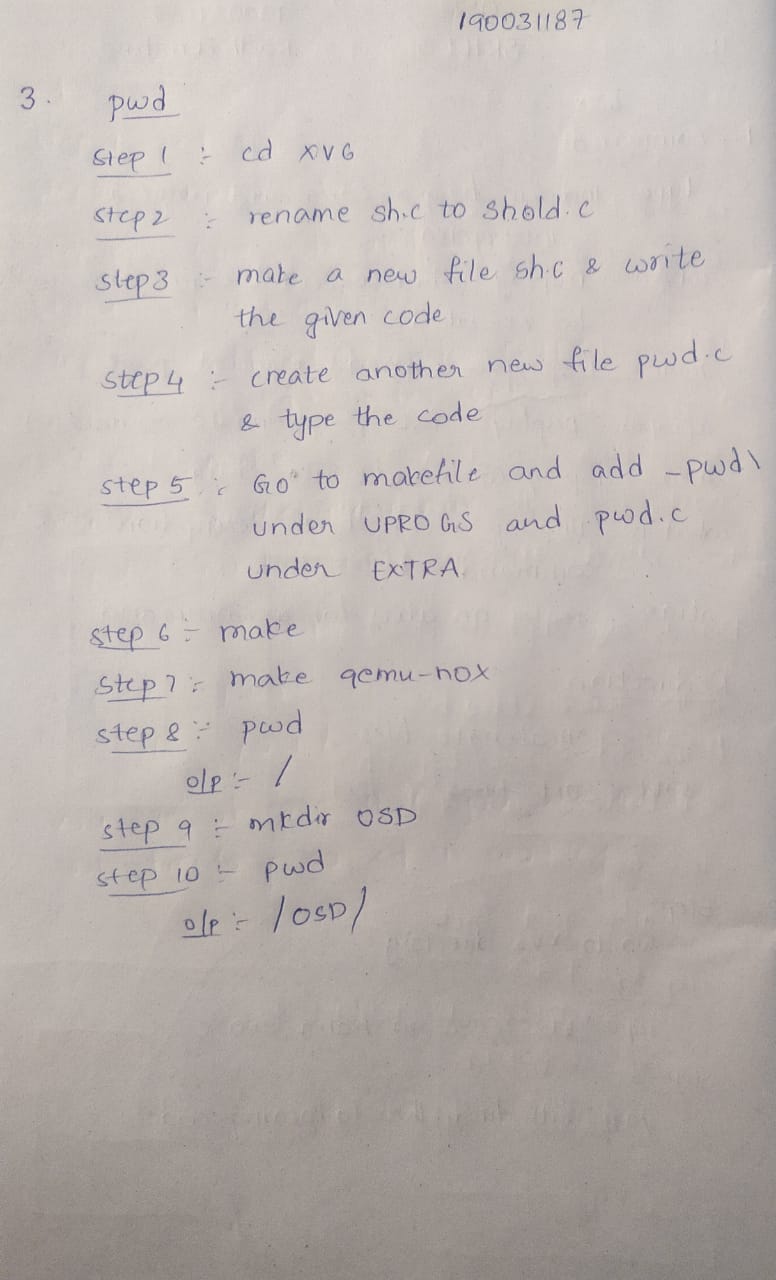
if(ip){ iunlockput(ip);

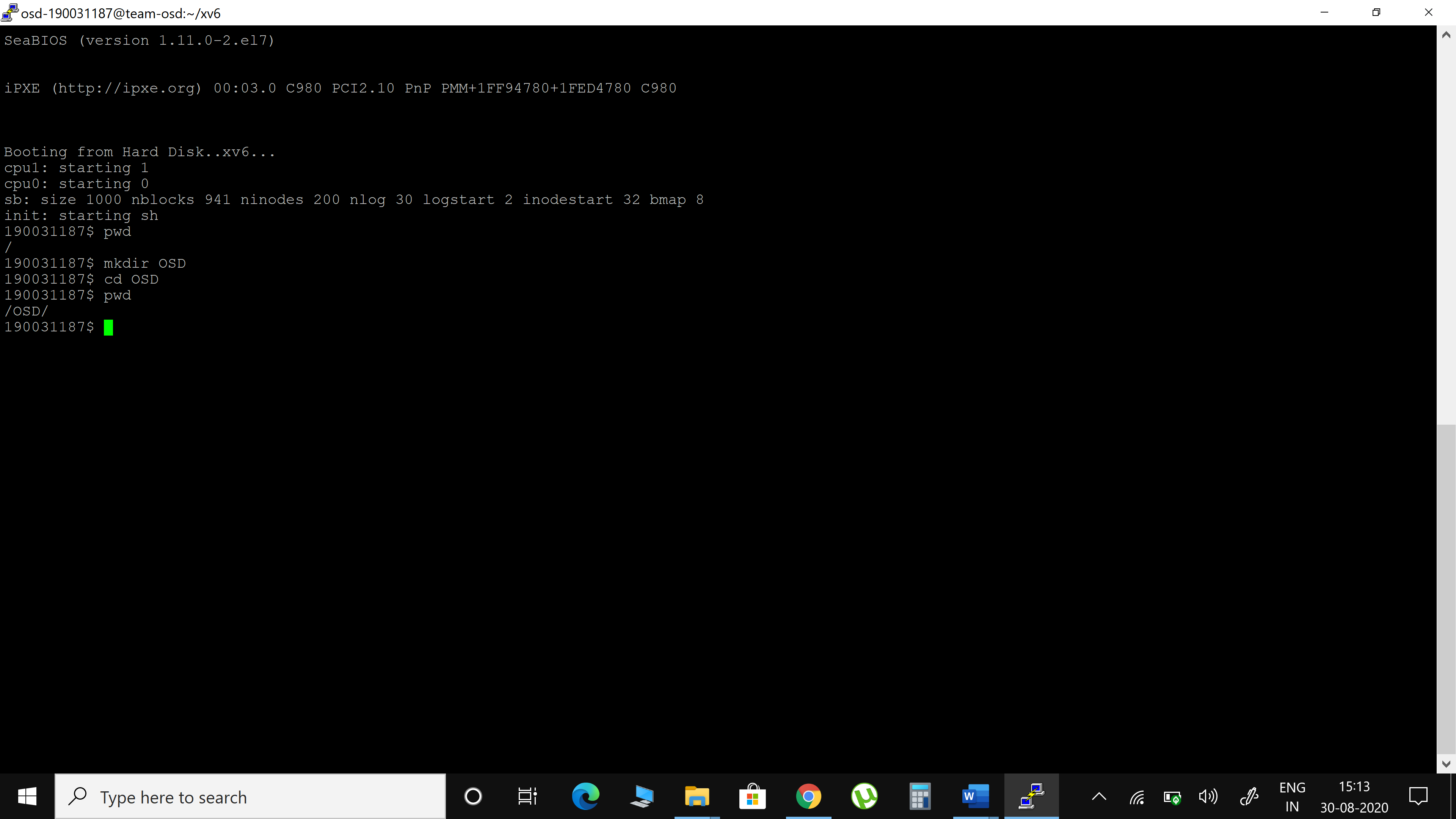
end\_op();

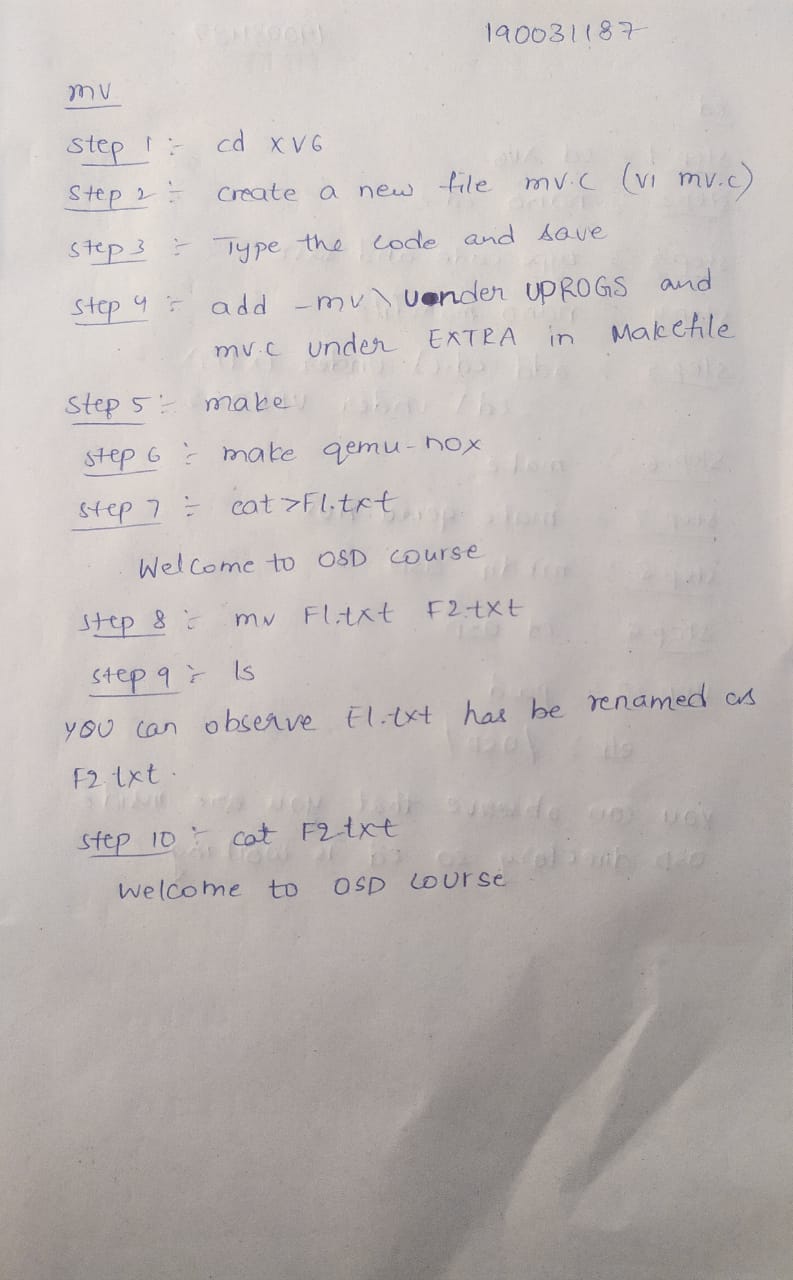
return -1;

}

1. **pwd, cd, mv(xv6 customization)**

****



****

