National Institute of Technology

Karnataka

OPERATING SYSTEM LAB

ASSIGNMENT :-1

NAME:-SHIVAM YADAV SECTION:-S2

ROLL NO:-211CS257 Code:-CS257

TYPES OF SYSTEM CALLS:

1-PROCESS CONTROL

2-FILE MANAGEMENT

3-DEVICE MANAGEMENT

4-INFORMATION MANAGEMENT

5-COMMUNICATION

1-PROCESS CONTROL SYSTEM CALLS:

1.fork()

2.exec()

3.exit()

4.wait()

5.getpid()

fork():To create a new process we use this system call.Actually it generates it's own copy(child process).

Since this function returns an integer value say 'n'

If n < 0 :- implies a new child process creation was unsuccessful.

n==0 :- means child process created successfully.

n > 0 :- means it is returned to parent/caller.

exec():-when user want to execute/process a new program/file while some other process is going on,then by calling this system call we can replace current process by our desired file/program for its execution.

exit() :-a computer process terminates itself by calling this function.It closes all other files and clear buffer ,temporary files.It takes int type argument range from 0-255.

wait():when main process is suspended due to ongoing another process untill it's done. At this instant

System invoke wait() call.

getpid():In order to get process id of ongoing process ,we use this system call.

2- FILE STRUCTRE REALATED SYSTEM CALLS:

1.open()

2.read()

3.write()

4.close()

open():- int open(const char * pathname,int flags,mode_t mode);

It creates a new file description in system wide table that has records related to all opened files in machine.

As argument it requires:

-path to a file using pointers

-flag(mode in which we want to operate)

i.e :- O_RDONLY,O_WRONLY,O_RDWR etc.

read():- ssize_t read(int fd,void *buf,size_t count);

It read from file descriptor(fd) upto mentioned size in argument into a buffer(buf). On successful reading it returns no . of bytes readed else 0 or <0.

write():- ssize_t write(int fd,const void *buf,size_t count);

It reads data from given buffer and write into mentioned file descriptor.On successful execution it return count of total character/bytes written else 0 for nothing written and <0 for errno updated appropriately.

close():- int close(int fd);

As its protype depicts that it take file descriptor information to close file reffered by fd.On success it return 0 else <0 for setting errno.

Difference Between Linux and Windows OS

criteria	Linux	Window
Access	Open source	Owned by Microsoft
filename	Case sensitive	Case insensitive
kernel	monolithic	micro
Efficiency	More efficient	Less efficient
security	More security because of: -good memory management -user space and kernel space are well seprated -root access is not given to user	Less security as compaired to linux due to earlier reasons.
For Hacking	A good operating system	It generally not deal with such area

Account	3-accounts - Root - Regular -Service Account	4-Accounts -Administrator -Standard -Child -Guest
Slash in command	Forward slash used	Backward slash used
Super user	Root user -they all previleges	Administrator -they have all previleges
CPU Sheduling algorithm	CFS (completely fair scheduling)	Multilevel feedback queue