UNIX - Lab - Test-2

1BM18CS024
Dhanaraj.K
31/12/20

2)

(a) C Program to show usage of Fork operation and wexit status macro.

```
=> #include <unistd.h>
#include <stdio.h>
#include <sys/wait.h>
#include <stdlib.h>
#include <fcntl.h>

int main ( int argc, char * argv[ ] ).
{
    int fd, exitStatus;
    int exitval = 10;
    fd = open ( argv[1], O_WRONLY | O_CREAT |
O_TRUNC, 0664);
    write (fd," Original Process writes \n");
    Switch (fork() ){
      case 0 : write ( fd;"child Process wrote \n",21)
               close(fd);
               Printf(" child : Terminating with
exit value %d \n", exitval);
               exit (exitvalue);
               break;
      default : wait (& exit status);
               Printf(" Parent : child terminated
exit value %d \n", WEXITSTATUS ( exitstatus ));
               write (fd," Parent Process writes \n", 22
               exit (20);
    }
```

1BM18CS023
Dhanraj. K

2) (b) Shell program to check if permission of two files are same.

=>

```
#! bin/shell
if [$# -ne 2]
then.
echo " Please enter 2 arguments"
elif [ !-e $1 -o !-e $2]
then.
echo " File does not exist".
else.
Per1 = `ls -l $1 | cut -c 2-10`
Per2 = `ls -l $2 | cut -c 2-10`
if [$Per1 = $Per2]
then
    echo " Permissions are identical "
    echo " Permision is $Per1".
else.
echo " Permissions are not identical".
echo " Permission of $1 is $Per1".
echo " Permission of $2 is $Per2"
fi
fi
```