OSL WEEK 4

Q1)

#include <sys/types.h>

#include <sys/stat.h>

#include <unistd.h>

#include <stdio.h>

#include <stdlib.h>

void help() { printf("USAGE: inode [name\_of\_file]\n");

}

int main(int argc, char \* argv[]) { if ( argc != 2 ) { help(); exit(EXIT\_FAILURE);

}

struct stat curr\_stat;

int r\_val = lstat( argv[1], &curr\_stat ); if ( r\_val < 0 ) { perror("lstat"); exit(EXIT\_FAILURE);

}

printf("Inode number: %ld\n", curr\_stat.st\_ino);

}



Q2)

#include <sys/types.h>

#include <sys/stat.h>

#include <unistd.h>

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#include <pwd.h> #include <grp.h>

void help() { printf("USAGE: stat [name\_of\_file]\n");

}

void pp(int r\_val, char \* str) { if ( r\_val == 0 ) { printf("-");

} else { printf("%s", str);

}

}

void print\_filetype(mode\_t m) { if ( S\_ISREG(m) ) { printf("Regular file");

} else if ( S\_ISDIR(m) ) { printf("Directory");

} else if ( S\_ISCHR(m) ) { printf("Character device");

} else if ( S\_ISBLK(m) ) { printf("Block device");

} else if ( S\_ISFIFO(m) ) { printf("Named pipe");

} else if ( S\_ISLNK(m) ) { printf("Symbolic link");

} else if ( S\_ISSOCK(m) ) { printf("Socket");

}

}

int main(int argc, char \* argv[]) { if ( argc != 2 ) { help(); exit(EXIT\_FAILURE);

}

struct stat curr\_stat; struct passwd \* file\_usr = NULL; struct group \* file\_grp = NULL; int r\_val = lstat( argv[1], &curr\_stat );

if ( r\_val < 0 ) { perror("lstat"); exit(EXIT\_FAILURE);

}

mode\_t m = curr\_stat.st\_mode;

printf("Inode number: %ld\n", curr\_stat.st\_ino); printf("Device id: %ld\n", curr\_stat.st\_dev); printf("Mode: %d\n", m); printf("File permissions: "); pp( m & S\_IRUSR, "r" ); pp( m & S\_IWUSR, "w" ); pp( m & S\_IXUSR, "x" ); pp( m & S\_IRGRP, "r" ); pp( m & S\_IWGRP, "w" ); pp( m & S\_IXGRP, "x" ); pp( m & S\_IROTH, "r" ); pp( m & S\_IWOTH, "w" ); pp( m & S\_IXOTH, "x" ); printf("\n"); printf("File type: "); print\_filetype(m); printf("\n");

file\_usr = getpwuid(curr\_stat.st\_uid);

if ( file\_usr == NULL ) { perror("getpwuid"); exit(EXIT\_FAILURE);

}

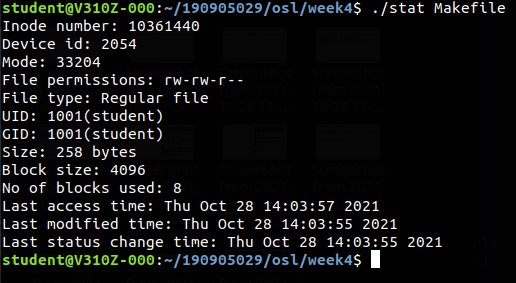
file\_grp = getgrgid(curr\_stat.st\_gid); if ( file\_grp == NULL ) { perror("getgrgid"); exit(EXIT\_FAILURE);

}

printf("UID: %d(%s)\n", curr\_stat.st\_uid, file\_usr->pw\_name); printf("GID: %d(%s)\n", curr\_stat.st\_gid, file\_grp->gr\_name); printf("Size: %ld bytes\n", curr\_stat.st\_size); printf("Block size: %ld\n", curr\_stat.st\_blksize); printf("No of blocks used: %ld\n", curr\_stat.st\_blocks); printf("Last access time: %s", ctime(&curr\_stat.st\_atime)); printf("Last modified time: %s", ctime(&curr\_stat.st\_mtime));

printf("Last status change time: %s", ctime(&curr\_stat.st\_ctime)); free(file\_usr); free(file\_grp);

}



Q3)

#include <stdio.h>

#include <stdlib.h>

#include <limits.h> #include <unistd.h>

void help() { printf("USAGE: hlink [name\_of\_file]

[name\_of\_destination\_file]\n");

}

int main(int argc, char \* argv[]) { if ( argc != 3 ) { help();

exit(EXIT\_FAILURE);

}

int r\_val = link(argv[1], argv[2]);

if ( r\_val < 0 ) { perror("link"); exit(EXIT\_FAILURE);

} char \* path = realpath(argv[2], NULL);

if ( path == NULL ) { perror("realpath"); exit(EXIT\_FAILURE);

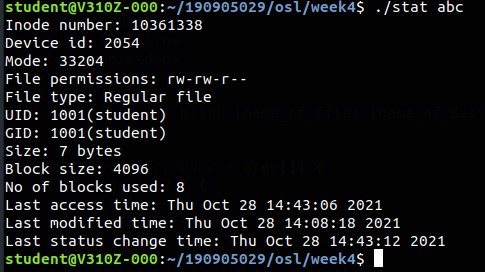
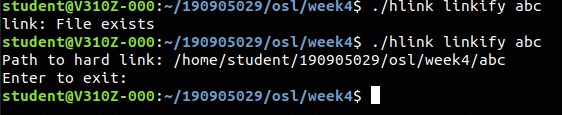
} printf("Path to hard link: %s\nEnter to exit: ", path); getc(stdin);

r\_val = unlink(argv[2]); if ( r\_val < 0 ) { perror("unlink"); exit(EXIT\_FAILURE);

}

free(path);

}



Q4)

#include <stdio.h>

#include <stdlib.h>

#include <limits.h> #include <unistd.h>

void help() { printf("USAGE: symlink [name\_of\_file]

[name\_of\_destination\_file]\n");

}

int main(int argc, char \* argv[]) { if ( argc != 3 ) { help();

exit(EXIT\_FAILURE);

}

int r\_val = symlink(argv[1], argv[2]);

if ( r\_val < 0 ) { perror("link"); exit(EXIT\_FAILURE);

} char \* path = realpath(argv[2], NULL);

if ( path == NULL ) { perror("realpath"); exit(EXIT\_FAILURE);

} printf("Path to hard link: %s\nEnter to exit: ", path); getc(stdin);

r\_val = unlink(argv[2]); if ( r\_val < 0 ) { perror("unlink"); exit(EXIT\_FAILURE);

}

free(path);

}

