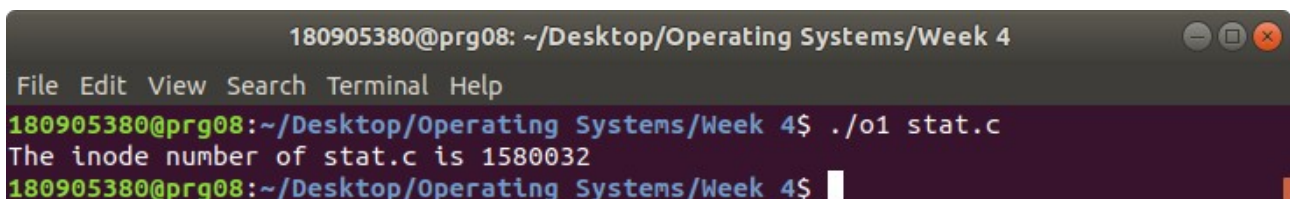


Q1. Write a program to find the inode number of an existing file in a directory. Take the input as a filename and print the inode number of the file.

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <unistd.h>
#include <sys/stat.h>

int main (int argc, char *argv[])
{
    struct stat sb;
    int ret;
    if (argc < 2)
    {
        fprintf(stderr, "usage: %s <file>\n", argv[0]);
        return 1;
    }
    ret = stat(argv[1], &sb);
    if (ret)
    {
        perror ("stat");
        return 1;
    }
    printf("The inode number of %s is %ld\n", argv[1],sb.st_ino);
    return 0;
}
```

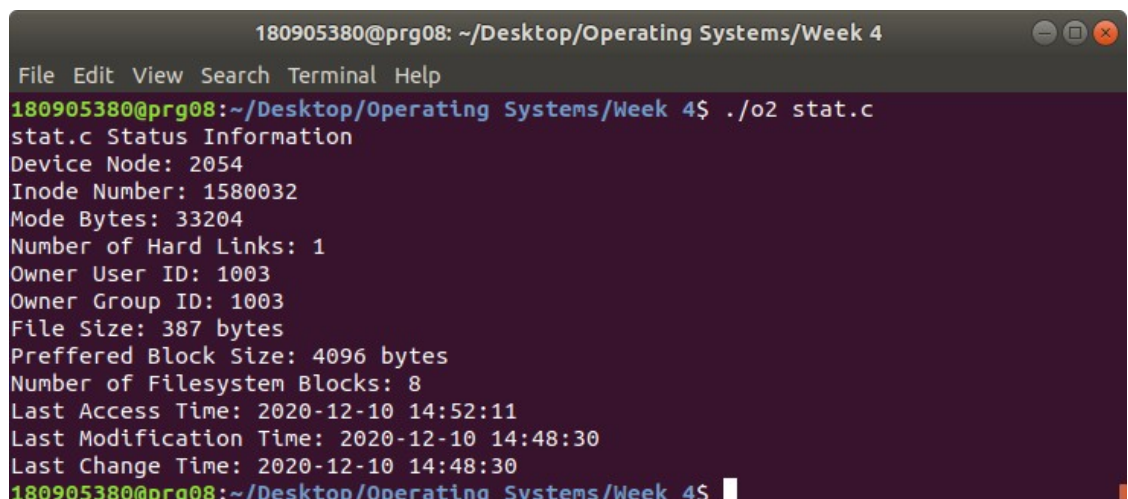
A terminal window titled "180905380@prg08: ~/Desktop/Operating Systems/Week 4" with standard window controls. The terminal shows the command `./o1 stat.c` being executed, which outputs "The inode number of stat.c is 1580032". The prompt then returns to `180905380@prg08:~/Desktop/Operating Systems/Week 4$`.

```
180905380@prg08: ~/Desktop/Operating Systems/Week 4
File Edit View Search Terminal Help
180905380@prg08:~/Desktop/Operating Systems/Week 4$ ./o1 stat.c
The inode number of stat.c is 1580032
180905380@prg08:~/Desktop/Operating Systems/Week 4$
```

Q2. Write a program to print out the complete stat structure of a file.

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <unistd.h>
#include <sys/stat.h>
#include <time.h>

int main (int argc, char *argv[])
{
    struct stat sb;
    int ret;
    if (argc < 2)
    {
        fprintf(stderr, "usage: %s <file>\n", argv[0]);
        return 1;
    }
    ret = stat(argv[1], &sb);
    if (ret)
    {
        perror ("stat");
        return 1;
    }
    printf("%s Status Information\n", argv[1]);
    printf("Device Node: %ld\n",sb.st_dev);
    printf("Inode Number: %ld\n",sb.st_ino);
    printf("Mode Bytes: %d\n",sb.st_mode);
    printf("Number of Hard Links: %ld\n",sb.st_nlink);
    printf("Owner User ID: %d\n",sb.st_uid);
    printf("Owner Group ID: %d\n",sb.st_gid);
    printf("File Size: %ld bytes\n",sb.st_size);
    printf("Preffered Block Size: %ld bytes\n",sb.st_blksize);
    printf("Number of Filesystem Blocks: %ld\n",sb.st_blocks);
    char* buff;
    strftime(buff, 20, "%Y-%m-%d %H:%M:%S", localtime(&sb.st_atime));
    printf("Last Access Time: %s\n",buff);
    strftime(buff, 20, "%Y-%m-%d %H:%M:%S", localtime(&sb.st_mtime));
    printf("Last Modification Time: %s\n",buff);
    strftime(buff, 20, "%Y-%m-%d %H:%M:%S", localtime(&sb.st_mtime));
    printf("Last Change Time: %s\n",buff);
    return 0;
}
```

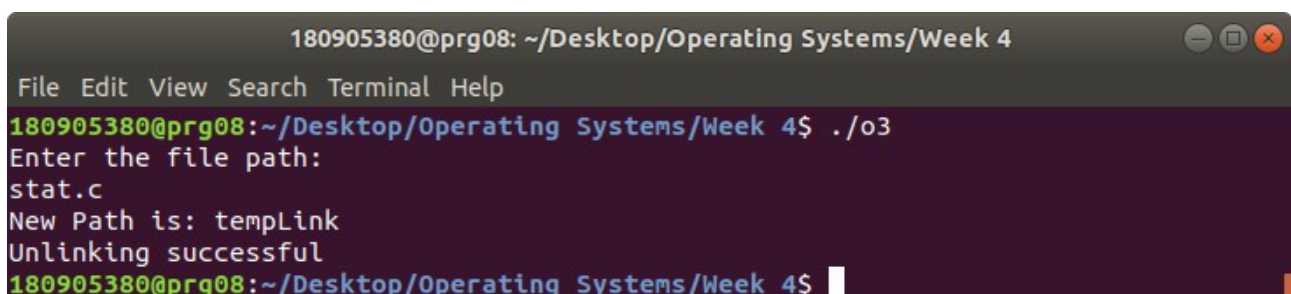


The screenshot shows a terminal window titled "180905380@prg08: ~/Desktop/Operating Systems/Week 4". The terminal displays the output of the command `./o2 stat.c`, which is the execution of the `stat.c` program. The output shows the status information for a file, including device node, inode number, mode bytes, number of hard links, owner user and group IDs, file size, preferred block size, number of filesystem blocks, and last access, modification, and change times.

```
180905380@prg08: ~/Desktop/Operating Systems/Week 4
File Edit View Search Terminal Help
180905380@prg08:~/Desktop/Operating Systems/Week 4$ ./o2 stat.c
stat.c Status Information
Device Node: 2054
Inode Number: 1580032
Mode Bytes: 33204
Number of Hard Links: 1
Owner User ID: 1003
Owner Group ID: 1003
File Size: 387 bytes
Preffered Block Size: 4096 bytes
Number of Filesystem Blocks: 8
Last Access Time: 2020-12-10 14:52:11
Last Modification Time: 2020-12-10 14:48:30
Last Change Time: 2020-12-10 14:48:30
180905380@prg08:~/Desktop/Operating Systems/Week 4$
```

Q3. Write a program to create a new hard link to an existing file and unlink the same. Accept the old path as input and print the newpath.

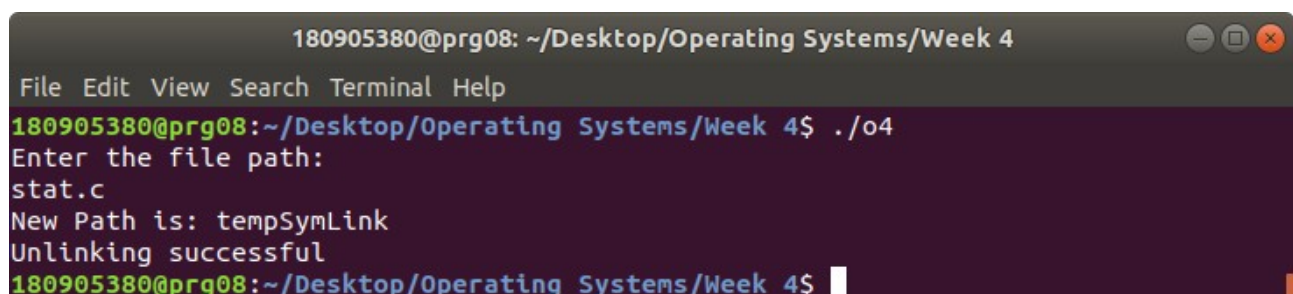
```
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<sys/types.h>
#include<wait.h>
#include<sys/stat.h>
#include<errno.h>
int main()
{
    char fPath[100],newPath[100]="tempLink";
    printf("Enter the file path:\n");
    scanf("%s",fPath);
    int status = link(fPath,newPath);
    if(status== -1)
    {
        printf("Error occured while linking\n");
        printf("Errno: %d\n",errno);
        exit(1);
    }
    printf("New Path is: %s\n",newPath);
    status = unlink(newPath);
    if(status == -1)
    {
        printf("Error occured while unlinking\n");
        printf("Errno: %d\n",errno);
        exit(1);
    }
    printf("Unlinking successful\n");
}
```

A terminal window titled "180905380@prg08: ~/Desktop/Operating Systems/Week 4" with a menu bar (File, Edit, View, Search, Terminal, Help). The prompt is "180905380@prg08:~/Desktop/Operating Systems/Week 4\$". The user enters "./o3". The program outputs "Enter the file path:", "stat.c", "New Path is: tempLink", and "Unlinking successful". The prompt returns to "180905380@prg08:~/Desktop/Operating Systems/Week 4\$".

```
180905380@prg08: ~/Desktop/Operating Systems/Week 4
File Edit View Search Terminal Help
180905380@prg08:~/Desktop/Operating Systems/Week 4$ ./o3
Enter the file path:
stat.c
New Path is: tempLink
Unlinking successful
180905380@prg08:~/Desktop/Operating Systems/Week 4$
```

Q4. Write a program to create a new soft link to an existing file and unlink the same. Accept the old path as input and print the newpath.

```
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<sys/types.h>
#include<wait.h>
#include<sys/stat.h>
#include<errno.h>
int main()
{
    char fPath[100],newPath[100]="tempSymLink";
    printf("Enter the file path:\n");
    scanf("%s",fPath);
    int status = symlink(fPath,newPath);
    if(status== -1)
    {
        printf("Error occured while linking\n");
        printf("Errno: %d\n",errno);
        exit(1);
    }
    printf("New Path is: %s\n",newPath);
    status = unlink(newPath);
    if(status == -1)
    {
        printf("Error occured while unlinking\n");
        printf("Errno: %d\n",errno);
        exit(1);
    }
    printf("Unlinking successful\n");
}
```

A terminal window titled "180905380@prg08: ~/Desktop/Operating Systems/Week 4" with a menu bar (File, Edit, View, Search, Terminal, Help). The prompt is "180905380@prg08:~/Desktop/Operating Systems/Week 4\$". The user enters "./o4", and the program outputs "Enter the file path:", "stat.c", "New Path is: tempSymLink", and "Unlinking successful". The prompt returns to "180905380@prg08:~/Desktop/Operating Systems/Week 4\$".

```
180905380@prg08: ~/Desktop/Operating Systems/Week 4
File Edit View Search Terminal Help
180905380@prg08:~/Desktop/Operating Systems/Week 4$ ./o4
Enter the file path:
stat.c
New Path is: tempSymLink
Unlinking successful
180905380@prg08:~/Desktop/Operating Systems/Week 4$
```