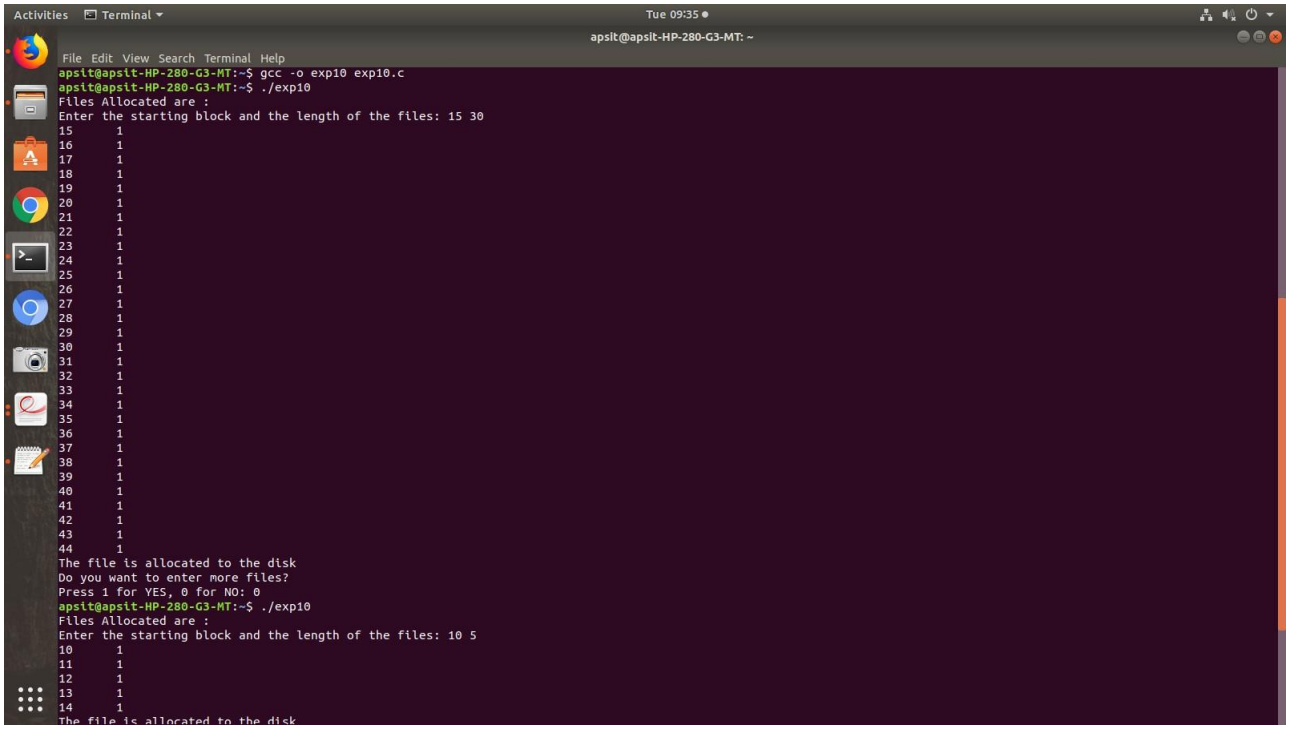
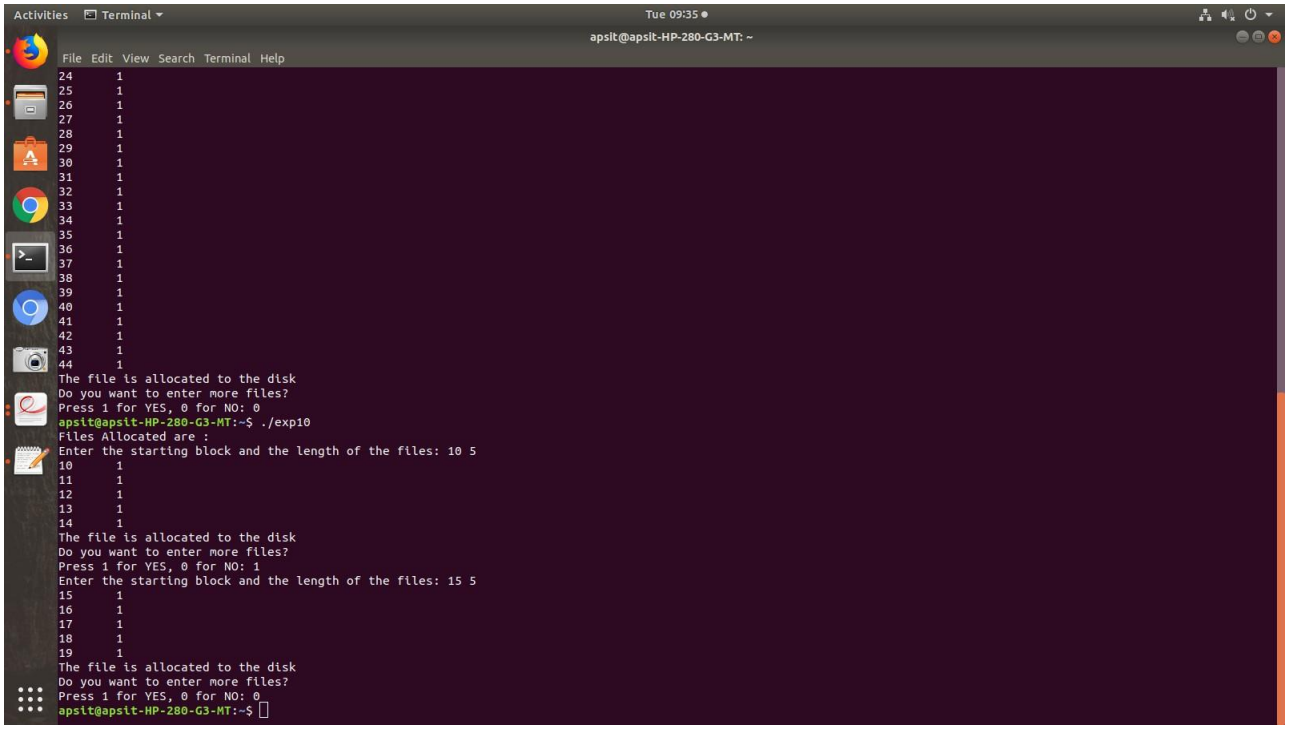
NAME:- Komal Chitnis MOODLEID:-20102068
DIV:-A ROLLNO:-26

OSEXPERIMENT-11

```
#include
<stdio.h>#include
<conio.h>#include<stdl
ib.h>voidrecurse(intfile
s[]){
intflag=0,startBlock, len,j, k,ch;
printf("Enter the starting block and the length of the files:
");scanf("%d%d",&startBlock, &len);
for (j=startBlock; j<(startBlock+len);
j++){if(files[j] ==0)
flag++;
}
if(len==flag){
for (int k=startBlock; k<(startBlock+len);
k++){if(files[k] == 0){
files[k]=1;printf("%d\t%d\n",k
,files[k]);
}
}
if(k!=(startBlock+len-1))
printf("Thefileisallocatedtothedisk\n");
}
else
printf("The file is not allocated to the
disk\n");printf("Do you want to enter more
files?\n");printf("Press 1 for YES, 0 for NO:
");scanf("%d",&ch);
if (ch ==
1)recurse(file
s);else
exit(0);
return;
}
intmain()
{
intfiles[50];
for(int
i=0;i<50;i++)files[i]
=0;
printf("Files Allocated are
:\n");recurse(files);
getch();ret
urn0;
}
```

OUTPUT:-





PROGRAM:-

```
#include
<stdio.h>#include
<conio.h>#include
<stdlib.h>
int files[50], indexBlock[50], indBlock,
n;voidrecurse1();
void
recurse2();voidr
ecurse1(){
printf("Enter the index block:
");scanf("%d",&indBlock);
if(files[indBlock]!=1){
printf("Enterthe numberofblocksandthe numberoffiles
```

```
needed for the index %d on the disk: ",
indBlock);scanf("%d",&n);
}
else{
printf("%d is already allocated\n",
indBlock);recurse1();
}
recurse2();
}
void
recurse2(){intch
;
intflag = 0;
for (int i=0; i<n;
i++){scanf("%d",
&indexBlock[i]);if
(files[indexBlock[i]] ==
0)flag++;
}
if(flag ==n){
for (int j=0; j<n;
j++){files[indexBlock[j]
]=1;
}
printf("Allocated\n");pri
ntf("File
Indexed\n");for(intk=0;
k<n;k++){
printf("%d ------>%d:%d\n",indBlock, indexBlock[k],
files[indexBlock[k]]);
}
}
else{
printf("File in the index is already
allocated\n");printf("Enter another indexed
file\n");recurse2();
}
printf("Do you want to enter more
files?\n");printf("Enter 1 for Yes, Enter 0 for
No: ");scanf("%d",&ch);
if (ch ==
1)recurse1(
);elseexit(0
);return;
}
intmain()
{
for(int
i=0;i<50;i++)files[i]
=0;recurse1();
return0;
}OUTPUT

:-
```