5)WRITE A FUNCTION FORK(INT fd,CHAR * BUFF,INT B_SIZE ,INT N,INT SKIP) THAT READS TO BUFF FROM FILE DESCRIPTOR N BLOCKS OF SIZE B_SIZE EACH. THE LAST ARHUMENT SPECIFIES ,HOW MANY BYTES TO SKI AFTER READING EACH BLOCK RETURN -1. THE OPERATON IS UNSUCESSFULL RETURN THE TOTAL NO OF BYTES READ.

```
#include<stdio.h>
#include<sys/types.h>
#include<sys/wait.h>
#include<unistd.h>
#include<stdlib.h>
#include<fcntl.h>
int foo(int fd,char * buff,int b_size,int n,int skip)

{

int sum=0;

for(int i=0;i<b_size;i++)

{

sum+=read(fd,buff,b_size);

lseek(fd,skip,SEEK_CUR);

}

return sum;

}

int main()

{

int fd,b_size,skip,n;

fd=open("text.txt",O_RDONLY);

if(fd>0)

{

perror("r1");

exit(1);
```

```
}
printf("enter no of blocks ");
scanf("%d",&n);
printf("enter size of blocks");
scanf("%d",&b_size);
printf("emter number of char to be skipped");
scanf("%d",&skip);
char* buff=(char *)calloc(b_size,sizeof(char));
if(foo(fd,buff,b_size,n,skip)==-1)
printf("!!!!!!!!!!UNSUCESSFULL!!!!!!!");
else
printf("%d\n ",foo(fd,buff,b_size,n,skip));
return 0;
}
```