1)

```
#include<stdio.h>
#include<unistd.h>
#include<sys/types.h>
void main(){
        int fd;
        mode_t mode=S_IROTH;
        char *filename=”tmp/file”;
        fd=creat(filename,mode);
        data[100]=”Hritik Kothari\n 8677\n SEIT”;
        write(fd,data,strlen(data));
        close(fd);
}
```

2)

```
#include<stdio.h>
#include<unistd.h>
#include<sys/types.h>
void main(){
        char *filename=”tmp/file”;
        mode_t mode=S_IROTH;
        int fd=open(path,mode);
        char data[100];
        read(fd,data,100);
        printf(data);
        close(fd);
}
```

3)

a. dup():

The dup() system call creates a copy of a file descriptor.

- It uses the lowest-numbered unused descriptor for the new descriptor.

- If the copy is successfully created, then the original and copy file descriptors may be used interchangeably.
- They both refer to the same open file description and thus share file offset and file status flags.

b. isseek():

From a given file (e.g. input.txt) read the alternate nth byte and write it on another file with the help of “lseek”.

lseek (C System Call): lseek is a system call that is used to change the location of the read/write pointer of a file descriptor. The location can be set either in absolute or relative terms.