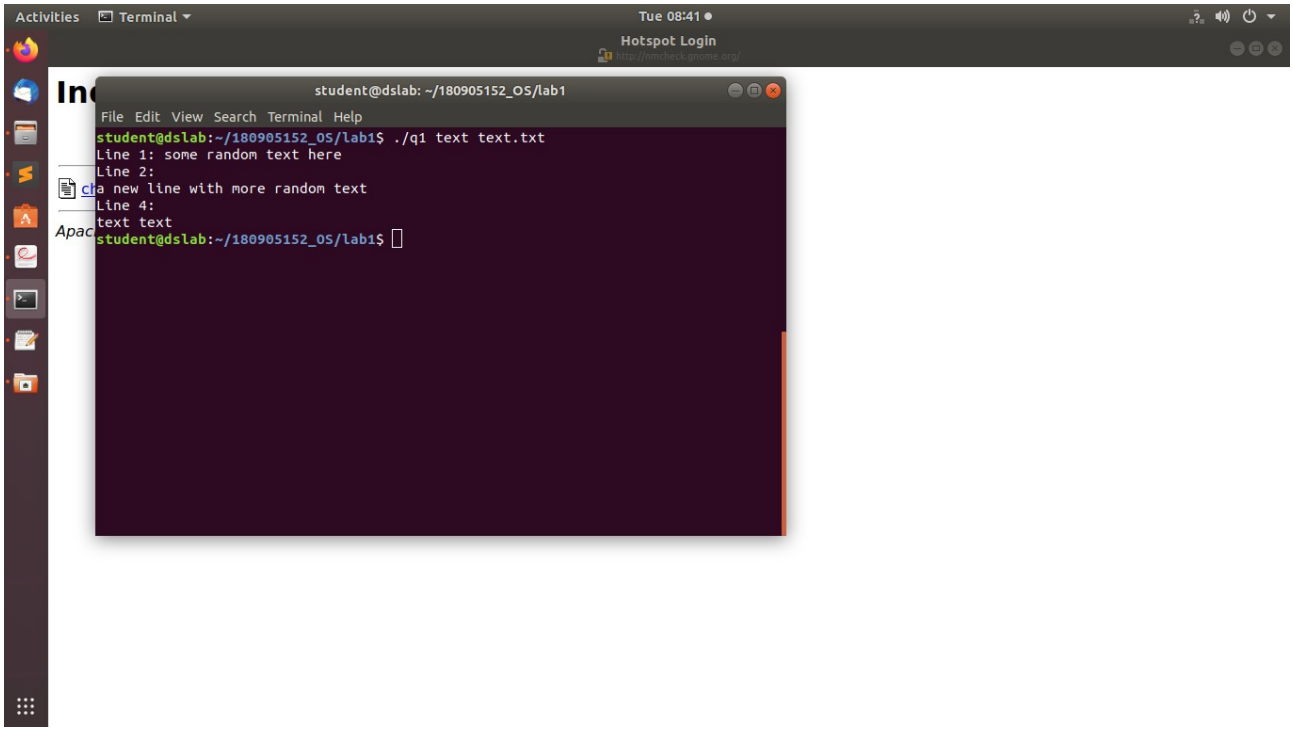
Lab 1

Q1



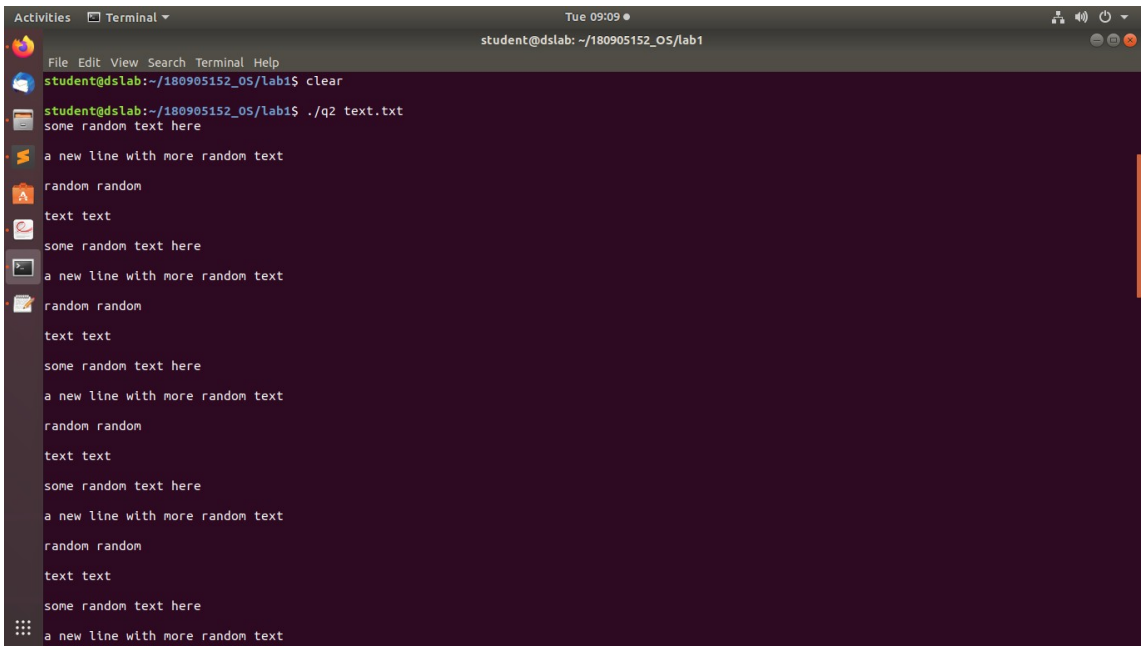
Code:

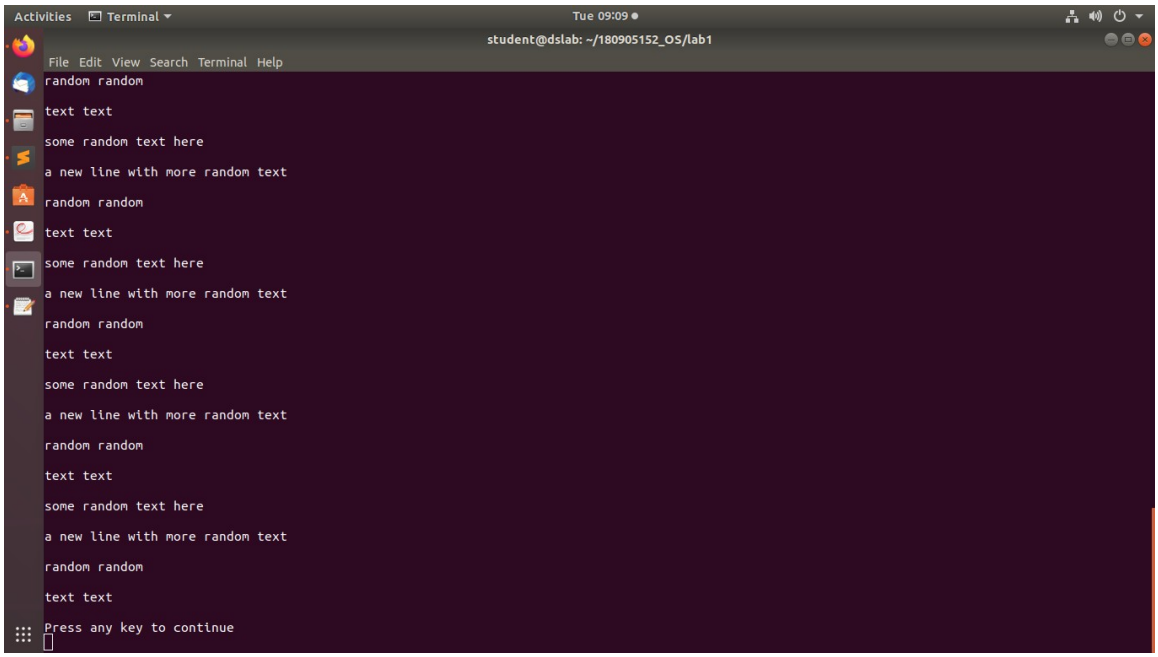
```
#include<unistd.h>
#include<sys/stat.h>
#include<fcntl.h>
#include<stdlib.h>
#include<stdio.h>
#include<string.h>

int main(int argc, char *argv[])
{
	char buf[500];
	char c;
	int i=0,in;
	if(argc<3){
		printf("Error!");
		exit(0);
	}
	in = open(argv[2],O_RDONLY);
	if(in==-1){
		printf("\nFile Not found!");
		exit(0);
	}
	int linecounter = 0;
	while(read(in,&c,sizeof(char))>0)
	{
```

```
                if(c=='\n')
                {
                        i=0;
                        linecounter++;
                        if(strstr(buf,argv[1])!=NULL)
                        {
                                printf("Line %d: ",linecounter);
                                printf("%s \n",buf);
                        }
                        memset(buf, 0, sizeof(buf));
                }
                buf[i] = c;
                i++;
        }
}
```

Q2



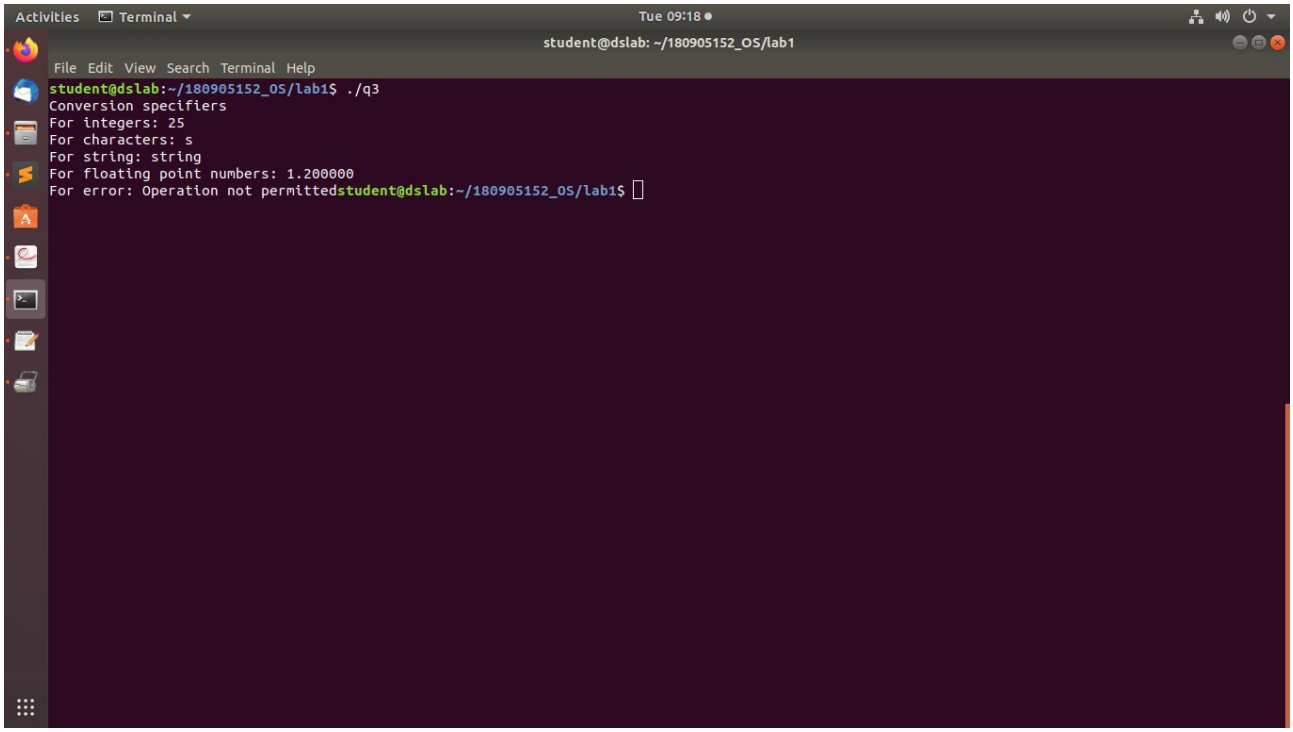


Code:

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

int main(int argc, char* argv[])
{
	char buf[500];
	char c;
	int i=0,in;
	if(argc<2){
		printf("Error!");
		exit(0);
	}
	in = open(argv[1],O_RDONLY);
	if(in==-1){
		printf("\nFile Not found!");
		exit(0);
	}
	int linecounter = 0;
	while(read(in,&c,sizeof(char))>0)
	{
		if(c=='\n')
		{
			i=0;
			linecounter++;
			printf("%s \n",buf);
			memset(buf, 0, sizeof(buf));
			if(linecounter==20){
				printf("\nPress any key to continue\n" );
				getchar();
				linecounter=0;
			}
		}
		buf[i] = c;
		i++;
	}
}
```

Q3

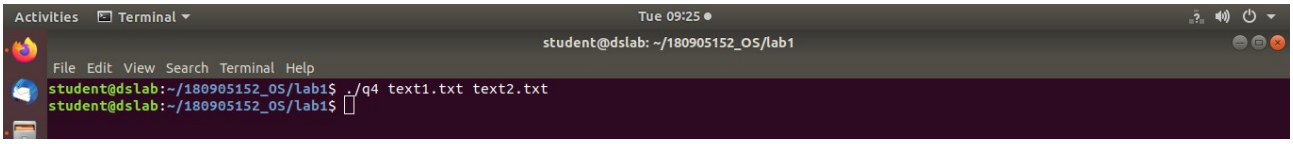


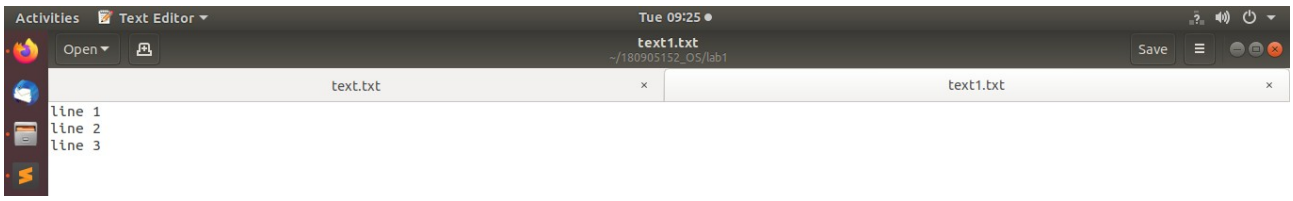
Code:

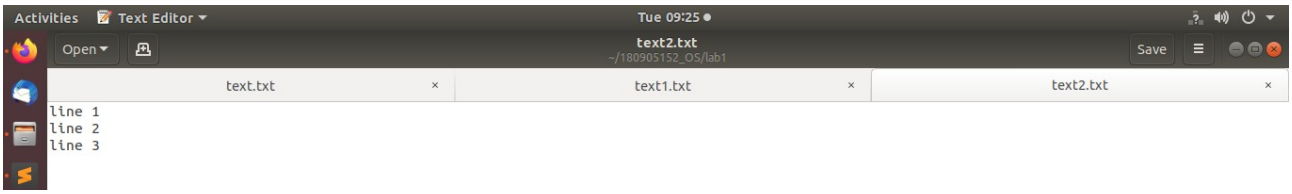
```
#include<stdio.h>
#include<errno.h>

void main()
{
	printf("Conversion specifiers\n");
	printf("For integers: %d\n",25);
	printf("For characters: %c\n",'s');
	printf("For string: %s\n","string");
	printf("For floating point numbers: %f\n",1.2);
	errno = EPERM;
	printf("For error: %m");
}
```

Q4







Code:

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

int main(int argc, char *argv[])
{
        char c;
        int in, out;
        in = open(argv[1], O_RDONLY);
        out = open(argv[2], O_WRONLY|O_CREAT, S_IRUSR | S_IWUSR);
        while(read(in,&c,1) == 1)
        write(out,&c,1);
        exit(0);
}
```