

```
1 #include"VFSheader.h"
2
3 extern UFDT UFDTArr[50];
4 extern SUPERBLOCK SUPERBLOCKobj;
5 extern PINODE head ;
6
7
8 int main()
9 {
10     char *ptr=NULL;
11     int ret=0,fd=0,count=0;
12     char command[4][80],str[80],arr[1024],tempstr[80];
13
14     InitialiseSuperBlock();
15     CreatedILB();
16
17     while (1) {
18         fflush(stdin);
19         strcpy_s(str,"");
20
21         printf("\nVirtual File System :>");
22         fgets(str,80,stdin);
23         count=sscanf(str,"%s%s%s",command[0],command[1],command[2],command[3]);
24
25         if (count == 1) {
26             if (_stricmp(command[0],"ls")==0) {
27                 ls_file();
28             }
29             else if (_stricmp(command[0],"closeall")==0) {
30                 CloseAllFile();
31                 printf("All File Closed Successfully\n" );
32                 continue;
33             }
34             else if (_stricmp(command[0],"clear")==0) {
35                 system("cls");
36                 continue;
37             }
38             else if (_stricmp(command[0],"help")==0) {
39                 DisplayHelp();
40                 continue;
41             }
42             else if (_stricmp(command[0],"exit")==0) {
43                 printf("Terminating Virtual File System Created By Ayush Agarwal\n" );
44                 break;
45             }
46             else
47             {
48                 printf("\nError:Command not found !!!\n" );
49                 continue;
50             }
51         }
52         else if (count == 2) {
53             if (_stricmp(command[0],"stat")==0) {
54                 ret = stat_file(command[1]);
55                 if (ret==-1) {
56                     printf("ERROR : Incorrect parameters\n" );
57                 }
58                 if (ret == -2)
59                 {
60                     printf("Error : There is no such file\n" );
61                 }
62                 continue;
63             }
64             else if (_stricmp(command[0],"fstat")==0) {
65                 ret = fstat_file(atoi(command[1]));
66                 if (ret==-1) {
```

```

67         printf("ERROR : Incorrect parameters\n" );
68     }
69     if (ret == -2)
70     {
71         printf("Error : There is no such file\n" );
72     }
73     continue;
74 }
75 else if (_stricmp(command[0],"close")==0) {
76     ret = CloseFileByName(command[1]);
77     if(ret == -1)
78         printf("Error : There is no such file\n" );
79     continue;
80 }
81 else if (_stricmp(command[0],"rm")==0) {
82     ret = rm_File(command[1]);
83     if(ret == -1)
84         printf("Error : There is no such file\n" );
85     continue;
86 }
87 else if (_stricmp(command[0],"man")==0) {
88     man(command[1]);
89 }
90 else if (_stricmp(command[0],"write")==0) {
91     fd =GetFDFromName(command[1]);
92     if (fd==-1) {
93         printf("Error: Incorrect parameters\n" );
94         continue;
95     }
96     printf("Enter the data \n" );
97     scanf("%s^\n",arr );
98     ret=strlen(arr);
99     if (ret == 0) {
100         printf("Error : Incorrect parameters\n" );
101         continue;
102     }
103     ret = WriteFile(fd,arr,ret);
104     if (ret == -1) {
105         printf("Error : Permission Denied \n");
106     }
107     if (ret == -2) {
108         printf("Error : There is no sufficient memory to write\n" );
109     }
110     if (ret == -3) {
111         printf("Error : It is not a regular file\n" );
112     }
113 }
114 else if (_stricmp(command[0],"truncate")==0) {
115     ret = truncate_File(command[1]);
116     if (ret == -1) {
117         printf("Error : Incorrect parameters\n" );
118     }
119 }
120 else
121 {
122     printf("\nError : Command not Found !!! \n" );\
123     continue;
124 }
125 }
126 else if (count == 3 ) {
127     if (_stricmp(command[0],"create")==0) {
128         printf("%s",command[2]);
129         ret = CreateFile(command[1],atoi(command[2]));
130         if (ret >=0)
131             printf("\nFile is successfully created with file descriptor \n" );
132         if(ret == -1)

```

```

133         printf("Error : Incorrect parameters\n" );
134     if (ret == -2) {
135         printf("Error : There is no inodes\n" );
136     }
137     if (ret == -3)
138         printf("Error : File already Exists\n" );
139     if (ret == -4)
140         printf("Error : memory llocation failure\n" );
141     continue;
142 }
143 else if (_stricmp(command[0],"open")==0) {
144     ret = OpenFile(command[1],atoi(command[2]));
145     if (ret >=0)
146         printf("File is Successfully opened with file descriptor \n" );
147     if (ret == -1)
148         printf("Error : Incorrect parameters\n" );
149     if (ret == -2)
150         printf("Error : File not present\n" );
151     if (ret == -3)
152         printf("Error : Permission Denied\n" );
153     continue;
154 }
155 else if (_stricmp(command[0],"read")==0) {
156     fd = GetFDFromName(command[1]);
157     if (fd == -1) {
158         printf("Error : Incorrect parameters\n" );
159         continue;
160     }
161     ptr = (char *)malloc(sizeof(atoi(command[2]))+1);
162     if(ptr == NULL)
163     {
164         printf("Error : Incorrect parameters\n" );
165         continue;
166     }
167     ret = ReadFile(fd,ptr,atoi(command[2]));
168     if (ret == -1)
169         printf("Error : File not existing\n" );
170     if (ret == -2)
171         printf("Error : Permission Denied\n" );
172     if (ret == -3)
173         printf("Error : Reached at the end of file\n" );
174     if (ret == -4)
175         printf("Error : It is not a regular file\n" );
176     if (ret == 0)
177         printf("Error : File empty\n" );
178     if (ret > 0) {
179         _write(2,ptr,ret);
180     }
181     continue;
182 }
183 else{
184     printf("\nError : Command not found !!!\n" );
185     continue;
186 }
187 }
188 else if (count == 4) {
189     if (_stricmp(command[0],"lseek")==0) {
190         fd=GetFDFromName(command[1]);
191         if (fd == -1) {
192             printf("Error : Incorrect parameters\n" );
193             continue;
194         }
195         ret = LseekFile(fd,atoi(command[2]),atoi(command[3]));
196         if (ret == -1) {
197             printf("Error : Unable to perform lseek\n" );
198         }

```

```
199         }
200         else{
201             printf("\nError: Command Not Found\n" );
202             continue;
203         }
204     }
205     else
206     {
207         printf("\nError : Command not found\n" );
208         continue;
209     }
210 }
211 return 0;
212 }
213
```