

## Terza simulazione

```
1  #include<stdio.h>
2  #include<stdlib.h>
3  #include<string.h>
4  #include<unistd.h>
5  #include<sys/wait.h>
6
7  #define MAXLEN 256
8
9  FILE *in;
10 int caratteri=0,linee=0;
11
12 void file_check( char * path){
13     in= fopen(path,"r");
14     if(in==NULL){
15         fprintf(stderr,"File non aperto\n");
16         exit(1);
17     }
18 }
19
20 void logica_figlio(int argc,char ** argv){
21     if(argc==3){
22         while(feof(in)==0){
23             char c= getc(in);
24             if(c=='\n'){
25                 linee++;
26             }
27             else{
28                 caratteri++;
29             }
30         }
31         if(strcmp(argv[1],"-l")==0){
32             fprintf(stdout,"%d\n",linee);
33         }
34         else{
35             fprintf(stdout,"%d\n",caratteri-1);
36         }
37     }
38     else{
39         while(feof(in)==0){
40             char c= getc(in);
41             if(c=='\n'){
42                 linee++;
43             }
44             else{
45                 caratteri++;
46             }
47         }
48     }
49 }
```

```

48     if(strcmp(argv[1],"-c")==0 && strcmp(argv[2],"-c")==0){
49         fprintf(stdout,"%d %d\n",caratteri-1,linee);
50     }
51     else{
52         fprintf(stdout,"%d %d\n",linee,caratteri-1);
53     }
54 }
55 }
56
57 void logica_pipe(int argc,char ** argv){
58     char tmp;
59     while(read(STDIN_FILENO,&tmp,1)>0){
60         if(tmp=='\n'){
61             linee++;
62         }
63         else{
64             caratteri++;
65         }
66     }
67     caratteri--;
68     if(argc==3){
69         if(strcmp(argv[1],"-l")==0 && strcmp(argv[2],"-c")==0){
70             printf("%d %d\n",linee,caratteri);
71         }
72         else if(strcmp(argv[1],"-c")==0 && strcmp(argv[2],"-l")==0){
73             printf("%d %d\n",caratteri,linee);
74         }
75     }
76     else{
77         if(strcmp(argv[1],"-c")==0){
78             printf("%d\n",caratteri);
79         }
80         else if(strcmp(argv[1],"-l")==0){
81             printf("%d\n",linee);
82         }
83     }
84 }
85
86 void arg_check(int argc, char **argv){
87     if(argc<2 || argc>4){
88         fprintf(stderr,"Argomenti errati\n");
89         exit(2);
90     }
91     else if(!isatty(fileno(stdin)) &&
92             (strcmp(argv[1],"-c")==0 || strcmp(argv[1],"-l")==0) ||
93             (strcmp(argv[1],"-l")==0 && strcmp(argv[2],"-c")==0) ||
94             (strcmp(argv[1],"-c")==0 && strcmp(argv[2],"-l")==0)){
95         int figlio=fork();
96         if(figlio==0){
97             logica_pipe(argc,argv);
98         }
99         else if(figlio>0){
100             while(wait(NULL)>0);
101         }
102         else{
103             fprintf(stderr,"Figlio non creato\n");

```

```

104     }
105 }
106 else if((strcmp(argv[1], "-l")==0 && strlen(argv[2])<=2) ||
107         (strcmp(argv[1], "-c")==0 && strlen(argv[2])<=2)){
108     fprintf(stderr, "Argomenti errati\n");
109     exit(3);
110 }
111 else if((strcmp(argv[1], "-l")==0 && strcmp(argv[2], "-c")==0) ||
112         strcmp(argv[1], "-c")==0 && strcmp(argv[2], "-l")==0){
113     file_check(argv[3]);
114     int figlio=fork();
115     if(figlio==0){
116         logica_figlio(argc, argv);
117     }
118     else if(figlio>0){
119         while(wait(NULL)>0);
120     }
121     else{
122         fprintf(stderr, "Figlio non creato\n");
123         exit(4);
124     }
125 }
126 else if((strcmp(argv[1], "-l")==0 && strlen(argv[2])>2) ||
127         (strcmp(argv[1], "-c")==0 && strlen(argv[2])>2)){
128     file_check(argv[2]);
129     int figlio=fork();
130     if(figlio==0){
131         logica_figlio(argc, argv);
132     }
133     else if(figlio>0){
134         while(wait(NULL)>0);
135     }
136     else{
137         fprintf(stderr, "Figlio non creato\n");
138         exit(4);
139     }
140 }
141 else{
142     fprintf(stderr, "Argomenti errati\n");
143     exit(5);
144 }
145 }
146
147 int main(int argc, char ** argv) {
148     arg_check(argc, argv);
149     return 0;
150 }

```