```c
1   #if VERSIONE == 1
2   #include <stdio.h>
3   #include <stdlib.h>
4
5   int main(int argc, char *argv[])
6   {
7       FILE *fin = stdin, *fout = stdout;
8       int n=10;
9       switch(argc) {
10          case 4:
11              n = atoi(argv[3]);
12          case 3:
13              if(argv[2][0]!='-') {
14                  fout = fopen(argv[2], "w");
15                  if(!fout) {
16                      perror(argv[2]);
17                      return(2);
18                  }
19              }
20          case 2:
21              if(argv[1][0]!='-') {
22                  fin = fopen(argv[1], "r");
23                  if(!fin) {
24                      perror(argv[1]);
25                      return(1);
26                  }
```

```
27                  }
28                  if(argv[1][1]=='v') {
29                      fprintf(stderr, "%s VERSIONE=%d\n", argv[0],
                        VERSIONE);
30                      return 0;
31                  }
32          case 1:
33              break;
34          default:
35              fprintf(stderr, "uso: %s [input-file [output-file
                [numero di interi per riga]]]\n", argv[0]);
36              return(3);
37      }
38      int tmp, i=n;
39      while(fscanf(fin, "%d", &tmp) == 1) {
40          fprintf(fout, "%d", tmp);
41          if(i==1) {
42              // putc scrive un carattere in un file
43              putc('\n', fout);
44              i=n;
45          } else {
46              putc(' ', fout);
47              i--;
48          }
49      }
50      if(i<n) putc('\n', fout);
```

```
51        return 0;
52  }
53  #endif /* VERSIONE == 1 */
```

```
54   #if VERSIONE == 2
55   #include <stdio.h>
56
57   int main(int argc, char *argv[])
58   {
59       FILE *fin = stdin, *fout = stdout;
60       switch(argc) {
61           case 3:
62               if(argv[2][0]!='-') {
63                   fout = fopen(argv[2], "w");
64                   if(!fout) {
65                       perror(argv[2]);
66                       return(2);
67                   }
68               }
69           case 2:
70               if(argv[1][0]!='-') {
71                   fin = fopen(argv[1], "r");
72                   if(!fin) {
73                       perror(argv[1]);
74                       return(1);
75                   }
76               }
77               if(argv[1][1]=='v') {
78                   fprintf(stderr, "%s VERSIONE=%d\n", argv[0],
                        VERSIONE);
```

```
79                       return 0;
80                   }
81             case 1:
82                 break;
83             default:
84                 fprintf(stderr, "uso: %s [input-file [output-file]]\n",
                        argv[0]);
85                 return(3);
86         }
87         char tmp[100];
88         while(fgets(tmp, sizeof(tmp), fin)) {
89             fputs(tmp, fout);
90         }
91         return 0;
92     }
93  #endif /* VERSIONE == 2 */
```

```
 94   #if VERSIONE == 3
 95   #include <stdio.h>
 96
 97   int main(int argc, char *argv[])
 98   {
 99       FILE *fin = stdin, *fout = stdout;
100       switch(argc) {
101           case 3:
102               if(argv[2][0]!='-') {
103                   fout = fopen(argv[2], "w");
104                   if(!fout) {
105                       perror(argv[2]);
106                       return(2);
107                   }
108               }
109           case 2:
110               if(argv[1][0]!='-') {
111                   fin = fopen(argv[1], "r");
112                   if(!fin) {
113                       perror(argv[1]);
114                       return(1);
115                   }
116               }
117               if(argv[1][1]=='v') {
118                   fprintf(stderr, "%s VERSIONE=%d\n", argv[0],
                       VERSIONE);
```

```
119                    return 0;
120                }
121            case 1:
122                break;
123            default:
124                fprintf(stderr, "uso: %s [input-file [output-file]]\n",
                       argv[0]);
125                return(3);
126        }
127        int tmp;
128        while( (tmp=getc(fin)) >= 0) {
129            putc(tmp, fout);
130        }
131        return 0;
132    }
133    #endif /* VERSIONE == 3 */
134
```