FND12

readin.c source code

```c
#include <stdio.h>
#include <stdlib.h>
int main(int argc, int* argv[])
{
    FILE *pFIn = fopen(argv[1], "r");
    char c = fgetc(pFIn);
    if (pFIn == (FILE*)NULL)
    {
        fprintf(stderr, "%s does not exist.\n", argv[1]);
        return EXIT_FAILURE;
    }
    while (c != EOF)
    {
        printf("%c", c);
        c = fgetc(pFIn);
    }
    fclose(pFIn);
    return EXIT_SUCCESS;
}
```

writeout.c source code

```c
1   #include <stdlib.h>
2   #include <stdio.h>
3
4
5   int main(int argc, char* argv[])
6   {
7       FILE* pFIn;
8       pFIn = fopen(argv[1], "w");
9       char c;
10      c = getc(stdin);
11      if(pFIn == (FILE*)NULL)
12      {
13          fprintf(stderr, "%s does not exist.\n", argv[1]);
14          return EXIT_FAILURE;
15      }
16
17
18      while(c != EOF){
19          fprintf(pFIn, "%c", c);
20          c = getc(stdin);
21      }
22      fclose(pFIn);
23      return EXIT_SUCCESS;
24  }
25
```

Terminal Output

```
[mmoustaf@gc112m38 FND12]$ gcc -o readin readin.c
readin.c: In function 'main':
readin.c:5:5: warning: passing argument 1 of 'fopen' from incompatible pointer type [enabled by default]
    FILE *pFIn = fopen(argv[1], "r");
    ^

In file included from readin.c:1:0:
/usr/include/stdio.h:272:14: note: expected 'const char * __restrict__' but argument is of type 'int *'
 extern FILE *fopen (const char *__restrict __filename,
               ^

[mmoustaf@gc112m38 FND12]$ gcc -o writeout writout.c
[mmoustaf@gc112m38 FND12]$ ./readin otherFile.txt | ./writeout out.txt
[mmoustaf@gc112m38 FND12]$ diff otherFile.txt out.txt
[mmoustaf@gc112m38 FND12]$ ▊
```

Out.txt

Every hour that you're behind the keyboard
is sixty minutes that you're not doing something else.
                    -Clifford Stoll