

Sample Output:

```
Bonny@DESKTOP-7361LLA ~/OS
$ ./kitty 3 #include <stdio.h>
hello testhello test10 bytes is transfered; 2 times of read from <standard input>
>; 1 times of write to <standard output>
```

```
Bonny@DESKTOP-7361LLA ~/OS
$ ./kitty -o output
hello
this is a test
21 bytes is transfered; 3 times of read from <standard input>; 2 times of write
to output
WARN: <standard input> is binary file
```

```
Bonny@DESKTOP-7361LLA ~/OS
$ ./kitty -- we
hello
hello
6 bytes is transfered; 2 times of read from <standard input>; 1 times of write to <standard output>
WARN: <standard input> is binary file
test
test
5 bytes is transfered; 2 times of read from <standard input>; 1 times of write to <standard output>
WARN: <standard input> is binary file
heelooooejfere
rererersa
25 bytes is transfered; 2 times of read from we; 1 times of write to <standard output>
WARN: we is binary file
```

```
Bonny@DESKTOP-7361LLA ~/OS
$ ./kitty we
heelooooejfere
rererersa
25 bytes is transfered; 2 times of read from we; 1 times of write to <standard output>
WARN: we is binary file
```

```
Bonny@DESKTOP-7361LLA ~/OS
$ ./kitty -o output -- we
hello
6 bytes is transfered; 2 times of read from <standard input>; 1 times of write to output
WARN: <standard input> is binary file
test
5 bytes is transfered; 2 times of read from <standard input>; 1 times of write to output
WARN: <standard input> is binary file
25 bytes is transfered; 2 times of read from we; 1 times of write to output
WARN: we is binary file
```

```
Bonny@DESKTOP-7361LLA ~/OS
$ ./kitty hello
ERROR: hello is not a name of a valid file with error No such file or directory
```

Bonny@DESKTOP-7361LLA ~/OS

\$./kitty -o output we

ERROR: Fail to open/create output for writing with error Permission denied

Bonny@DESKTOP-7361LLA ~/OS

\$./kitty -o outfile we

69299 bytes is transfered; 18 times of read from we; 17 times of write to outfile

WARN: we is binary file

Bonny@DESKTOP-7361LLA ~/OS

\$./kitty -o outfile we

ERROR: an error occurred when opening we with error Permission denied

Other situation cannot be tested but reflected in the codes:

Partial Write Situation:

```
if((byteWrite = write(fout, buf, byteRead)) < byteRead){
    int s = 0;
    int j;
    for(j = byteWrite; j < byteRead; j++){
        buf[s] = buf[j];
        s++;
    }
    if(write(fout, buf, (byteRead - byteWrite)) < (byteRead - byteWrite)){
        //Partial write again
        fprintf(stderr, "ERROR: Only a part of data is wrote to %s from %s", inName, outName);
        return -1;
    }
}
```

Error Close:

```
if(fout != 1){
    if(close(fout) < 0){
        fprintf(stderr, "ERROR: Fail to close %s with error %s\n", outName, strerror(errno));
        return -1;
    }
}
return 0;
```