

```

1  /*-----+
2  File Name: format.c
3  +-----+
4
5  +-----+
6  Programming III COP4338
7  Author: Daniel Gonzalez P#4926400
8  assignment 5: Date Validate / Format
9  Date: 11/08/2016 ELECTION DAY!!!!
10
11 program description
12     Input: Accept input for the first program via the command-line arguments.
13           Input will be the number of valid entries to be redirected from
14           the dates input file (dates.dat). A zero indicates to input all
15           entries from the dates input file. Your program should validate
16           (day, month & year - see page 111 for validation ideas) and skip
17           corrupt dates in the dates.dat file (see page 159 for scanning
18           ideas). This validated input will then be piped out to the
19           second program.
20           The second program will accept these validated dates in the
21           month/day/year format and convert each of them to the day,
22           abbreviated month & year format - both exhibited above. The
23           abbreviated month should consist of the first three letters of
24           the month, capitalized. These converted results will be redirected
25           to the output file (output.dat), followed by a copy of the
26           original (dates.dat) data.
27
28     Output: Generates an output file (output.dat) that contains a
29           converted list of dates in day, abbreviated month & year
30           format (i.e. 1 JAN 1900), followed by the original list of
31           dates in month/day/year format (i.e. 1/1/1900). This output file
32           will be the result of appending the input file (dates.dat), which
33           is accessed by the first program, with the result output file
34           (output.dat), generated by the second program.
35
36
37  +-----+
38  | I Daniel Gonzalez #4926400 hereby certify that this collective work is |
39  | my own and none of it is the work of any other person or entity.      |
40  +-----+
41
42
43 how to compile and execute:
44     1.Open the terminal
45       Go to the program folder that contains all the files required for
46       the program to compile including all header files(*.h).
47       Run the following command "make"
48
49     2.Open the terminal
50       Go to the program folder that contains all the files required for
51       the program to compile including all header files(*.h).
52       COMPILE: "gcc -Wall -w -lm readDate.c dateValidate.c -o validateDate"
53
54 Program execution:
55 From the terminal enter:
56     "./validateDate < dates.dat [X] | ./format > output.dat"
57
58     X: is the amount of validated dates
59
60
61
62
63  +-----*/
64
65 #include "general.h"
66
67
68
69

```

```

70 /**-----+
71     //Get a line of data.
72     //Print data with correct format.
73     //Print original data. */
74 int main(int argc, char *argv[])
75 {
76     /* Declare variables */
77     DateKey date;
78     Data data;
79
80     /* Main process */
81     while (fgets(data.input, MAX_LINE, stdin) != NULL){
82
83         if(data.input[0] == HAND_SHAKING_SIGNAL){
84             break;
85         }else{
86
87             date = getDateKeys(data.input);
88             printfDate(data.output, DATE_FORMAT_SHORT_MONTH, &date);
89             fputs(data.output, stdout);
90         }
91     }
92
93     while (fgets(data.input, MAX_LINE, stdin) != NULL) {
94         fputs(data.input, stdout);
95     }
96
97     /* Output results */
98     return NO_ERRORS;
99 }

```