

EGR 226: Microcontroller Programming and Applications

Winter 2021

Instructor: Prof. Trevor Ekin

Lab 2: C-Programming Refresher 2

Jake Carlson

February 3, 2021

Contents

| | |
|------------------------------------------------------|-----------|
| 1. Objectives | 3 |
| 2. Introduction | 3 |
| 2.1. Part 1: Resistance Value Calculator | 3 |
| 2.2. Part 2: Book Database | 3 |
| 3. Procedure | 3 |
| 3.1. Part 1: Resistance Value calculator | 3 |
| 3.1.1. Discussing the Steps | 3 |
| 3.1.2. Band 1 Assignment in calcResistance | 3 |
| 3.2. Part 2: Book Database | 4 |
| 3.2.1. Discussing the Steps | 4 |
| 3.2.2. Print Book Function | 4 |
| 4. Results/Discussion | 5 |
| 4.1. Conclusion | 5 |
| 4.2. Example Results | 5 |
| 4.3. Part 1 example | 5 |
| 4.4. Part 2 Title example | 6 |
| 4.5. Part 2 Author example | 6 |
| 4.6. Part 2 ISBN example | 6 |
| Appendices | 8 |
| A. Source Code: Lab 2 Part 1 main.c | 8 |
| B. Source Code: Lab 2 Part 2 main.c | 14 |

1. Objectives

The objective of this lab was to go more in depth into c programming with another refresher. In this lab a new program was created from building off of the previous resistor color-band calculator which now also calculates the resistance value after a user inputs each color of the four bands. In the second part of this lab a new program needed to use structures to make a searchable database of books using a .csv file.

2. Introduction

2.1. Part 1: Resistance Value Calculator

In the first part of Lab 2 a new program was created from part 1 of Lab 1 and adding on to it. New functions were needed in order to do the opposite of the previous program which is calculating resistance values from the users input of colors for each color band. These two programs need to work in unison in order for the user to be able to choose if they want to convert from resistance value to color bands, or color bands to resistance value.

2.2. Part 2: Book Database

In the second part of Lab 2 a program was created that searches an entire database of books using structures. This program allows a user to input a title, author, or ISBN, as search criteria in order to locate the correct book.

3. Procedure

3.1. Part 1: Resistance Value calculator

3.1.1. Discussing the Steps

This program was created by using part 1 from the previous lab and adding new functions such as getColorBands, and calcResistance. The function getColorbands collects the users input of 4 colors for each of the 4 bands. The function calcResistance is used to assign each band with the correct resistance value associated with the color. This function then prints the resistance value and tolerance. Finally, this program needs to work with the first program, so the original prompt had to be altered for the user to decide which calculator is necessary.

3.1.2. Band 1 Assignment in calcResistance

This was a more challenging part of the program because the first 2 bands need to be added together in order to hold the correct value. When simply adding the 2 bands together the incorrect value will be calculated. To find the correct value band 1 needs to increment by 10 for each color

since Brown + Brown should be 11. If both brown bands were assigned to 1 then the value would be 2 which is incorrect. an example is shown below.

```

178
179 //band 1
180 if(c1 == 'K'){//assigning band 1 with 0-90
181     b1 = 0;
182 }
183 else if(c1 == 'N'){//going by increments of 10 since it is added to band 2 later
184     b1 = 10;
185 }
186 else if(c1 == 'R'){
187     b1 = 20;
188 }
189 else if(c1 == 'O'){
190     b1 = 30;
191 }
192 else if(c1 == 'Y'){
193     b1 = 40;
194 }
195 else if(c1 == 'G'){
196     b1 = 50;
197 }
198 else if(c1 == 'B'){
199     b1 = 60;
200 }
201 else if(c1 == 'V'){
202     b1 = 70;
203 }
204 else if(c1 == 'E'){
205     b1 = 80;
206 }
207 else {
208     b1 = 90;
209 }

```

3.2. Part 2: Book Database

3.2.1. Discussing the Steps

This program was created by defining a book structure that includes the title, author, ISBN, pages, and year published. Then a parse file function was created in order to open the file and scan in the book information into the book array. After this a print book function was created to print the book information in a clean manor so the user can read the output. Finally, three functions for searching the title, author, and ISBN were created so that the user can choose what to search by and these functions will search the database. After searching these functions pass back the correct strings to be printed.

3.2.2. Print Book Function

This is a challenging function because not only does the book need to be printed neatly, but it also needs to print the correct book that the user is looking for. Another challenge was that some information in the book database is not given such as pages or the year published, so there needs to be a solution for that issue as well. Print book function can be seen below.

```

140
141 void print_book(book entered_array[], int enter){
142
143     int i;
144
145     for(i = 0; i < enter; i++){//loop that prints title, author, ISBN, page, and year published in that order
146         printf("Title: %s\n", entered_array[i].title);//title print
147         printf("Author: %s\n", entered_array[i].author_name);//author print
148         printf("ISBN: %s\n", entered_array[i].ISBN);//ISBN print
149
150         if(entered_array[i].pages == -1){
151             printf("Pages: N/A\n");//used in case pages are N/A
152         }
153         else{
154             printf("Pages: %d\n", entered_array[i].pages);//prints pages
155             fflush(stdout);
156         }
157         if(entered_array[i].year_published == -1){
158             printf("Published: N/A\n");//used if year published is N/A
159         }
160         else{
161             printf("Published: %d\n", entered_array[i].year_published);//prints year published
162         }
163     }
164     return (0);
165 }
166

```

4. Results/Discussion

4.1. Conclusion

In conclusion there were multiple challenges in this c programming refresher lab. Many of them were overcome by spending extra time to think through the process. One of the challenges was searching through the book data base, but an example was given in the lab manual to help overcome the challenge. Overall this lab was another great refresher and will ensure that each student is ready to continue.

4.2. Example Results

4.3. Part 1 example

Running the program in Part 1 entering Brown, Brown, Brown, Brown.

```

W:\EGR226\EGR226_Jake_Carlson\Lab2Part1\bin\Debug\Lab2Part1.exe

```

| Character | Color | Band 1 & 2 | Band 3 | Tolerance |
|-----------|--------|------------|-------------|-----------|
| K | Black | 0 | *1 | |
| N | Brown | 1 | *10 | +/-1% |
| R | Red | 2 | *100 | +/-2% |
| O | Orange | 3 | *1,000 | |
| Y | Yellow | 4 | *10,000 | |
| G | Green | 5 | *100,000 | +/-0.5% |
| B | Blue | 6 | *1,000,000 | +/-0.25% |
| V | Violet | 7 | *10,000,000 | +/-0.1% |
| E | Grey | 8 | | +/-0.05% |
| W | White | 9 | | |
| D | Gold | | *0.1 | +/-5% |
| S | Silver | | *0.01 | +/-10% |

```

-----
Please enter a 1 to convert a resistance value to a color code.
Please enter a 2 to convert a color code to a resistance value.
2
Enter A Character for band 1:N
Enter A Character for band 2:N
Enter A Character for band 3:N
Enter A Character for band 4:N
Resistor color code entered: N, N, N, N
This resistor is [110]Ohms with a +/-2% tolerance
Process returned 0 (0x0)   execution time : 6.968 s
Press any key to continue

```

4.4. Part 2 Title example

Running the program in Part 2 entering the title "sapien".

```
W:\EGR226\EGR226_Jake_Carlson\Lab2Part2\bin\Debug\Lab2Part2.exe
What would you like to search by? [0] Title, [1] Author, or [2] ISBN
0
Enter Title criteria:
sapien
Title: Sapiens: A Brief History of Humankind
Author: Yuval Noah Harari
ISBN: N/A
Pages: 443
Published: 2014
What would you like to search by? [0] Title, [1] Author, or [2] ISBN
-
```

4.5. Part 2 Author example

Running the program in Part 2 entering the author "yuval".

```
W:\EGR226\EGR226_Jake_Carlson\Lab2Part2\bin\Debug\Lab2Part2.exe
What would you like to search by? [0] Title, [1] Author, or [2] ISBN
1
Enter Author criteria:
yuval
Title: Sapiens: A Brief History of Humankind
Author: Yuval Noah Harari
ISBN: N/A
Pages: 443
Published: 2014
What would you like to search by? [0] Title, [1] Author, or [2] ISBN
-
```

4.6. Part 2 ISBN example

Running the program in Part 2 entering the ISBN "140443223".

```
W:\EGR226\EGR226_Jake_Carlson\Lab2Part2\bin\Debug\Lab2Part2.exe
What would you like to search by? [0] Title, [1] Author, or [2] ISBN
2
Enter ISBN criteria:
140443223
Title: The Mabinogion
Author: Unknown
ISBN: 140443223
Pages: 311
Published: 1976
What would you like to search by? [0] Title, [1] Author, or [2] ISBN
```

Appendices

A. Source Code: Lab 2 Part 1 main.c

```

1  /*
    *****
2  * Author: Jake Carlson
3  * Course: EGR 226 — 902
4  * Date: 02/1/2021
5  * Project: lab21part1
6  * File: lab2part1main.c
7  * Description: This program takes an input from the user in ohms and solves to find
    the
8  * resistors color-bands or takes color-bands and converts to resistance.
9  *
10 *****
    */
11
12
13 #include <stdio.h>
14 #include <stdlib.h>
15
16 void getColorBands(char*, char*, char*, char*);
17 void calcResistance(char, char, char, char);
18 void prompt(void); //Declaring function prototypes
19 int calcResistorColors(int x);
20 int getIntBetween ();
21 int getInput();
22
23 int main(){
24
25 int enteredVal;
26 int userChoice;
27 int goodEnter = 0;
28 char c1, c2, c3, c4;
29
30
31 prompt();
32 do{
33     printf("Please enter a 1 to convert a resistance value to a color code.\n"); //
    new user prompt
34     printf("Please enter a 2 to convert a color code to a resistance value.\n");
35     scanf("%d", &userChoice);
36
37     if(userChoice == 1){ //if user chooses 1 then call functions from previous lab
38         enteredVal = getIntBetween(); //function calls

```



```

39     calcResistorColors(enteredVal);
40     goodEnter = 1;
41 }
42 else if(userChoice == 2){//if 2 call new functions and new prompt
43     printf("Enter A Character for band 1:");
44     scanf(" %c", &c1);
45     printf("\nEnter A Character for band 2:");
46     scanf(" %c", &c2);
47     printf("\nEnter A Character for band 3:");
48     scanf(" %c", &c3);
49     printf("\nEnter A Character for band 4:");
50     scanf(" %c", &c4);
51
52     getColorBands(c1, c2, c3, c4);
53     calcResistance(c1, c2, c3, c4);
54     goodEnter = 1;
55 }
56
57 else{
58     printf("Error\n");//error check
59     goodEnter = 0;
60 }
61 } while(goodEnter != 1);
62
63 return (0);}
64
65
66 void prompt(void){ //printf statements for prompt
67
68     printf("-----Resistor Codes-----\n");
69     printf("| Character | Color | Band 1 & 2 | Band 3 | Tolerance |\n");
70     printf("| K | Black | 0 | *1 | +/-1%% |\n");
71     printf("| N | Brown | 1 | *10 | +/-2%% |\n");
72     printf("| R | Red | 2 | *100 | |\n");
73     printf("| O | Orange | 3 | *1,000 | |\n");
74     printf("| Y | Yellow | 4 | *10,000 | |\n");
75     printf("| G | Green | 5 | *100,000 | +/-0.5%% |\n");
76     printf("| B | Blue | 6 | *1,000,000 | +/-0.25%% |\n");
77     printf("| V | Violet | 7 | *10,000,000 | +/-0.1%% |\n");
78     printf("| E | Grey | 8 | | +/-0.05%% |\n");
79     printf("| W | White | 9 | | |\n");
80     printf("| D | Gold | | *0.1 | +/-5%% |\n");
81     printf("| S | Silver | | *0.01 | +/-10%% |\n");
82     printf("-----\n\n");
83
84 }
85 int getIntBetween (int val){
86

```

```
87     printf("Please enter a value from 1 to 99M ohms\n");
88     val = 0; //Value entered
89     char wrong[100]; //storage for error when using alphabet
90     int pro, i=0;
91
92 while(i == 0){ //while loop to make sure value is between 1 and 99
93
94     pro= scanf("%d", &val);
95
96     if (pro == 1){
97         if (val > 0 && val <= 99000000){//checking value
98             i += 1;
99         }
100        else {
101            printf("Please enter a value from 1 to 99M ohms\n");
102        }
103    }
104    else{
105        scanf("%s", &wrong); //saving the incorrect characters to wrong
106        printf("Please enter a value from 1 to 99M ohms\n");
107    }
108 }
109 return (val);
110
111 }
112
113
114
115 int calcResistorColors(int enteredVal){
116
117     char colors [10][10] = {
118         "Black", "Brown", "Red", "Orange", "Yellow", "Green", "Blue", "Violet", "Grey",
119         "White" //array for colors of resistor bands
120     };
121
122     int b1, b2, b3; //declaring band 1 band 2 and band 3;
123     int counter = 0;
124
125     printf("A resistor of %d Ohms would have a color code of: ", enteredVal);
126
127     do{
128         enteredVal = enteredVal/10;
129         counter = counter +1;
130     }
131     while (enteredVal >= 100); //do while loop to divide the resistor value by 10 and
132     increment a counter variable
```

```
133     if (counter == 0){//assigning band 3 with 0-7
134         b3 =0;
135     }
136     else if (counter ==1){
137         b3 =1;
138     }
139     else if (counter ==2){
140         b3 =2;
141     }
142     else if (counter ==3){
143         b3 =3;
144     }
145     else if (counter ==4){
146         b3 =4;
147     }
148     else if (counter ==5){
149         b3 =5;
150     }
151     else if (counter ==6){
152         b3 =6;
153     }
154     else if (counter ==7){
155         b3 =7;
156     }
157         b1 = enteredVal / 10;
158         b2 = enteredVal % 10;
159
160 printf(" %s-%s-%s", colors[b1], colors[b2], colors[b3]); //final printf
161
162
163 return (enteredVal);}
164
165 void getColorBands(char* c1, char* c2, char* c3, char* c4){//new function that
    passes with pointers
166     char color[4] = {c1, c2, c3, c4};
167
168     printf("Resistor color code entered: %c, %c, %c, %c\n", c1, c2, c3, c4);
169
170     return (0);
171 }
172
173 void calcResistance(char c1, char c2, char c3, char c4){
174
175
176     int b1, b2, b3, total;
177     double b4;
178
179 //band 1
```

```
180     if(c1 == 'K'){//assigning band 1 with 0-90
181         b1 = 0;
182     }
183     else if(c1 == 'N'){//going by increments of 10 since it is added to band 2 later
184         b1 = 10;
185     }
186     else if(c1 == 'R'){
187         b1 = 20;
188     }
189     else if(c1 == 'O'){
190         b1 = 30;
191     }
192     else if(c1 == 'Y'){
193         b1 = 40;
194     }
195     else if(c1 == 'G'){
196         b1 = 50;
197     }
198     else if(c1 == 'B'){
199         b1 = 60;
200     }
201     else if(c1 == 'V'){
202         b1 = 70;
203     }
204     else if(c1 == 'E'){
205         b1 = 80;
206     }
207     else {
208         b1 = 90;
209     }
210
211 //band 2
212
213     if(c2 == 'K'){//assigning band 2 with 0-9
214         b2 = 0;
215     }
216     else if(c2 == 'N'){
217         b2 = 1;
218     }
219     else if(c2 == 'R'){
220         b2 = 2;
221     }
222     else if(c2 == 'O'){
223         b2 = 3;
224     }
225     else if(c2 == 'Y'){
226         b2 = 4;
227     }
```

```
228     else if(c2 == 'G'){
229         b2 = 5;
230     }
231     else if(c2 == 'B'){
232         b2 = 6;
233     }
234     else if(c2 == 'V'){
235         b2 = 7;
236     }
237     else if(c2 == 'E'){
238         b2 = 8;
239     }
240     else {
241         b2 = 9;
242     }
243
244 //band 3
245
246     if(c3 == 'K'){//assigning band 3 with appropriate multiplier
247         b3 = 1;
248     }
249     else if(c3 == 'N'){
250         b3 = 10;
251     }
252     else if(c3 == 'R'){
253         b3 = 100;
254     }
255     else if(c3 == 'O'){
256         b3 = 1000;
257     }
258     else if(c3 == 'Y'){
259         b3 = 10000;
260     }
261     else if(c3 == 'G'){
262         b3 = 100000;
263     }
264     else if(c3 == 'B'){
265         b3 = 1000000;
266     }
267     else if(c3 == 'V'){
268         b3 = 10000000;
269     }
270     else if(c3 == 'D'){
271         b3 = 0.1;
272     }
273     else if(c3 == 'S'){
274         b3 = 0.01;
275     }
```

```

276
277     else {
278         b3 = 0;
279     }
280
281     //band 4
282     if(c4 == 'K'){//assigning band 4 with appropriate tolerance
283         b4 = 1;
284     }
285     else if(c4 == 'N'){
286         b4 = 2;
287     }
288     else if (c4 == 'G'){
289         b4 = 0.5;
290     }
291     else if(c4 == 'B'){
292         b4 = 0.25;
293     }
294     else if(c4 == 'V'){
295         b4 = 0.1;
296     }
297     else if(c4 == 'E'){
298         b4= 0.05;
299     }
300     else if(c4 == 'D'){
301         b4 = 5;
302     }
303     else if(c4 == 's'){
304         b4 = 10;
305     }
306     else {
307         b4 = 0;
308     }
309
310     total = ((b1+b2)*b3);//adding band 1 and 2 then multiplying by the multiplier
311
312     printf("This resistor is [%d]Ohms with a +/-%g%% tolerance.", total, b4);//final
        print
313
314
315 return (0);
316 }

```

B. Source Code: Lab 2 Part 2 main.c

```

1 /*
    *****

```

```

2 * Author: Jake Carlson
3 * Course: EGR 226 - 902
4 * Date: 02/1/2021
5 * Project: lab2part2
6 * File: main.c
7 * Description: This program uses structures to create a database of books from
8 * an external file.
9 *
10 *****
    */
11
12 #include <stdio.h>
13 #include <stdlib.h>
14 #include <string.h>
15 #include <ctype.h>
16 #ifndef NULL
17 #define NULL 0
18 #endif // NULL
19
20 typedef struct{//Defining the book structure
21     char title[255];
22     char author_name[50];
23     char ISBN[10];
24     int pages;
25     int year_published;
26 }book;
27 int input(int low, int high); //Functions used in program
28 int parse_file(char name[], book my_book_array[]);
29 void print_book(book entered_array[], int enter);
30 void search_title(book my_book_array[]);
31 void search_author(book my_book_array[]);
32 void search_ISBN(book my_book_array[]);
33
34
35
36 int main()
37 {
38     int i, j = 1, low = 0, high = 2;
39     char name[25] = "BookList.csv";
40     book my_book_array[360];
41
42
43     parse_file(name, my_book_array); //calling function to fill array
44
45     while(j){
46         i = input(low, high); //calling input function to use what user has entered
47         if(i == 0){
48             search_title(my_book_array); // if 0 the program will use title function

```

```
49  }
50  else if(i == 1){
51      search_author(my_book_array); //if 1 the program will use author function
52  }
53  else{
54      search_ISBN(my_book_array); // if 2 the program will use ISBN function
55  }}
56
57      return 0;
58
59 }
60 int input(int low, int high){
61
62     int test = 1;
63     int user, loop;
64     char che[10];
65
66     printf("What would you like to search by? [0] Title, [1] Author, or [2] ISBN\n")
67     ; //user prompt
68
69     while(test){ //error checking loop
70         loop = scanf("%d", &user); //sets whatever the user enters to loop
71         if(loop != 1){
72             printf("Try again.\n");
73         }
74         else if (user > high){
75             printf("Try again.\n"); //else if's to make sure the user enters between
76             0 and 2
77         }
78         else if (user < low){
79             printf("Try again.\n");
80         }
81         else{
82             test = 0;
83         }
84         fflush(stdin);
85     }
86
87     che[0] = user;
88     che[strlen(che) - 2] = '\0'; //using array to set the entry to the first value
89
90     user = che[0];
91
92     return user;
93 }
94 int parse_file(char name[], book my_book_array[]){
```



```
95     int i = 0;
96     char chec[512];
97
98     FILE* infile = fopen(name, "r");//opening file and returning 0 in case the file
    does not exist
99     if (infile == NULL){
100         return 0;
101     }
102
103     while(fgets(chec, 512, infile)){//loop to collect each line from given csv file
104
105         char* point = strtok(chec, ",");//adding commas and newline
106
107         if(strcmp(point, "N/A"))//validate string
108             strcpy(my_book_array[i].title, point);//title parse
109
110         point = strtok(NULL, ",\n");
111         if(strcmp(point, "N/A"))//validate string
112             strcpy(my_book_array[i].author_name, point);//author parse
113
114         point = strtok(NULL, ",\n");
115         if(strcmp(point, "N/A"))//validate string
116             strcpy(my_book_array[i].ISBN, point);//ISBN parse
117
118         else
119             strcpy(my_book_array[i].ISBN, "N/A");
120
121         point = strtok(NULL, ",\n");
122         if (strcmp(point, "N/A")){//validate string
123             my_book_array[i].pages = atoi (point);//page count parse
124         }
125         else {
126             my_book_array[i].pages = -1;
127         }
128         point = strtok(NULL, ",\n");
129         if(strcmp(point, "N/A"))//validate string
130             my_book_array[i].year_published = atoi(point);//year published parse
131
132         else {
133             my_book_array[i].year_published = -1;
134         }
135         i++;
136     }
137     fclose(infile);//closing file
138     return (1);
139 }
140
141 void print_book(book_entered_array[], int enter){
```

```

142
143     int i;
144
145     for(i = 0; i < enter; i++){//loop that prints title , author, ISBN, page, and
year published in that order
146         printf("Title: %s\n", entered_array[i].title);//title print
147         printf("Author: %s\n", entered_array[i].author_name);//author print
148         printf("ISBN: %s\n", entered_array[i].ISBN);//ISBN print
149
150         if(entered_array[i].pages == -1){
151             printf("Pages: N/A\n");//used in case pages are N/A
152         }
153         else{
154             printf("Pages: %d\n", entered_array[i].pages);//prints pages
155             fflush(stdout);
156         }
157         if(entered_array[i].year_published == -1){
158             printf("Published: N/A\n");//used if year published is N/A
159         }
160         else{
161             printf("Published: %d\n\n", entered_array[i].year_published);//prints
year published
162         }
163     }
164     return (0);
165
166 }
167
168 void search_title(book my_book_array[]) {
169
170     char cha1[50], cha2[50], cha3[50];
171     int i, j = 0, num = 360;
172     char* posis = NULL; //pointer in case no title is found
173     book my_entered[512];
174
175     printf("Enter Title criteria:\n");//user prompt
176
177
178     fgets(cha1, 50, stdin);
179     fflush(stdin);
180     cha1[strlen(cha1)-1] = '\0';//fgets for the users search input
181
182     strcpy(cha2, cha1);//copying the entered characters and changing the first to an
uppercase character
183     cha2[0] = toupper(cha2[0]);
184
185     strcpy(cha3, cha1);//copying again to make all of the characters uppercase
186

```

```

187     for(i = 0; i < strlen(cha1); i++){
188         cha3[i] = toupper(cha3[i]);
189     }
190
191     for(i = 0; i < num; i++){
192
193         posis = (strstr(my_book_array[i].title , cha1));
194
195         if(posis != NULL){//if statement when not NULL there is a match
196             strcpy(my_entered[j].title , my_book_array[i].title);//copying one array
to the others
197             strcpy(my_entered[j].author_name, my_book_array[i].author_name);
198             strcpy(my_entered[j].ISBN, my_book_array[i].ISBN);
199             my_entered[j].pages = my_book_array[i].pages;
200             my_entered[j].year_published = my_book_array[i].year_published;
201
202             j++;
203         }
204         posis = (strstr(my_book_array[i].title , cha2));
205         if(posis != NULL){//if statement when not NULL there is a match
206             strcpy(my_entered[j].title , my_book_array[i].title);//copying one array
to the others
207             strcpy(my_entered[j].author_name, my_book_array[i].author_name);
208             strcpy(my_entered[j].ISBN, my_book_array[i].ISBN);
209             my_entered[j].pages = my_book_array[i].pages;
210             my_entered[j].year_published = my_book_array[i].year_published;
211
212             j++;
213         }
214         posis = (strstr(my_book_array[i].title , cha3));
215         if(posis != NULL){//if statement when not NULL there is a match
216             strcpy(my_entered[j].title , my_book_array[i].title);//copying one array
to the others
217             strcpy(my_entered[j].author_name, my_book_array[i].author_name);
218             strcpy(my_entered[j].ISBN, my_book_array[i].ISBN);
219             my_entered[j].pages = my_book_array[i].pages;
220             my_entered[j].year_published = my_book_array[i].year_published;
221
222             j++;
223         }
224     }
225     if(strlen(my_entered[0].title) == 0){//check to see if character string exists
and the length if it does
226         printf("No Title found.\n");
227     }
228     print_book(my_entered, j);//passing needed array and int back to print book
229
230     return (0);

```

```

231
232 }
233
234 void search_author(book my_book_array[]) {
235
236     char cha1[50], cha2[50], cha3[50];
237     int i, j = 0, num = 360;
238     char* posis = NULL; //pointer in case no title is found
239     book my_entered[512];
240
241     printf("Enter Author criteria:\n"); //user prompt
242
243
244     fgets(cha1, 50, stdin); //fgets for user search input
245     fflush(stdin);
246     cha1[strlen(cha1)-1] = '\0';
247
248     strcpy(cha2, cha1); //copying the entered characters and changing the first to an
        uppercase character
249     cha2[0] = toupper(cha2[0]);
250
251     strcpy(cha3, cha1); //copying again to make all of the characters uppercase
252
253     for(i = 0; i < strlen(cha1); i++){
254         cha3[i] = toupper(cha3[i]);
255     }
256
257
258     for(i = 0; i < num; i++){
259
260         posis = strstr(my_book_array[i].author_name, cha1);
261
262         if(posis != NULL){ //if statement when not NULL there is a match
263             strcpy(my_entered[j].title, my_book_array[i].title); //copying one array
to the others
264             strcpy(my_entered[j].author_name, my_book_array[i].author_name);
265             strcpy(my_entered[j].ISBN, my_book_array[i].ISBN);
266             my_entered[j].pages = my_book_array[i].pages;
267             my_entered[j].year_published = my_book_array[i].year_published;
268
269             j++;
270         }
271         posis = (strstr(my_book_array[i].author_name, cha2));
272         if(posis != NULL){ //if statement when not NULL there is a match
273             strcpy(my_entered[j].title, my_book_array[i].title); //copying one array
to the others
274             strcpy(my_entered[j].author_name, my_book_array[i].author_name);
275             strcpy(my_entered[j].ISBN, my_book_array[i].ISBN);

```

```

276         my_entered[j].pages = my_book_array[i].pages;
277         my_entered[j].year_published = my_book_array[i].year_published;
278
279         j++;
280     }
281     posis = (strstr(my_book_array[i].author_name, cha3));
282     if(posis != NULL){//if statement when not NULL there is a match
283         strcpy(my_entered[j].title, my_book_array[i].title);//copying one array
to the others
284         strcpy(my_entered[j].author_name, my_book_array[i].author_name);
285         strcpy(my_entered[j].ISBN, my_book_array[i].ISBN);
286         my_entered[j].pages = my_book_array[i].pages;
287         my_entered[j].year_published = my_book_array[i].year_published;
288
289         j++;
290     }
291 }
292
293     if(strlen(my_entered[0].title) == 0){//check to see if character string exists
and the length if it does
294         printf("No Author found.\n");
295     }
296     print_book(my_entered, j);//passing array and int to print book
297
298     return (0);
299
300
301 }
302
303 void search_ISBN(book my_book_array[]) {
304
305     book my_entered[512];
306     int i, j = 0, num = 360;
307     char* posis = NULL;//pointer in case no title is found
308     char cha1[50];
309
310     printf("Enter ISBN criteria:\n");//user prompt
311
312     fgets(cha1, 50, stdin);//fgets the users search input
313     fflush(stdin);
314     cha1[strlen(cha1)-1] = '\0';
315
316     for(i = 0; i < num; i++){//going throughout the array to locate users search
input
317         posis = strstr(my_book_array[i].ISBN, cha1);
318
319         if(posis != NULL){//match found again as long as not NULL
320             strcpy(my_entered[j].title, my_book_array[i].title);//copying one array

```

```
    to the others
321         strcpy(my_entered[j].author_name, my_book_array[i].author_name);
322         strcpy(my_entered[j].ISBN, my_book_array[i].ISBN);
323         my_entered[j].pages = my_book_array[i].pages;
324         my_entered[j].year_published = my_book_array[i].year_published;
325
326         j++;
327     }
328 }
329 if(strlen(my_entered[0].title) == 0){//check to see if character string exists
and the length if it does
330     printf("ISBN not found.\n");
331 }
332 print_book(my_entered, j);//passing array and int to print book
333
334 return (0);
335 }
```