

Library Management System

ÖMER CENGİZ

University of Beira Interior- Erasmus Student

a47124@ubi.pt

16260056@firat.edu.tr

Portugal, Covilha

Abstract- Linked lists are very important in all programming languages. The most important feature that distinguishes Linked lists from arrays is that it has a dynamic structure. In addition, it makes the automation realistic by using the file exportable method. In such automation systems, it is also the file data structure.

Keywords: Linked List, Library management, File operations, Data structures, in the construction of Linked lists.

1. INTRODUCTION

While doing the library management application, I aimed to check the data input and output over Linked lists, to update and to keep this data in a regular file. As you know, the most important feature expected from automation such as library automation is its dynamic. The best data structure that can make the data of such a system accessible and dynamic is Linked Lists.

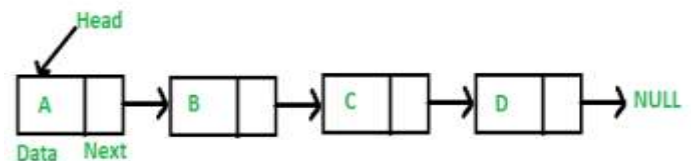
There are 2 menus in the library management application. The first menu consists of the main methods of the system that aim to make the system general and useful. From this menu, you can enter as many books as you prefer (Enter), add a new book to a created list (Insert), you can simply access the books in the system with the title of the book (Query by title), you can become the key to the books contained in the system via the ISBN code (Query by ISBN), you can delete the generated linked list by entering the ISBN code of a book you want, you can see all the data in the last Linked list (output), it allows you to save your

data and transfer your data to your linked list (Extract data from file) allows you to query with ISBN code and update on a book, and the last method allows you to save all the data in your linked list in your file (Save to File).

2. Simple Linked Lists

Arrays can be used to store similar types of linear data, but arrays have the following limitations.

- The size of the arrays is fixed: So we must know the upper limit on the number of elements in advance. Also, generally, the allocated memory is equal to the upper limit irrespective of the usage.
- Inserting a new element in an array of elements is expensive because the room has to be created for the new elements and to create room existing elements have to be shifted.



Advantages over arrays:

1. Dynamic size
2. Ease of insertion/deletion

Drawbacks:

1. Random access is not allowed. We have to access elements sequentially starting from the first node. So we cannot do binary search with linked

lists efficiently with its default implementation.

2. Extra memory space for a pointer is required with each element of the list.
3. Not cache friendly. Since array elements are contiguous locations, there is locality of reference which is not there in case of linked lists.

Representation:

A linked list is represented by a pointer to the first node of the linked list. The first node is called the head. If the linked list is empty, then the value of the head is NULL.

Each node in a list consists of at least two parts:

1. Data
2. Pointer (Or Reference) to the next node

In C, we can represent a node using structures. Below is an example of a linked list node with integer data.

3.File operations


Until now, operations using the C program have been done in a command prompt / terminal that is not stored anywhere. However, in the software industry, most programs are written to store information retrieved from the program. One such way is to store the received information in a file. The different operations that can be performed on a file are:

1. Creation of a new file (opened with "a" or "a +" or "w" or "w ++")
2. Open an existing file (fopen)
3. Reading from file (fscanf or fgets)
4. Writing to a file (fprintf or fputs)
5. Going to a specific location in a file (fseek, rewind)
6. Close a file (fclose)

3.1. Library management system file data type

The name of the is "program.txt". There are 9 different pieces of information about a book, in order of keeping in the file: ISBN, title, language, first author, second author, publisher company, year of publication, scientific area, price.

4. Library management application

The programming language used in this library system is the C language. 7 variables are used to define a book, namely ISBN (key), title, language, publication date, publisher company, scientific area and price. Since the data structure is a Linked list, these book variables are hidden in the nodes in the linked list. Now we will examine methods and class structures more closely to understand the automation algorithm. The values are all reserved with ';'.


4.1 book * getData ()

This method allows us to import the values from the "program.txt" file into a linked list. The data in the "program.txt" file is ";" Since it is separated by symbol, this method also uses the values ";" I did the cutting process by reading up to. In this method, the data is transferred to the relevant nodes in the linked list by using the strcncpy () method.

```

book*getData(){
    FILE *fp2;
    book*h, *tail, *p;
    h=tail=(book*)malloc(sizeof(book));
    h->next = NULL;
    int n = findline();
    if ( (fp2 = fopen("program.txt","r"))==NULL) {
        printf("The file could not be opened!!!");
        exit(1);
    }
    for (int i = 0; i < n; i++)
    {
        p = (book*)malloc(sizeof(book));
        char line_data[1024];
        fgets(line_data, 1024, fp2);
        char* token = strtok(line_data, ",");
        char* token1 = strtok(NULL, ",");
        char* token2 = strtok(NULL, ",");
        char* token3 = strtok(NULL, ",");
        char* token4 = strtok(NULL, ",");
        char* token5 = strtok(NULL, ",");
        char* token6 = strtok(NULL, ",");
        char* token7 = strtok(NULL, ",");
        char* token8 = strtok(NULL, ",");
        strcpy(&p->date.name, token);
        strcpy(&p->date.ISBN, token1);
        strcpy(&p->date.language, token2);
        strcpy(&p->date.firstAuthor, token3);
        strcpy(&p->date.secondAuthor, token4);
        strcpy(&p->date.publisherCompany, token5);
        strcpy(&p->date.yearPublication, token6);
        strcpy(&p->date.scientificArea, token7);
        strcpy(&p->date.price, token8);
        p->next = NULL;
        tail->next = p;
        tail = p;
    }
    fclose(fp2);
    return h;
}

```

Equivalent in the main menu:

```

----- Library Management System -----
QMER CENGIZ 47124 UBI
***** 1. Enter (New list) *****
***** 2. Insert *****
***** 3. Query by title *****
***** 4. Query by ISBN *****
***** 5. Delete *****
***** 6. Output *****
***** 7. Extract data from file *****
***** 8. Update *****
***** 9. Operations *****
***** 10. Save to File *****
Note: Only enter once, otherwise it will overwrite
please choose:

```

4.2.book*newFile(book*h)

This method creates an empty list every time it is run. It is a method where the user can add as many books as they want. The head node of the linked list is added and this head node is advanced one step after each insertion.

```

book*newFile(){
    book*h, *tail, *p;
    h=tail=(book*)malloc(sizeof(book));
    h->next = NULL;int n = 0;unsigned int count=1;
    printf("Enter the number of books to be entered:");
    scanf("%d", &n);
    for (int i = 0; i < n; i++){
        p = (book*)malloc(sizeof(book));
        printf("\n\t\t\tBook Count = %d\n\n",count);
        printf("Book ISBN = ");
        fgets(&p->date.ISBN, 256, stdin);
        scanf("%10[0-9a-zA-Z ]",&p->date.ISBN);
        printf("Book name = ");
        fgets(&p->date.name, 256, stdin);
        scanf("%10[0-9a-zA-Z ]",&p->date.name);
        printf("Book language = ");
        fgets(&p->date.language,256, stdin);
        scanf("%10[0-9a-zA-Z ]",&p->date.language);
        printf("Book first author name = ");
        fgets(&p->date.firstAuthor,256, stdin);
        scanf("%10[0-9a-zA-Z ]",&p->date.firstAuthor);
        printf("Book second author name = ");
        fgets(&p->date.secondAuthor, 256, stdin);
        scanf("%10[0-9a-zA-Z ]",&p->date.secondAuthor);
        printf("Book publisher company = ");
        fgets(&p->date.publisherCompany,256, stdin);
        scanf("%10[0-9a-zA-Z ]",&p->date.publisherCompany);
        printf("Book year of publication ");
        fgets(&p->date.yearPublication, 256, stdin);
        scanf("%10[0-9a-zA-Z ]",&p->date.yearPublication);
        printf("Book scientific area = ");
        fgets(&p->date.scientificArea, 256, stdin);
        scanf("%10[0-9a-zA-Z ]",&p->date.scientificArea);
        printf("Book price = ");
        fgets(&p->date.price, 256, stdin);
        scanf("%10[0-9a-zA-Z ]",&p->date.price);
        count++;
        p->next = NULL;
        tail->next = p;
        tail = p;
    }
    return h;
}

```

Equivalent in the main menu:

```

----- Library Management System -----
QMER CENGIZ 47124 UBI
***** 1. Enter (New list) *****
***** 2. Insert *****
***** 3. Query by title *****
***** 4. Query by ISBN *****
***** 5. Delete *****
***** 6. Output *****
***** 7. Extract data from file *****
***** 8. Update *****
***** 9. Operations *****
***** 10. Save to File *****
Note: Only enter once, otherwise it will overwrite
please choose:

```

4.3.insertBook ()

This method allows adding a previously created linked list book. I used fgets () and scanf () methods to get values containing spaces in this method and the methods that I get values from the user.

[illegible]

Equivalent in the main menu:

```

----- Library management System -----
OPR# CEMWII 47124 UBI
***** 1. Enter (New list) *****
***** 2. INSERT *****
***** 3. Query by title *****
***** 4. Query by ISBN *****
***** 5. Delete *****
***** 6. Output *****
***** 7. Extract data from file *****
***** 8. Update *****
***** 9. Operations *****
***** 10. Save to File *****
Note: Only enter once, otherwise it will overwrite
please choose:

```

4.4. Update ()

When updating, I can search the list named "strcmp (p-> date.ISBN, ISBN)" date and reach the relevant value via the ISBN key.

[illegible]

Equivalent in the main menu:

```

..... Library Management System .....
GMS C:\MSI\47124>lib
1. Enter New list .....
2. Enter ISBN .....
3. Query by title .....
4. Query by ISBN .....
5. Delete .....
6. Output .....
7. Extract data from File .....
8. UPDATE .....
9. Operations .....
10. Save to File .....

Note: Only enter once, otherwise it will overwrite
Please choose:

```


4.5. saveFile ()

One of the file operations, saveFile () saves the linked list created by the user to the file (program.txt) we store our data.

```
void saveFile(book*)
{
    book* p;
    FILE *fp2;
    unsigned int count=1;

    if ( (fp2=fopen("program.txt","a"))==NULL) {
        printf("The file is not opened!!!");
        exit(1);
    }

    for (p = h->next; p != NULL; p = p->next)
    {
        printf("\n\t\t\tBook Count = %d\n\n",count);
        printf("\t\t\tBook ISBN = %s\n",p->date.ISBN);
        printf("\t\t\tBook Name = %s\n",p->date.name);
        printf("\t\t\tBook language = %s\n",p->date.language);
        printf("\t\t\tBook first author name = %s\n",p->date.firstAuthor);
        printf("\t\t\tBook second author name = %s\n",p->date.secondAuthor);
        printf("\t\t\tBook publisher company = %s\n",p->date.publisherCompany);
        printf("\t\t\tBook year of publication = %s\n",p->date.yearPublication);
        printf("\t\t\tBook scientific area = %s\n",p->date.scientificArea);
        printf("\t\t\tBook price = %s\n",p->date.price);
        count++;
    }

    printf("\n\t\t\t*****THE DATA HAS BEEN EXPORTED TO THE FILE*****\n\n");
    fclose(fp2);
}
```

Equivalent in the main menu:

```
Library Management System
=====
OPER CEMFIZ 47124 UMI
=====
1. Enter (New List)
2. INSERT
3. Query by title
4. Query by ISBN
5. Delete
6. Output
7. Extract data from file
8. UPDATE
9. Operations
10. Save to File
Note: Only enter once, otherwise it will overwrite
please choose:
```

4.6. removeBook ()

In this method, the value that you want to delete from the linked list is found over the unique ISBN value. The deletion process is a little different. Here, the "tail" and "p" of the list are advanced together. When the value is found, the next node information is transferred to the tail after the node is deleted with the free (p) method.

```
void removeBook(book*h)
{
    char ISBN[50];
    book* p = h->next;
    book* tail = h;
    printf("Enter the ISBN of the book to be delete :");
    fgets(&ISBN,256, stdin);
    scanf("%10[0-9a-zA-Z ]",&ISBN);

    while (p != NULL)
    {
        if (strcmp(p->date.ISBN, ISBN)!=0)
        {
            p = p->next;
            tail = tail->next;
        }
        else
        {
            tail->next = p->next;
            free(p);
            return;
        }
    }
}
```

Equivalent in the main menu:

```
Library Management System
=====
OPER CEMFIZ 47124 UMI
=====
1. Enter (New List)
2. INSERT
3. Query by title
4. Query by ISBN
5. Delete
6. Output
7. Extract data from file
8. UPDATE
9. Operations
10. Save to File
Note: Only enter once, otherwise it will overwrite
please choose:
```

4.7. printOutput ()

It is made for the user to access all the books in the list whenever they want. It writes on the screen by browsing the nodes of the list one by one.

```
void printOutput(book*h)// output
{
    book* p;
    FILE *fp2;
    unsigned int count=1;
    p = h->next;
    if ( (fp2=fopen("program.txt","a"))==NULL) {
        printf("The file is not opened!!!");
        exit(1);
    }

    for (p = h->next; p != NULL; p = p->next)
    {
        printf("\n\t\t\tBook Count = %d\n\n",count);
        printf("\t\t\tBook ISBN = %s\n",p->date.ISBN);
        printf("\t\t\tBook name = %s\n",p->date.name);
        printf("\t\t\tBook language = %s\n",p->date.language);
        printf("\t\t\tBook first author name = %s\n",p->date.firstAuthor);
        printf("\t\t\tBook second author name = %s\n",p->date.secondAuthor);
        printf("\t\t\tBook publisher company = %s\n",p->date.publisherCompany);
        printf("\t\t\tBook year of publication = %s\n",p->date.yearPublication);
        printf("\t\t\tBook scientific area = %s\n",p->date.scientificArea);
        printf("\t\t\tBook price = %s\n",p->date.price);
        count++;
    }

    #close(fp2);
}
```

Equivalent in the main menu:

```
Library Management System
=====
OPER CEMFIZ 47124 UMI
=====
1. Enter (New List)
2. INSERT
3. Query by title
4. Query by ISBN
5. Delete
6. Output
7. Extract data from file
8. UPDATE
9. Operations
10. Save to File
Note: Only enter once, otherwise it will overwrite
please choose:
```

4.8.Operations ()

I wanted to divide the menu of the project into two with the operation method. Thus, I avoided confusion. At the same time, I created an Operations menu that the user can access at any time. Under this menu, the user can use specific filters and the output will have a simpler display.

```

void Operations(book*h)
{
    int i; int a = 1;
    while (a>0)
    {
        menu2();
        printf("please choose:");
        scanf("%d",&i);
        switch (i)
        {
            case 1:
                titleOfPrice(h);
                break;
            case 2:
                yearOfPublicationOfAllBook(h);
                break;
            case 3:
                scientificAreaLatestBook(h);
                break;
            case 4:
                firstAndSecondAuthorOfAllBook(h);
                break;
            case 5:
                scientificAreaLatestOfAllBook(h);
                break;
            case 6:
                firstAndSecondAuthorOfLastBook(h);
                break;
            case 7:
                scientificAreaAndFirstAuthorOfAllBook(h);
                break;
            case 8:
                languageOfAllBook(h);
                break;
            case 9:
                choose(h);
                break;
            default:
                printf("Invalid command! \n");
                a = -1;
                break;
        }
    }
}

```

Equivalent in the main menu:

```

----- Library Management System -----
----- OMEN CINGUIR 47124 GUI -----
***** 1. Enter (New List) *****
***** 2. INSERT *****
***** 3. Query by title *****
***** 4. Query by ISBN *****
***** 5. Delete *****
***** 6. Output *****
***** 7. Extract data from file *****
***** 8. UPDATE *****
***** 9. Operations *****
***** 10. Save to file *****
Note: Only enter once, otherwise it will overwrite
please choose:

```

```

----- Library Management System -----
***** 1. ----- FILTER ----- *****
***** 1. Title -----> Opida *****
***** 2. Year Of Publication -----> All Book *****
***** 3. Scientific Area -----> Recent Book *****
***** 4. First And Second Author -----> All Book *****
***** 5. ScientificArea -----> All Book *****
***** 6. First And Second Author -----> Recent Book *****
***** 7. Scientific Area And first Author -----> All Book *****
***** 8. Language -----> All Book *****
***** 9. BACK *****
Note: Only enter once, otherwise it will overwrite
please choose:

```

4.9.titleOfPrice ()

It queries the price over the title of the book and only writes the ISBN and Price variables.

```

void titleOfPrice(book*h)// price search
{
    char name[50];
    int control =0;
    unsigned int count=1;
    book*p = h->next;
    printf("Enter the title to find:");
    fgets(&name,256, stdin);
    scanf("%10[0-9a-zA-Z ]",&name);

    while (p != NULL)
    {
        if (strcmp(p->date.name, name)==0)
        {
            printf("\n\t\t\tBook Count = %d\n\n",count);
            printf("\t\t\tBook ISBN = %s\n",p->date.ISBN);
            printf("\t\t\tBook price (EURO) = %s\n",p->date.price);
            count++;
            control++;
        }
        p = p->next;
    }

    if (p == NULL&&control==0)
    {
        printf("Not found! \n");
        return;
    }
}

```

Equivalent in the main menu:

```

Enter the title to find:Ev

Book Count = 1

Book ISBN = 256488523
Book price (EURO) = 1,02

```

4.10.yearOfPublicationOfAllBook(book*h)

In this function, it searches among all books according to the publication year of the book.

```

void yearOfPublicationOfAllBook(book*h)
{
    char yearPublication[50];
    int control =0;
    unsigned int count=1;
    book*p = h->next;
    printf("Enter the year of publication to find:");
    fgets(&yearPublication,256, stdin);
    scanf("%10[0-9a-zA-Z ]",&yearPublication);

    while (p != NULL)
    {
        if (strcmp(p->date.yearPublication, yearPublication)==0)
        {
            printf("\n\t\t\tBook Count = %d\n\n",count);
            printf("\t\t\tBook ISBN = %s\n",p->date.ISBN);
            printf("\t\t\tBook name = %s\n",p->date.name);
            printf("\t\t\tBook language = %s\n",p->date.language);
            printf("\t\t\tBook first author name = %s\n",p->date.firstAuthor);
            printf("\t\t\tBook second author name = %s\n",p->date.secondAuthor);
            printf("\t\t\tBook publisher company = %s\n",p->date.publisherCompany);
            printf("\t\t\tBook year of publication = %s\n",p->date.yearPublication);
            printf("\t\t\tBook scientific area = %s\n",p->date.scientificArea);
            printf("\t\t\tBook price (EURO)= %s\n",p->date.price);
            count++;
            control++;
        }
        p = p->next;
    }

    if (p == NULL&&control==0)
    {
        printf("Not found! \n");
    }
}

```

Equivalent in the main menu:

```
----- Library Management System -----
----- 1. ----- FILTER ----- QUERY -----
----- 2. Year Of Publication -----> All Book -----
----- 3. Scientific Area -----> Recent Book -----
----- 4. First And Second Author -----> All Book -----
----- 5. ScientificArea -----> AllBook -----
----- 6. First And Second Author -----> Recent Book -----
----- 7. Scientific Area And First Author -> All Book -----
----- 8. Language -----> All Book -----
----- 9. BACK -----
Note: Only enter once, otherwise it will overwrite
please choose:1
Enter the year of publication to find:2021

Book Count = 1

Book ISBN = 9890C7465X
Book name = Algoritmos Em C#
Book language = Portuguese
Book first author name = Yang hu
Book second author name = Harper Lee
Book publisher company = Kindle
Book year of publication = 2021
Book scientific area = Programming
Book price (EUR0) = 4.80

Book Count = 2

Book ISBN = 989WIZVC6
Book name = C# Unity 3d
Book language = Portuguese
Book first author name = Alexander Ar.
Book second author name = Harper Lee
Book publisher company = Kindle
Book year of publication = 2021
Book scientific area = Programming
Book price (EUR0) = 7.80
```

Equivalent in the main menu:

```
----- Library Management System -----
----- 1. ----- FILTER ----- QUERY -----
----- 2. Year Of Publication -----> All Book -----
----- 3. Scientific Area -----> Recent Book -----
----- 4. First And Second Author -----> All Book -----
----- 5. ScientificArea -----> AllBook -----
----- 6. First And Second Author -----> Recent Book -----
----- 7. Scientific Area And First Author -> All Book -----
----- 8. Language -----> All Book -----
----- 9. BACK -----
Note: Only enter once, otherwise it will overwrite
please choose:3
Enter the scientific area to find:FICTION

Book Count = 1

Book ISBN = 95616S1162
Book name = A Pockingbird
Book language = English
Book first author name = Andrew Ay.
Book second author name = Harper Lee
Book publisher company = Fast
Book year of publication = 1998
Book scientific area = FICTION
Book price (EUR0) = 3.45
```

4.12.firstAndSecondAuthorOfAllBook(book*h)

Returns all books after filtering according to the first and second author names entered.

4.11.scientificAreaLatestBook(book*h)

If you type in scientific field information, it will give you the last book information added to the list according to the entry.

```
void scientificAreaLatestBook(Book*b)
{
    char scientificArea[50];
    int control = 0;
    unsigned int count = 1;
    book*p = b->next;
    printf("Enter the scientific area to find:");
    fgets(scientificArea, 50, stdin);
    scanf("%100s", &scientificArea);
    while (p != NULL)
    {
        if (strcmp(p->data.scientificArea, scientificArea) == 0)
        {
            p = p->next;
        }
        else
        {
            printf("\n\t\t\tBook Count = %d\n", count);
            printf("\n\t\t\tBook ISBN = %s\n", p->data.ISBN);
            printf("\n\t\t\tBook name = %s\n", p->data.name);
            printf("\n\t\t\tBook language = %s\n", p->data.language);
            printf("\n\t\t\tBook first author name = %s\n", p->data.firstAuthor);
            printf("\n\t\t\tBook second author name = %s\n", p->data.secondAuthor);
            printf("\n\t\t\tBook publisher company = %s\n", p->data.publisherCompany);
            printf("\n\t\t\tBook year of publication = %s\n", p->data.yearPublication);
            printf("\n\t\t\tBook scientific area = %s\n", p->data.scientificArea);
            printf("\n\t\t\tBook price (EUR0) = %s\n", p->data.price);
            count++;
            return;
        }
    }
    if (p == NULL)
    {
        printf("Not found!\n");
    }
}
```

```
void firstAndSecondAuthorOfAllBook(book*b)
{
    char firstAuthor[50];
    char secondAuthor[50];
    int control = 0;
    unsigned int count = 1;
    book*p = b->next;
    printf("Enter the first author to find:");
    fgets(firstAuthor, 50, stdin);
    scanf("%100s", &firstAuthor);
    printf("Enter the second author to find:");
    fgets(secondAuthor, 50, stdin);
    scanf("%100s", &secondAuthor);
    while (p != NULL)
    {
        if (strcmp(p->data.firstAuthor, firstAuthor) == 0 && strcmp(p->data.secondAuthor, secondAuthor) == 0)
        {
            printf("\n\t\t\tBook Count = %d\n", count);
            printf("\n\t\t\tBook ISBN = %s\n", p->data.ISBN);
            printf("\n\t\t\tBook name = %s\n", p->data.name);
            printf("\n\t\t\tBook language = %s\n", p->data.language);
            printf("\n\t\t\tBook first author name = %s\n", p->data.firstAuthor);
            printf("\n\t\t\tBook second author name = %s\n", p->data.secondAuthor);
            printf("\n\t\t\tBook publisher company = %s\n", p->data.publisherCompany);
            printf("\n\t\t\tBook year of publication = %s\n", p->data.yearPublication);
            printf("\n\t\t\tBook scientific area = %s\n", p->data.scientificArea);
            printf("\n\t\t\tBook price (EUR0) = %s\n", p->data.price);
            count++;
            control++;
        }
        p = p->next;
    }
    if (p == NULL && control == 0)
    {
        printf("Not found!\n");
    }
}
```

Equivalent in the main menu:

```
----- Library Management System -----
----- 1. ----- FILTER ----- QUERY -----
----- 2. Year Of Publication -----> All Book -----
----- 3. Scientific Area -----> Recent Book -----
----- 4. First And Second Author -----> All Book -----
----- 5. ScientificArea -----> AllBook -----
----- 6. First And Second Author -----> Recent Book -----
----- 7. Scientific Area And First Author -> All Book -----
----- 8. Language -----> All Book -----
----- 9. BACK -----
Note: Only enter once, otherwise it will overwrite
please choose:4
Enter the first author to find:Cem ULIN
Enter the second author to find:Ayşe DOĞAN

Book Count = 1

Book ISBN = 256498523
Book name = EV
Book language = Turkish
Book first author name = Cem ULIN
Book second author name = Ayşe DOĞAN
Book publisher company = papatya
Book year of publication = 1997
Book scientific area = Moon
Book price (EUR0) = psychology
```


The function that writes all books as a result of filtering according to the scientific field information entered

[illegible]

4.14.firstAndSecondAuthorOfLastBook(book*h)

```
void firstAndSecondAuthor(const Book& book) {
    char firstAuthor[50];
    char secondAuthor[50];
    assigned let count=1;
    book.p = book.t;
    print("Enter the first author to find:");
    fgets(firstAuthor, 50, stdin);
    scanf("%50s", &p-1), &firstAuthor);
    print("Enter the second author to find:");
    fgets(secondAuthor, 50, stdin);
    scanf("%50s", &p-1), &secondAuthor);
    while (p != NULL) {
        if (strcmp(p->data.firstAuthor, firstAuthor) != 0 && strcmp(p->data.secondAuthor, secondAuthor) != 0) {
            p = p->next;
        }
        else {
            printf("University Book Count = %d\n", count);
            printf("University ISBN = %s\n", p->data.ISBN);
            printf("University Book name = %s\n", p->data.name);
            printf("University Book language = %s\n", p->data.language);
            printf("University Book first author name = %s\n", p->data.firstAuthor);
            printf("University Book second author name = %s\n", p->data.secondAuthor);
            printf("University Book publisher company = %s\n", p->data.publisherCompany);
            printf("University Book year of publication = %s\n", p->data.yearPublication);
            printf("University Book scientific area = %s\n", p->data.scientificArea);
            printf("University Book price (USD) = %s\n", p->data.price);
            count++;
            return;
        }
    }
    if (p == NULL) {
        printf("Not found!\n");
    }
}
```

```

Library Management System
=====
1. ----- FILTER -----> Price ----- QUERY -----
2. Title -----> All Book -----
3. Year Of Publication -----> Recent Book -----
4. Scientific Area -----> All Book -----
5. First And Second Author -----> All Book -----
6. Scientific Area -----> Recent Book -----
7. First And Second Author -----> All Book -----
8. Scientific Area And First Author -----> All Book -----
9. language -----> All Book -----
10. BACK -----> -----

Note: Only enter once, otherwise it will overwrite
please choose:
Enter the first author to find: Cum UZUN
Enter the second author to find: Ayse DOGAN

Book Count = 1

Book ISBN = 256498523
Book name = FV
Book language = Turkish
Book first author name = Cum UZUN
Book second author name = Ayse DOGAN
Book publisher company = papatya
Book year of publication = 1997
Book scientific area = Moon
Book price (TLND) = psychology

```

```
void languageOfAllBooks(Book*& books)
{
    char language[50];
    int control = 0;
    unsigned int count = 1;
    Book* p = &bnext;
    printf("Enter the language to find:");
    fgets(language, 50, stdin);
    scanf("%100s-%s-%s %s", &language);
    while (p != NULL)
    {
        if (strcmp(p->date.language, language) == 0)
        {
            printf("\n\t\t\tBook Count = %d\n", count);
            printf("\t\t\tBook ISBN = %s\n", p->date.ISBN);
            printf("\t\t\tBook name = %s\n", p->date.name);
            printf("\t\t\tBook Language = %s\n", p->date.language);
            printf("\t\t\tBook first author name = %s\n", p->date.firstauthor);
            printf("\t\t\tBook second author name = %s\n", p->date.secondauthor);
            printf("\t\t\tdecl printf(const char * __restrict, format, ...)", p);
            printf("\t\t\tBook scientific area = %s\n", p->date.scientificArea);
            printf("\t\t\tBook price ($USD) = %s\n", p->date.price);
            count++; control++;
        }
        p = p->nnext;
    }
    if (p == NULL && control == 0)
    {
        printf("Not found! \n");
    }
}
```

```

Library Management System
*****
1. FILTER -----> QUERY-----
2. Title -----> Price -----
3. Year of Publication -----> All Book -----
4. Scientific Area -----> Recent Book -----
5. First And Second Author -----> All Book -----
6. Scientific Area -----> AllBook -----
7. First And Second Author -----> Recent Book -----
8. Scientific Area And First Author -----> All Book -----
9. Language -----> All Book -----
10. BACK -----> -----
Note: Only enter the language to #ind:English
once, otherwise it will overwrite
please choose:0
Enter the language to #ind:English

Book Count = 1

Book ISBN = 0140286470
Book name = Zero
Book language = English
Book first author name = Charles Seife
Book second author name = Herman Melville
Book publisher company = Paperback
Book year of publication = 2000
Book scientific area = Mathematics
Book price (EURO) = 5.75

Book Count = 2

Book ISBN = 0375588325
Book name = Cosmos
Book language = English
Book first author name = Carl Sagan
Book second author name = Harper Lee
Book publisher company = Random House
Book year of publication = 1980
Book scientific area = Physics
Book price (EURO) = 7.00

```



```

Book Count = 7

Book ISBN = 0316341516
Book name = Mythology
Book language = English
Book first author name = Edith Hamilton
Book second author name = Edin Ersoy
Book publisher company = XXX,1989
Book year of publication = CLASSICS
Book scientific area = 4.85
Book price (EUR0) =

Book Count = 8

Book ISBN = 0452284241
Book name = Animal Farm
Book language = English
Book first author name = George Orwell
Book second author name = Carolina C.
Book publisher company = Sun
Book year of publication = 1908
Book scientific area = FICTION
Book price (EUR0) = 2.80

Book Count = 9

Book ISBN = 0073385492
Book name = American History
Book language = English
Book first author name = Brinkley
Book second author name = Catarina V.
Book publisher company = Sun
Book year of publication = 2001
Book scientific area = TEXTBOOK
Book price (EUR0) = 3.40

```

REFERENCES

[1]. <https://www.geeksforgeeks.org/linked-list-set-1-introduction/>, linked list |set 1,author: ashwani khemani/geekforgeeks, last updated : 16 Sep, 2020, Accessed :11 April 2021

[2]. <https://www.programmersought.com/article/86134506218/>, C language, Programmer Sought, last updated : 2018-2021,Accessed: 18 April 2021

[3]. <https://stackoverflow.com/questions/1247989/how-do-you-allow-spaces-to-be-entered-using-scanf>, About C programming syntax, Last updated : 20 Jul 2021, Accessed : 27 April 2021