Experiment 1-A

#include <iostream>

#include <string.h>

using namespace std;

class library

{

string title;

string author;

int book\_record;

int number\_of\_copies;

string subject;

public:

void getdata();

void display();

void searchtitle(library l[], int n);

void searchauthor(library l[], int n);

};

void library::getdata()

{

fflush(stdin);

cout << "\nEnter the title of the book: " << endl;

getline(cin, title);

fflush(stdin);

cout << "Enter the author of the book: " << endl;

getline(cin, author);

fflush(stdin);

cout << "Enter the subject of the book: " << endl;

getline(cin, subject);

fflush(stdin);

cout << "Enter the number of copies of the book: " << endl;

cin >> number\_of\_copies;

fflush(stdin);

}

void library::display()

{

cout << "\nTitle of the book: " << title << endl;

cout << "Author of the book: " << author << endl;

cout << "Number of copies of the book: " << number\_of\_copies << endl;

cout << "Subject of the book: " << subject << "\n\n";

}

void library::searchtitle(library l[], int n)

{

string t;

int flag = 0;

cout << "Enter the title of the book you want to search: " << endl;

fflush(stdin);

getline(cin, t);

for (int i = 0; i < n; i++)

{

if (l[i].title == t)

{

l[i].display();

flag = 1;

}

}

if (flag == 0)

{

cout << "Book not found" << endl;

}

}

void library::searchauthor(library l[], int n)

{

string a;

int flag = 0;

cout << "Enter the author of the book you want to search: " << endl;

fflush(stdin);

getline(cin, a);

for (int i = 0; i < n; i++)

{

if (l[i].author == a)

{

l[i].display();

flag = 1;

}

}

if (flag == 0)

{

cout << "Book not found" << endl;

}

}

int main()

{

int n, count = 0;

library l[100];

cout << "Enter the number of books you want to enter: " << endl;

cin >> n;

for (int i = 0; i < n; i++)

{

count++;

l[i].getdata();

}

int ch;

do

{

cout << "\nEnter your choice: " << endl;

cout << "1. Insert book record" << endl;

cout << "2. Search book record based on title" << endl;

cout << "3. Search book record based on author" << endl;

cout << "4. Display all the book records" << endl;

cout << "5. Exit" << endl;

cin >> ch;

switch (ch)

{

case 1:

l[count].getdata();

count++;

break;

case 2:

l[0].searchtitle(l, n);

break;

case 3:

l[0].searchauthor(l, n);

break;

case 4:

for (int i = 0; i < count; i++)

l[i].display();

break;

case 5:

break;

default:

cout << "Invalid choice" << endl;

}

} while (ch != 5);

return 0;

}

OUTPUT:

Enter the number of books you want to enter:

2

Enter the title of the book:

The Rough

Enter the author of the book:

Mary Jen

Enter the subject of the book:

Fiction

Enter the number of copies of the book:

23

Enter the title of the book:

Happens

Enter the author of the book:

Seth Roegan

Enter the subject of the book:

Thriller

Enter the number of copies of the book:

42

Enter your choice:

1. Insert book record

2. Search book record based on title

3. Search book record based on author

4. Display all the book records

5. Exit

4

Title of the book: The Rough

Author of the book: Mary Jen

Number of copies of the book: 23

Subject of the book: Fiction

Title of the book: Happens

Author of the book: Seth Roegan

Number of copies of the book: 42

Subject of the book: Thriller

Enter your choice:

1. Insert book record

2. Search book record based on title

3. Search book record based on author

4. Display all the book records

5. Exit

2

Enter the title of the book you want to search:

The Rough

Title of the book: The Rough

Author of the book: Mary Jen

Number of copies of the book: 23

Subject of the book: Fiction

Enter your choice:

1. Insert book record

2. Search book record based on title

3. Search book record based on author

4. Display all the book records

5. Exit

3

Enter the author of the book you want to search:

Seth Roegan

Title of the book: Happens

Author of the book: Seth Roegan

Number of copies of the book: 42

Subject of the book: Thriller

Enter your choice:

1. Insert book record

2. Search book record based on title

3. Search book record based on author

4. Display all the book records

5. Exit

5

Experiment 1-B

#include <iostream>

#include <string.h>

using namespace std;

class library

{

string title;

string author;

int book\_record;

int number\_of\_copies;

string subject;

public:

void getdata();

void display();

void searchtitle(library l[], int n);

void searchauthor(library l[], int n);

};

void library::getdata()

{

fflush(stdin);

cout << "\nEnter the title of the book: " << endl;

getline(cin, title);

fflush(stdin);

cout << "Enter the author of the book: " << endl;

getline(cin, author);

fflush(stdin);

cout << "Enter the subject of the book: " << endl;

getline(cin, subject);

fflush(stdin);

cout << "Enter the number of copies of the book: " << endl;

cin >> number\_of\_copies;

fflush(stdin);

}

void library::display()

{

cout << "\nTitle of the book: " << title << endl;

cout << "Author of the book: " << author << endl;

cout << "Number of copies of the book: " << number\_of\_copies << endl;

cout << "Subject of the book: " << subject << "\n\n";

}

void library::searchtitle(library l[], int n)

{

string t;

int flag = 0;

cout << "Enter the title of the book you want to search: " << endl;

fflush(stdin);

getline(cin, t);

for (int i = 0; i < n; i++)

{

if (l[i].title == t)

{

l[i].display();

flag = 1;

}

}

if (flag == 0)

{

cout << "Book not found" << endl;

}

}

void library::searchauthor(library l[], int n)

{

string a;

int flag = 0;

cout << "Enter the author of the book you want to search: " << endl;

fflush(stdin);

getline(cin, a);

for (int i = 0; i < n; i++)

{

if (l[i].author == a)

{

l[i].display();

flag = 1;

}

}

if (flag == 0)

{

cout << "Book not found" << endl;

}

}

int main()

{

int n, count = 0;

library \*b = new library[100];

cout << "Enter the number of books you want to enter: " << endl;

cin >> n;

for (int i = 0; i < n; i++)

{

count++;

b[i].getdata();

}

int ch;

do

{

cout << "\nEnter your choice: " << endl;

cout << "1. Insert book record" << endl;

cout << "2. Search book record based on title" << endl;

cout << "3. Search book record based on author" << endl;

cout << "4. Display all the book records" << endl;

cout << "5. Exit" << endl;

cin >> ch;

switch (ch)

{

case 1:

b[count].getdata();

count++;

break;

case 2:

b->searchtitle(b, count);

break;

case 3:

b->searchauthor(b, count);

break;

case 4:

for (int i = 0; i < count; i++)

{

b[i].display();

}

break;

case 5:

break;

default:

cout << "Invalid choice" << endl;

}

} while (ch != 5);

delete[] b;

return 0;

}

OUTPUT:

Enter the number of books you want to enter:

2

Enter the title of the book:

The Rough

Enter the author of the book:

Mary Jen

Enter the subject of the book:

Fiction

Enter the number of copies of the book:

23

Enter the title of the book:

Happens

Enter the author of the book:

Seth Roegan

Enter the subject of the book:

Thriller

Enter the number of copies of the book:

42

Enter your choice:

1. Insert book record

2. Search book record based on title

3. Search book record based on author

4. Display all the book records

5. Exit

4

Title of the book: The Rough

Author of the book: Mary Jen

Number of copies of the book: 23

Subject of the book: Fiction

Title of the book: Happens

Author of the book: Seth Roegan

Number of copies of the book: 42

Subject of the book: Thriller

Enter your choice:

1. Insert book record

2. Search book record based on title

3. Search book record based on author

4. Display all the book records

5. Exit

2

Enter the title of the book you want to search:

The Rough

Title of the book: The Rough

Author of the book: Mary Jen

Number of copies of the book: 23

Subject of the book: Fiction

Enter your choice:

1. Insert book record

2. Search book record based on title

3. Search book record based on author

4. Display all the book records

5. Exit

3

Enter the author of the book you want to search:

Seth Roegan

Title of the book: Happens

Author of the book: Seth Roegan

Number of copies of the book: 42

Subject of the book: Thriller

Enter your choice:

1. Insert book record

2. Search book record based on title

3. Search book record based on author

4. Display all the book records

5. Exit

5

Experiment 2-A

#include <iostream>

using namespace std;

void swap(int &a, int &b)

{

int temp;

temp = a;

a = b;

b = temp;

}

int main()

{

int x = 10, y = 20;

cout << "Before swapping: " << endl;

cout << "x = " << x << endl;

cout << "y = " << y << endl;

swap(x, y);

cout << "After swapping: " << endl;

cout << "x = " << x << endl;

cout << "y = " << y << endl;

return 0;

}

OUTPUT:

Before swapping:

x = 10

y = 20

After swapping:

x = 20

y = 10

Experiment 2-B

#include <iostream>

using namespace std;

class complex

{

int a, b;

public:

void setdata(int x, int y)

{

a = x;

b = y;

}

void showdata()

{

cout << "a = " << a ;

cout << " b = " << b << endl;

}

friend complex sum(complex, complex);

};

inline complex sum(complex o1, complex o2)

{

complex o3;

o3.setdata((o1.a + o2.a), (o1.b + o2.b));

return o3;

}

int main()

{

complex c1, c2, c3;

c1.setdata(3, 4);

c2.setdata(5, 6);

cout<<"c1 --> ";

c1.showdata();

cout<<"c2 --> ";

c2.showdata();

cout<<"c3 --> ";

c3 = sum(c1, c2);

c3.showdata();

return 0;

}

OUTPUT:

c1 --> a = 3 b = 4

c2 --> a = 5 b = 6

c3 --> a = 8 b = 10

Experiment 2-C

#include <iostream>

using namespace std;

void volume(int a)

{

cout << "Volume of cube is: " << a \* a \* a << endl;

}

void volume(int l, int b, int h)

{

cout << "Volume of cuboid is: " << l \* b \* h << endl;

}

void volume(float r)

{

cout << "Volume of sphere is: " << (4 / 3) \* 3.14 \* r \* r \* r << endl;

}

int main()

{

int a, l, b, h;

float r;

cout << "Enter the side of the cube: " << endl;

cin >> a;

volume(a);

cout << "Enter the length, breadth and height of the cuboid: " << endl;

cin >> l >> b >> h;

volume(l, b, h);

cout << "Enter the radius of the sphere: " << endl;

cin >> r;

volume(r);

return 0;

}

OUTPUT:

Enter the side of the cube:

4

Volume of cube is: 64

Enter the length, breadth and height of the cuboid:

4 5 2

Volume of cuboid is: 40

Enter the radius of the sphere:

2

Volume of sphere is: 25.12

Assignment 1

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

struct book\_record

{

int number;

char \*subject;

char \*title;

char \*author;

};

void inputbooks(struct book\_record \*book)

{

fflush(stdin);

printf("Enter serial number of book: ");

scanf("%d", &book->number);

getchar();

printf("Enter subject: ");

book->subject = (char \*)malloc(sizeof(char) \* 100);

fgets(book->subject, 100, stdin);

fflush(stdin);

printf("Enter title: ");

book->title = (char \*)malloc(sizeof(char) \* 100);

fgets(book->title, 100, stdin);

fflush(stdin);

printf("Enter author: ");

book->author = (char \*)malloc(sizeof(char) \* 100);

fgets(book->author, 100, stdin);

}

void printbooks(struct book\_record \*book)

{

printf("\nSerial number of book: %d\n", book->number);

printf("\nSubject: %s\n", book->subject);

printf("Title: %s\n", book->title);

printf("Author: %s\n", book->author);

}

int choicee()

{

int choice;

printf("\n1.Insert a book record\n2.Search book based on subject\n3.Search book based on author name\n4.Display all book records\n5.Exit");

printf("\n\nEnter your choice: ");

scanf("%d", &choice);

return choice;

}

void insertbooks(struct book\_record \*book)

{

inputbooks(book);

}

void searchbooks(struct book\_record \*book,int count)

{

char \*subject;

subject = (char \*)malloc(sizeof(char) \* 100);

printf("\nEnter subject of book to be searched: ");

fflush(stdin);

fgets(subject, 100, stdin);

fflush(stdin);

for (int i = 0; i < count + 1; i++)

{

if (strcmp(book[i].subject, subject) == 0)

{

printbooks(&book[i]);

}

}

}

void searchbooks1(struct book\_record \*book,int count)

{

char \*author;

author = (char \*)malloc(sizeof(char) \* 100);

printf("\nEnter author of book to be searched: ");

fflush(stdin);

fgets(author, 100, stdin);

fflush(stdin);

for (int i = 0; i < count + 1; i++)

{

if (strcmp(book[i].author, author) == 0)

{

printbooks(&book[i]);

}

}

}

void menu(struct book\_record \*book,int count)

{

int choice;

do

{

choice = choicee();

switch (choice)

{

case 1:

insertbooks(&book[count]);

count++;

break;

case 2:

searchbooks(book,count);

break;

case 3:

searchbooks1(book,count);

break;

case 4:

for (int i = 0; i < count + 1; i++)

{

printbooks(&book[i]);

}

break;

case 5:

break;

default:

printf("\nInvalid choice");

break;

}

} while (choice != 5);

}

int main()

{

int count = 0;

struct book\_record book[100];

menu(book,count);

return 0;

}

OUTPUT:

1.Insert a book record

2.Search book based on subject

3.Search book based on author name

4.Display all book records

5.Exit

Enter your choice: 1

Enter serial number of book: 324

Enter subject: Math

Enter title: Good Math

Enter author: Heer

1.Insert a book record

2.Search book based on subject

3.Search book based on author name

4.Display all book records

5.Exit

Enter your choice: 1

Enter serial number of book: 324

Enter subject: Meals

Enter title: Macdonald

Enter author: ee ya eeya yo

1.Insert a book record

2.Search book based on subject

3.Search book based on author name

4.Display all book records

5.Exit

Enter your choice: 4

Serial number of book: 324

Subject: Math

Title: Good Math

Author: Heer

Serial number of book: 324

Subject: Meals

Title: Macdonald

Author: ee ya eeya yo

1.Insert a book record

2.Search book based on subject

3.Search book based on author name

4.Display all book records

5.Exit

Enter your choice: 2

Enter subject of book to be searched: Meals

Serial number of book: 324

Subject: Meals

Title: Macdonald

Author: ee ya eeya yo

1.Insert a book record

2.Search book based on subject

3.Search book based on author name

4.Display all book records

5.Exit

Enter your choice: 3

Enter author of book to be searched: Heer

Serial number of book: 324

Subject: Math

Title: Good Math

Author: Heer

1.Insert a book record

2.Search book based on subject

3.Search book based on author name

4.Display all book records

5.Exit

5