```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

// Function prototype
void student(); //View Student Portfolio
void admin(); //View Admin Portfolio
void addbook(); // Adding Book Function
void returnbook(); // Function for Return Book
void issue(); // Funtion for Issued Book
void view(); //Funtion for see all book list
void registr(); // Register Student New
void search(); // Search Book
void vieweee();
void viewcse();
void datadelet(); // Delete Data
void record();
void alldata(); // View Individual Issued Book
void viewissue(); // Funtion for Issued Book
void viewallissue(); // View all Issed Book
struct ST{ // Student Registration Structure
    char stdname[50];
    int stdid;
};
struct ISS{ // Issued Book Structure
    int id;
    char name[50];
    char bk[50];
    char isdate[50];
};
struct SI{ //Add Book Structure

    int id;
    char book[100];
    char author[100];
    char dept[20];
    char self[10];
    char date[40];
};
void viewallissue() {
    system("cls");
    struct ISS s;  // Use the correct structure name here
    FILE *fPtr;

    fPtr = fopen("issue_bin.dat", "rb");
    if (fPtr == NULL) {
        printf("Error opening file for reading.\n");
        return;
    }

    while (fread(&s, sizeof(struct ISS), 1, fPtr) == 1) {
        if (s.id != 0) {  // Assuming '0' represents an empty record
            printf("\nID=%d, Name=%s, Book=%s, Date=%s\n", s.id, s.name, s.bk, s.isdate);
        }
    }

    fclose(fPtr);
}

void viewissue(){
    system("cls");
    int n;
    printf("\n  1| Show All Issued Book ");
    printf("\n  2| Show Individual Data ");
    printf("\n");
    scanf("%d",&n);
```

```c
    if(n==2) alldata();
    if(n==1) viewallissue();
}
int main() {
    struct ST newStudent;
    FILE *fle;
    fle = fopen("student_bin.dat", "rb");
    int n;
    char id[10];
    char pass[] = "weq123Admin"; //Admin Fixed Password
    char user[20];

    printf("||||||||||||||||||||||||| Welcome to RIN G1 Library |||||||||||||||||||||||||\n");
    printf("                            1) Admin\n");
    printf("                            2) Student\n");
    printf("                            3) Student Registration\n");
    scanf("%d", &n);

    if (n == 2) {
        printf("\nEnter Student ID :");
        scanf("%s", id);

        int found = 0;

        // Loop until the end of the file or until the Student is found
        while (fread(&newStudent, sizeof(struct ST), 1, fle) == 1) {
            // Check if the student ID matches
            if (newStudent.stdid == atoi(id)) {
                student();
                found = 1;
                break;
            }
        }

        fclose(fle);

        if (found == 0) {
            printf("Student not found!\n");
        }
    } else if (n == 1) {
        printf("Enter Password : ");
        scanf("%s", user);

        if (strcmp(user, pass) == 0) {
            admin();
        } else {
            printf("Incorrect Password !!!!!!");
        }
    } else if (n == 3) {
        registr();
    }

    return 0;
}
void student(){ //Student Portofolio Home page
    int dec;
    while(1){
    printf("\n");
    printf("  (1) View Book \n");
    printf("  (2) Issued Book \n");
    printf("  (3) Exist \n");
    scanf("%d",&dec);
    if(dec==1) view();
    else if(dec==2) record();
    else if(dec==3) break;
    }
```

```c

}
void registr() { //Resister New Struct Funtion
    struct ST new = {0, ""};
    FILE *newstd;

    newstd = fopen("student_bin.dat", "rb+");

    if (newstd == NULL) {
        newstd = fopen("student_bin.dat", "wb+");
        for (int i = 0; i < 1000; i++) { // Corrected the comment
            fwrite(&new, sizeof(struct ST), 1, newstd);
        }
    }

    printf("\nEnter Student ID : ");
    scanf("%d", &new.stdid);
    printf("\nEnter Student Name : ");
    scanf("%s", new.stdname);

    fseek(newstd, (new.stdid % 1000 - 1) * sizeof(struct ST), SEEK_SET);
    fwrite(&new, sizeof(struct ST), 1, newstd);
    printf("\nRegistration Successful!\n");


    fclose(newstd);
}


void admin() { //Admin Home page Function
    int n;
 system("cls");
 //This code uses preprocessor directives to include different commands based on the operating system. Let's break it down://
    while(1){
        printf("\n");
    printf("  1 | Add Book\n");
    printf("  2 | View Book List\n");
    printf("  3 | Return Book\n");
    printf("  4 | Issue Book\n");
    printf("  5 | Search Book\n");
    printf("  6 | View record of issued books\n");
    printf("  7 | Delete record from Book List\n");
    printf("  8 | Exit\n");
    scanf("%d",&n);
    if(n==8) break;
    else if(n==1) addbook();
    else if(n==2) view();
    else if(n==3) returnbook();
    else if(n==4) issue();
    else if(n==7) datadelet();
    else if(n==6) viewissue();
    else if(n==5) search();
    }
}
void alldata(){ // View All Issued Books from File
 system("cls");
    int nm;
    struct ISS man;
    FILE *isst;

    isst = fopen("issue_bin.dat", "rb");
    if (isst == NULL) {
        printf("Error opening file for reading.\n");
        return;
    }
```

```c

    printf("Enter Student ID : ");
    scanf("%d", &nm);

    int found = 0;

    // Loop until the end of the file or until the book is found
    while (!feof(isst)) {
        fread(&man, sizeof(struct ISS), 1, isst);

        // Check if the book name matches
        if (man.id==nm) {
                printf("Book Found !!!!\n");
            printf(
"=============================================================================================================
===\n");
    printf("| %-10s | %-30s | %-20s | %-15s \n", "Student ID", "Student Name", "Book Name", "Issued Date");
    printf(
"=============================================================================================================
===\n");
    printf("| %-10d | %-30s | %-20s | %-15s \n",
                    man.id, man.name, man.bk, man.isdate);
                    found = 1;
            break;
        }

        }


    fclose(isst);

    if (found==0) {
        printf("Book not found!\n");
    }
}


void record() { //View Issued Book list by Individual
   system("cls");
    int nm;
    struct ISS man;
    FILE *isst;

    isst = fopen("issue_bin.dat", "rb");
    if (isst == NULL) {
        printf("Error opening file for reading.\n");
        return;
    }

    printf("Enter Student ID : ");
    scanf("%d", &nm);

    int found = 0;

    // Loop until the end of the file or until the book is found
    while (!feof(isst)) {
        fread(&man, sizeof(struct ISS), 1, isst);

        // Check if the book name matches
        if (man.id==nm) {
                printf("Book Found !!!!\n");
            printf(
"=============================================================================================================
===\n");
    printf("| %-10s | %-30s | %-20s | %-15s \n", "Student ID", "Student Name", "Book Name", "Issued Date");
    printf(
```

```
"=============================================================================================================
===\n");
258       printf("| %-10d | %-30s | %-20s | %-15s \n",
259                       man.id, man.name, man.bk, man.isdate);
260                       found = 1;
261               break;
262           }
263
264           }
265
266
267       fclose(isst);
268
269       if (found==0) {
270           printf("Book not found!\n");
271       }
272   }
273
274
275   void returnbook() { //Return Issued Book Function
276       system("cls");
277       struct ISS book;
278       FILE *dlt;
279       int n;
280
281       printf("Enter Student ID: ");
282       scanf("%d", &n);
283
284       dlt = fopen("issue_bin.dat", "rb+");
285       if (dlt == NULL) {
286           printf("Error opening file for reading.\n");
287           return;
288       }
289
290       int found = 0;
291
292       // Loop until the end of the file
293       while (fread(&book, sizeof(struct ISS), 1, dlt) == 1) {
294           // Check if the student ID matches
295           if (book.id == n) {
296               fseek(dlt, -sizeof(struct ISS), SEEK_CUR);
297               struct ISS emptyBook = {0, "", "", "", ""};
298               fwrite(&emptyBook, sizeof(struct ISS), 1, dlt);
299               found = 1;
300               break;
301           }
302       }
303
304       fclose(dlt);
305
306       if (found == 0) {
307           printf("Student not issued any book!\n");
308       } else {
309           printf("Return Successful!\n");
310       }
311   }
312
313   void search() { //Search Book from library- Function
314       system("cls");
315       char nm[100];
316       struct SI book;
317       FILE *src;
318
319       src = fopen("book_bin.dat", "rb");
320       if (src == NULL) {
321           printf("Error opening file for reading.\n");
```

```
322         return;
323     }
324 
325     printf("Enter Book Name : ");
326     scanf("%s", nm);
327 
328     int found = 0;
329 
330     // Loop until the end of the file or until the book is found
331     while (!feof(src)) {
332         fread(&book, sizeof(struct SI), 1, src);
333 
334         // Check if the book name matches
335         if (strcmp(book.book, nm) == 0) {
336                 printf("Book Found !!!!\n");
337              printf(
"=============================================================================================================
===\n");
338     printf("| %-10s | %-30s | %-20s | %-15s | %-10s | %-15s |\n", "Book ID", "Book Name", "Author",
"Department", "Self No", "Date");
339     printf(
"=============================================================================================================
===\n");
340     printf("| %-10d | %-30s | %-20s | %-15s | %-10s | %-15s |\n",
341                    book.id, book.book, book.author, book.dept, book.self, book.date);
342                    found = 1;
343             break;
344         }
345 
346         }
347 
348 
349     fclose(src);
350 
351     if (found==0) {
352         printf("Book not found!\n");
353     }
354 }
355 
356 
357 void view(){ //View Book List Funtion Main Page
358    system("cls");
359     int ans;
360     printf(" 1] Electrical and Electronic Engineering\n");
361     printf(" 2] Computer Science and Engineering\n");
362     printf(" 3] Bachelor of Business Administration\n");
363     printf(" 4] Civil Engineering\n");
364     printf(" 5] Pharmacy\n");
365     scanf("%d",&ans);
366     if(ans==1)vieweee();
367     else if(ans==2)viewcse();
368     else if(ans==3)viewbba();
369     else if(ans==4)viewce();
370     else if(ans==5)viewpharm();
371 }
372 
373 void addbook() { //Add Book Page Funtion
374     struct SI book = {0, "", "", ""};
375     FILE *fptr;
376 
377     if (fptr == NULL) {
378         fptr = fopen("book_bin.dat", "wb+");
379         //Assume Varsity has 5 department and 400 book can store each department//
380         for (int i = 0; i < 2000; i++) {
381             fwrite(&book, sizeof(struct SI), 1, fptr);
382         }
```

```
383      }
384      if(fptr != NULL){
385              system("cls");
386          fptr=fopen("book_bin.dat","rb+");
387          printf("\nEnter Book ID :");
388          scanf("%d",&book.id);
389          printf("\nEnter Book Name :");
390          scanf("%s",book.book);
391          printf("\nEnter Author Name :");
392          scanf("%s",book.author);
393          printf("\nEnter Department :");
394          scanf("%s",book.dept);
395          printf("\nEnter Self No :");
396          scanf("%s",book.self);
397          printf("\nEnter Date :");
398          scanf("%s",book.date);
399          fseek(fptr,(book.id-1)*sizeof(struct SI),SEEK_SET);
400          fwrite(&book,sizeof(struct SI),1,fptr);
401      }
402
403      fclose(fptr);
404  }
405  void vieweee() { //View EEE Book List from "view"  main funtion
406      system("cls");
407      FILE *viu;
408      struct SI book;
409      viu = fopen("book_bin.dat", "rb");
410
411      printf(
"=============================================================================================================\n");
412      printf("| %-10s | %-30s | %-20s | %-15s | %-10s | %-15s |\n", "Book ID", "Book Name", "Author",
"Department", "Self No", "Date");
413      printf(
"=============================================================================================================\n");
414
415      int startRecord = 0;  // Adjust the start record as needed
416      int endRecord = 399;    // Adjust the end record as needed
417
418      // Move the file pointer to the start record
419      fseek(viu, (startRecord - 1) * sizeof(struct SI), SEEK_SET);
420
421      // Read and display records within the specified range
422      for (int i = startRecord; i <= endRecord && fread(&book, sizeof(struct SI), 1, viu) == 1; i++) {
423          if (book.id != 0) {
424              printf("| %-10d | %-30s | %-20s | %-15s | %-10s | %-15s |\n",
425                     book.id, book.book, book.author, book.dept, book.self, book.date);
426          }
427      }
428
429      printf(
"=============================================================================================================\n");
430
431      fclose(viu);
432  }
433  void viewcse() { //View CSE Book List from "view"  main funtion
434      system("cls");
435      FILE *viu;
436      struct SI book;
437      viu = fopen("book_bin.dat", "rb");
438
439      printf(
"=============================================================================================================\n");
440      printf("| %-10s | %-30s | %-20s | %-15s | %-10s | %-15s |\n", "Book ID", "Book Name", "Author",
"Department", "Self No", "Date");
441      printf(
"=============================================================================================================\n");
```

```
442
443      int startRecord = 400;  // Adjust the start record as needed
444      int endRecord = 799;    // Adjust the end record as needed
445
446      // Move the file pointer to the start record
447      fseek(viu, (startRecord - 1) * sizeof(struct SI), SEEK_SET);
448
449      // Read and display records within the specified range
450      for (int i = startRecord; i <= endRecord && fread(&book, sizeof(struct SI), 1, viu) == 1; i++) {
451          if (book.id != 0) {
452              printf("| %-10d | %-30s | %-20s | %-15s | %-10s | %-15s |\n",
453                     book.id, book.book, book.author, book.dept, book.self, book.date);
454          }
455      }
456
457      printf(
"=============================================================================================================\n");
458
459      fclose(viu);
460  }
461  void viewpharm() { //View Pharmacy Book List from "view"  main funtion
462      system("cls");
463      FILE *viu;
464      struct SI book;
465      viu = fopen("book_bin.dat", "rb");
466
467      printf(
"=============================================================================================================\n");
468      printf("| %-10s | %-30s | %-20s | %-15s | %-10s | %-15s |\n", "Book ID", "Book Name", "Author",
"Department", "Self No", "Date");
469      printf(
"=============================================================================================================\n");
470
471      int startRecord = 1600;  // Adjust the start record as needed
472      int endRecord = 2000;    // Adjust the end record as needed
473
474      // Move the file pointer to the start record
475      fseek(viu, (startRecord - 1) * sizeof(struct SI), SEEK_SET);
476
477      // Read and display records within the specified range
478      for (int i = startRecord; i <= endRecord && fread(&book, sizeof(struct SI), 1, viu) == 1; i++) {
479          if (book.id != 0) {
480              printf("| %-10d | %-30s | %-20s | %-15s | %-10s | %-15s |\n",
481                     book.id, book.book, book.author, book.dept, book.self, book.date);
482          }
483      }
484
485      printf(
"=============================================================================================================\n");
486
487      fclose(viu);
488  }
489  void viewce() { //View Civil Engineering Book List from "view"  main funtion
490      system("cls");
491      FILE *viu;
492      struct SI book;
493      viu = fopen("book_bin.dat", "rb");
494
495      printf(
"=============================================================================================================\n");
496      printf("| %-10s | %-30s | %-20s | %-15s | %-10s | %-15s |\n", "Book ID", "Book Name", "Author",
"Department", "Self No", "Date");
497      printf(
"=============================================================================================================\n");
498
499      int startRecord = 1200;  // Adjust the start record as needed
```

```c
    int endRecord = 1599;    // Adjust the end record as needed

    // Move the file pointer to the start record
    fseek(viu, (startRecord - 1) * sizeof(struct SI), SEEK_SET);

    // Read and display records within the specified range
    for (int i = startRecord; i <= endRecord && fread(&book, sizeof(struct SI), 1, viu) == 1; i++) {
        if (book.id != 0) {
            printf("| %-10d | %-30s | %-20s | %-15s | %-10s | %-15s |\n",
                   book.id, book.book, book.author, book.dept, book.self, book.date);
        }
    }

    printf(
"=================================================================================================================\n");

    fclose(viu);
}
void viewbba() { //View BBA Book List from "view"  main funtion
    system("cls");
    FILE *viu;
    struct SI book;
    viu = fopen("book_bin.dat", "rb");

    printf(
"=================================================================================================================\n");
    printf("| %-10s | %-30s | %-20s | %-15s | %-10s | %-15s |\n", "Book ID", "Book Name", "Author",
"Department", "Self No", "Date");
    printf(
"=================================================================================================================\n");

    int startRecord = 800;  // Adjust the start record as needed
    int endRecord = 1199;    // Adjust the end record as needed

    // Move the file pointer to the start record
    fseek(viu, (startRecord - 1) * sizeof(struct SI), SEEK_SET);

    // Read and display records within the specified range
    for (int i = startRecord; i <= endRecord && fread(&book, sizeof(struct SI), 1, viu) == 1; i++) {
        if (book.id != 0) {
            printf("| %-10d | %-30s | %-20s | %-15s | %-10s | %-15s |\n",
                   book.id, book.book, book.author, book.dept, book.self, book.date);
        }
    }

    printf(
"=================================================================================================================\n");

    fclose(viu);
}
void datadelet(){ //Delete Book information from library system Funtion
    system("cls");
    char nm[100];
    struct SI book;
    FILE *dlt;

    dlt = fopen("book_bin.dat", "rb+");
    if (dlt == NULL) {
        printf("Error opening file for reading.\n");
        return;
    }

    printf("Enter Book Name : ");
    scanf("%s", nm);

    int found = 0;
```

```c

    // Loop until the end of the file or until the book is found
    while (!feof(dlt)) {
        fread(&book, sizeof(struct SI), 1, dlt);

        // Check if the book name matches
        if (strcmp(book.book, nm) == 0) {
               fseek(dlt, -sizeof(struct SI), SEEK_CUR);
               struct SI book={0,"", "" , "" ,"","",""};
               fwrite(&book,sizeof(struct SI),1,dlt);
                   found = 1;
            break;
        }

        }


    fclose(dlt);

    if (found==0) {
        printf("Issued Book not found!\n");
    }
    else{
        printf("Book Data Remove Succesfully!!!!!");
    }
}

void issue() { //Issue Book Function ,Page
    system("cls");
    char nm[100];
    struct SI book;
    struct ISS isbook;
    FILE *src;

    src = fopen("book_bin.dat", "rb");
    if (src == NULL) {
        printf("Error opening file for reading.\n");
        return;
    }

    printf("Enter Book Name : ");
    scanf("%s", nm);

    int found = 0;

    // Loop until the end of the file or until the book is found
    while (!feof(src)) {
        fread(&book, sizeof(struct SI), 1, src);

        // Check if the book name matches
        if (strcmp(book.book, nm) == 0) {
            // issue book file
            struct ISS isbook = {0, "", "", "", ""};
            FILE *fptr;

            fptr = fopen("issue_bin.dat", "rb+");
            if (fptr == NULL) {
                fptr = fopen("issue_bin.dat", "wb+");
                // do it for 2000 student
                for (int i = 0; i < 2000; i++) {
                    fwrite(&isbook, sizeof(struct ISS), 1, fptr);
                }
            }
            if (fptr != NULL) {
                system("cls");
                fptr = fopen("issue_bin.dat", "rb+");
```

```
627                 printf("\nEnter Student ID :");
628                 scanf("%d", &isbook.id);
629                 printf("\nEnter Student Name :");
630                 scanf("%s", isbook.name);
631                 printf("\nEnter Book Name :");
632                 scanf("%s", isbook.bk);
633                 printf("\nIssue Date :");
634                 scanf("%s", isbook.isdate);
635                 fseek(fptr, ((isbook.id%1000) - 1) * sizeof(struct ISS), SEEK_SET);
636                 fwrite(&isbook, sizeof(struct ISS), 1, fptr);
637             }
638 
639             fclose(fptr);
640 
641             found = 1;
642             break;
643         }
644     }
645 
646     fclose(src);
647 
648     if (found == 0) {
649         printf("Book not found!\n");
650     }
651 }
```