ООП 2021/2022 - Проект 1

Тема 2 – Електронна библиотека

Документация

В същността си тази програма представля съвкупност от операции, които моделират една база данни в зависимост от потребителските параметри. Програмата трябва успешно да зарежда и да записва електронната библиотека съответно от и на твърдата памет, да осъществява подходящ потребителски интерфейс, както и успешно и ефективно да изпълнява зададените от потребителя команди. Не се изискват сложни алгоритми, единствено правилно управление на динамичната памет е необходимо.

Проектът се състои от 4 основни компонента – функциите на main.cpp, и класовете Library, Book, PasswordManager. Функциите в main.cpp осъществяват потребителския интерфейс, както и връзката с електронната библиотека, Library осъществява самата библиотека, както и командите, които могат да се прилагат върху нея. Book описва една книга от библиотеката, а PasswordManager управлява паролите за оторизиран достъп. Изпълнението на програмата минава през следните етапи:

1) Зареждане на библиотеката и паролите в паметта от файловете

2) Получаване на заявки от потребителя и изпълнението им.

3) Записване на библиотеката и паролите във файлове, освбождаване на динамичната памет и прекратяване на програмата.

Файлът Book.h съдържа, освен класа Book, няколко помощни функции, няколко избриоми типа и константи. Константите определят максималната дължина на низовете, които характеризират една книга.

void write\_bintext(ofstream&f,const char\* низ); - тази функция записва низа в двоичния файл, свързан с потока f. Първо записва дължината на низа+1, а след това и самия низ, заедно с терминиращата нула.

enum SearchMode{title\_m,author\_m,isbn\_m,description\_m}; - тип изброим, който характеризира режим на търсене.

enum SortMode{title\_s,author\_s,rating\_s}; - тип изброим който характеризира режим на сортиране.

enum PrintMode{sent\_mode,row\_mode}; - тип изброим, характеризиращ режима на отпечатване на съдържанието на една книга на екрана.

bool str\_eq(const char\*,const char\*,bool ignore = false); - връща true ако двата стринга - параметъри са равни, false – иначе. Параметърът ignore определя дали при сравнение да се игнорира регистъра на буквите.

Класът Book се състои от следните член-данни с частен достъп:

char \*author,\*heading,\*directory,\*description,\*isbn;

Те представляват указател към динамична памет, където е записан низ със съответната информация за книгата.

int rating; рейтънгът на книгата

bool loaded; true – ако указателите по-горе сочат към динамична памет, false – ако са *nullptr.*

Класът притежава и една функция с вътрешен достъп

void deleteDyn(); - изтрива динамичната памет, сочена от указетилте по-горе.

Следващите член-функции имат публичен достъп:

Конструкторът по подразбиране на Book присвоява на указателите стойността *nullptr*, а на другите полета – стойността 0.

Останалите конструктори са стандартните копиращ конструктор и преместващ конструктор, а операторите за присвояване са също стандартни и се обръщат към функцията deleteDyn() за освобождаване на паметта на присвояващия обект. Деструкторът съдържа единствено обръщение към същата функция.

bool load(ifstream& str); - тази функция приема параметър поток, свързан с двоичен файл с информация за книгата. Зарежда тази информация в обекта като заделя динамична памет. Връща true ако информацията е успешно прочетена, false – иначе.

void setField(fieldType); - тези функции са стандартни мутатори за полето Field, като функцията присвоява параметъра си на полето.

bool compBook(const char\* низ,SearchMode,bool ignore); - Връща true ако низа-параметър съвпада с поле на обекта, определено от SearchMode, false – иначе. Параметърът ignore определя дали да се игнорира регистърът на буквите при сравнението.

bool lessThan(const Book& book,SortMode); - връща true ако низа, сочен от полето на обекта, оперделено от SearchMode е лексикографски по-малък от низа, сочен от съответното поле на параметъра book.

void print(); - отепчатва на екрана пълна информация за книгата

void shortPrint(); - отпечатва на екрана полетата heading, author, isbn.

bool getFile(ifstream&); - Ако файлът с директория directory съществува, функцията връща true и свързва потока с файла. False – иначе.

bool delFile(); - Ако файлът с директория directory съществува, функцията връща true и изтрива файла. False – иначе.

void save(ofstream& file) – Записва низовете, сочени от полетата на обекта в двоичния файл file, използвайки функцията write\_bintext.