

Общи изисквания към домашната работа

Решенията, които предавате, трябва да отговарят на следните общи условия и ограничения:

1. Използвайте ресурсите (процесор, памет, диск) **ефективно**.
2. Постарайте се да подсите коректна работа на програмата ви с всякакви входни данни (включително некоректни).
3. Можете да използвате класове, но следвайте добрите практики, обсъдени на лекции.
4. Решение, което **не се компилира** или **грубо нарушава** добрите практики, ще бъде оценено с **0 точки**.
5. Предавайте решенията си като **ZIP архив**, който съдържа само изходни файлове (**.cpp** и **.hpp**). Името на ZIP архива трябва да е **spec_groupX_fnY.zip**, където **spec** е специалността Ви (I или IS), **X** е номерът на групата ви, а **Y** е факултетният Ви номер.
6. Не е позволено използването на класа `std::string`, както и на контейнерите от STL.

Библиотека

Във всяка библиотека се съхраняват и предоставят на читатели различни печатни издания като книги, периодични издания (вестници, списания) и комикси.

Всяка книга се характеризира със следните данни:

- автор;
- заглавие;
- издателство;
- жанр - 16 битово цяло число - прочетете по-долу
- кратко описание;
- библиотечен номер (цяло число);
- година на издаване.

Периодичните издания се характеризират със:

- заглавие;
- периодичност (ежеседмично, ежемесечно, веднъж в годината);
- кратко описание;
- брой;
- библиотечен номер (цяло число);
- година на издаване.

Комиксите се характеризират със следните характеристики:

- автор;
- заглавие;
- издателство;
- жанр - 16 битово цяло число - прочетете по-долу;
- кратко описание;
- периодичност (ежеседмично, ежемесечно, веднъж в годината);
- брой;

- библиотечен номер (цяло число);
- година на издаване.

В библиотеката има определен брой екземпляри от всяко печатно издание. Читателите, които са регистрирани в библиотеката, могат да заемат избрани печатни издания за срок от един месец, но не повече от 5 наведнъж.

За всеки регистриран читател, освен неговото име, се съхраняват всички прочетени книги (заети и върнати) и всички книги, които текущо са заети от читателя, но все още не са върнати в библиотеката.

Реализирайте клас **Library**, който описва библиотека и поддържа информация за печатните издания и регистрираните читатели. Данните за наличните в библиотеката печатни издания, както и за читателите, се извличат от един или повече файлове. Форматът на представяне на данните е по ваш избор. В началото тези файлове може да са празни. В тази задача приемерте, че съдържанието на файловете е коректно (не е нужно да правите специални проверки за това).

Реализирайте като минимум следните функционалности за библиотеката::

- Добавяне на ново печатно издание в библиотеката, подадено като аргумент.
- Премахване на печатно издание по подаден библиотечен номер.
- Последователно извеждане на следната информация за всички печатни издания в библиотеката: заглавие, тип, кратко описание, библиотечен номер. Печатните издания да са подредени първо по година и след това по заглавие. Периодичните издания с еднакво заглавие първо се подреждат по брой, след което по азбучен ред на заглавието.
- Добавяне на потребител.
- Премахване на потребител по подадено име.
- Извеждане на всички печатни издания, които са просрочени, подредени по тип и по азбучен ред на заглавието.
- Извеждане на читателите, които са заели, но все още не са върнали, печатно издание с указан библиотечен номер.
- Извеждане на информация за всички читатели, подредени по броя прочетени книги.
- Заемане на печатно издание от даден читател. Печатното издание се определя от библиотечния си номер.
- Връщане на заето печатно издание от даден читател. Печатното издание се определя от библиотечния си номер.

Реализирайте клас **User**, който представя читателя в библиотеката.

За него дефинирайте следните оператори:

- Оператори за еквивалентност `==` и `!=`, сравняващи два потребителя по името им;
- Оператори за наредба (`<`, `<=`, `>`, `>=`), сравняват два потребителя по броя прочетени книги;

- Оператор за индексване [], с аргумент библиотечен номер на печатно издание, който връща информация дали читателят вече е прочел това издание, дали в момента го е заел, или никога не го е заемал;
- Оператори за събиране с печатно издание (+ и +=), които реализират заемане на съответното издание от потребителя;
- Оператори за изваждане с печатно издание (- и -=), които реализират връщане на това заето от потребителя издание;
- Всякакви други оператори, които ще са ви полезни в решаването на задачата.

Жанрът в книгата и комикса трябва да се получи чрез побитово комбиниране на следните четири компонента (до 16 битово цяло число без знак):

Характеристика - съхранен в младшите 7 бита. Характеристики са например фантастика, трилъър, криминална проза и т.н

Тип - съхранен в следващите 4 бита. Тип е например поезия, проза, драма и т.н.

Целева аудитория - съхранява се в следващите 4 бита. Целева аудитория е например деца, юноши, възрастни и т.н.

Контрол по четност - най-старшият бит е контролен бит. Неговата стойност е 0 или 1 така, че броят на всички единични битове (включително този) за жанра да е четно число.

Реализирайте подходящи методи или предефинирани оператори за класовете **Book** и **Comics**, които осигуряват достъп (за четене и промяна) на отделните компоненти на жанра, както и на жанра като цяло.