

# Progetto I: Biblioteca

Laboratorio di Algoritmi e strutture dati  
N86001980 Oreste Leone

## 1) Specifica del progetto.

Lo scopo del progetto consiste nella realizzazione di un programma per la gestione di una biblioteca per studenti che coinvolge due attori principali:

- La biblioteca, sotto forma di elenco di libri identificati da un titolo univoco;
- Gli studenti, identificati da una matricola univoca.

Ogni studente può effettuare una richiesta conforme a due tipi:

- Prestito di un libro;
- Restituzione di un libro, precedentemente preso in prestito.

Le richieste vanno soddisfatte seguendo ove possibile l'ordine in cui sono state effettuate.

Si richiede inoltre una soluzione per evitare deadlock dovuti a richieste sospese che non possono essere soddisfatte, aggiungendo inoltre che il programma possa terminare solo se tutte le richieste sono state soddisfatte.

## 2) Approccio implementativo.

In linea con le specifiche richieste nella traccia, il programma è stato sviluppato con precise scelte implementative:

- a) L'utilizzatore designato per il programma può essere considerato come un impiegato della biblioteca; Esso si interfacerà con il programma tramite una console di comando dove:

- Possono essere inserite le richieste effettuate dagli studenti tramite appositi form;
- Può essere gestita la prima richiesta possibile;
- Può essere visualizzato l'elenco completo dei libri del catalogo;
- Può essere visualizzato l'elenco dei libri attualmente in prestito a studenti;
- Concludere il programma (se non sono presenti richieste da gestire e se tutti i libri del catalogo sono stati restituiti);

Per facilitare la conoscenza degli status del programma sono stati inoltre inseriti degli indicatori numerici per la presenza (e quantità) di richieste in sospeso e di libri attualmente in prestito.

Si richiede all'utente utilizzatore, per l'immissione di dati alfabetici quali nomi degli studenti e nomi dei libri, di utilizzare ove necessario il carattere " \_ " <underscore> al posto del più usuale " " <spazio> per evitare comportamenti indesiderati. Inoltre, le stringhe utilizzate sono CASE SENSITIVE, per cui prestare attenzione all'utilizzo di lettere maiuscole e minuscole.

- b)** La biblioteca viene popolata all'avvio del programma da 15 libri identificati tramite il loro nome che resteranno immutati in notazione e numero per tutta la durata dell'esecuzione. È stato ritenuto più consono nel mantenere questi dati, l'implementazione dei libri sotto forma di nodi di un' ABR ( Albero Binario di Ricerca ) in quanto le continue operazioni di ricerca ne beneficeranno in costo di Tempo.

In particolare, un libro è una struttura contenente il nome del libro, il riferimento allo studente che ne ha fatto richiesta (nullo, se ancora disponibile) e i riferimenti ai sotto alberi sinistro e destro.

La specifica del riferimento allo studente che ne ha fatto richiesta, che viene aggiornata solo in seguito alla effettiva gestione del prestito o della restituzione, è dovuta alla scelta di non accettare che uno studente possa chiedere in prestito più di un libro, seguita anche dalla disponibilità di un'unica unità presente del libro; permette inoltre di effettuare un controllo più efficiente su quali libri siano effettivamente disponibili o in prestito.

La visualizzazione dell'elenco dei libri, ossia la stampa dell' ABR, viene effettuata tramite una visita in ordine e di conseguenza, secondo ordinamento alfanumerico secondo lo standard ASCII.

- c)** Gli studenti all'interno del programma sono definiti anch'essi come nodi di un ABR per gli stessi motivi precedenti; lo studente è dunque una struttura contenente la sua matricola numerica, il suo nome e i riferimenti ai sotto alberi sinistro e destro.

Contrariamente all'ABR dei libri, subisce ripetute procedure di inserimento e rimozione: uno studente viene aggiunto come nodo solo quando effettua una richiesta di prestito e quando viene gestita la restituzione del libro, egli viene eliminato. In questo modo il programma, non essendo interessato ad una memorizzazione permanente degli studenti, impedisce ad uno studente che ha già fatto richiesta o che ha attualmente in prestito un libro di richiederne altri.

Siccome questo tipo d'approccio potrebbe precludere lo studente a fare richiesta di un ulteriore libro nel caso in cui il libro da egli richiesto sia momentaneamente indisponibile, il programma domanderà all'utente, al momento della gestione della richiesta, se lo studente in questione sia intenzionato ad attendere che il libro venga restituito (spostando la sua richiesta alla fine della coda) o se preferisce cancellare la propria richiesta (eliminando la richiesta e conseguentemente il nodo dello studente) permettendo così un ulteriore richiesta di prestito.

- d)** Le richieste sono strutture definite come nodi di una coda in cui sono definite il tipo di richiesta, il riferimento allo studente richiedente e il riferimento al libro e il riferimento al prossimo elemento della coda. La coda, gestita come una struttura contenente un puntatore alla testa e uno alla coda, può essere considerata anche come una lista puntata circolare. Le richieste vengono create e gestite con due funzioni:

-*"catchRequest"* provvede alla creazione della richiesta nel caso in cui tutte le condizioni siano rispettate:

---Prestito:

Controlla se lo studente è presente all'interno dell'ABR, se è vero allora lo studente ha già fatto una richiesta in attesa di gestione o ha attualmente in prestito uno dei libri del catalogo e la richiesta viene annullata;

Controlla se il libro richiesto ha un riferimento attivo verso uno studente, se è vero allora il libro non è disponibile al prestito e la richiesta viene annullata;

---Restituzione:

Controlla se il libro restituito appartiene al catalogo, se è falso allora non è possibile effettuare una restituzione in quanto il libro non è della biblioteca e non si prevede un ampliamento del catalogo;

Se la condizione precedente è soddisfatta ma il libro è ancora disponibile in biblioteca e quindi non è stato richiesto in prestito, non è possibile effettuare la restituzione poiché significherebbe avere un duplicato e l'operazione viene annullata.

Se le precedenti condizioni sono tutte soddisfatte, il programma provvede alla creazione della richiesta e al suo accodamento.

-*"tryRequest"* provvede alla gestione della richiesta presente in testa alla coda delle richieste. Dato che i controlli sulle principali condizioni sono state eseguite al momento della richiesta, la maggioranza delle richieste potrà essere soddisfatta e di conseguenza rimossa dalla coda, in particolare nel caso della restituzione di un libro, si provvede anche a rimuovere lo studente coinvolto dall'ABR per potergli permettere una nuova richiesta di prestito.

Abbiamo appena considerato "la maggioranza", questo perché al tempo di gestione della richiesta, il libro richiesto potrebbe non essere disponibile perché un altro studente avrà fatto richiesta di prestito per lo stesso libro ma gestito precedentemente. Se ciò accade, per evitare un blocco nella gestione delle richieste il programma provvederà ad utilizzare il meccanismo descritto in precedenza alla fine del blocco c).

Grazie alle scelte effettuate per la definizione dei dati possiamo verificare che le principali operazioni effettuate dal programma avvengono con un costo di tempo nel caso peggiore di  $O(n)$ .

### 3) Esempio di esecuzione

All'avvio dell'eseguibile, il programma provvede a inizializzare le principali strutture e restituisce a schermo l'elenco dei libri presenti in catalogo per favorire le prime operazioni da effettuare. Successivamente ci presenterà un menù interattivo che mostrerà le operazioni che l'utente utilizzatore potrà effettuare:

Aggiungere una richiesta, gestire una richiesta (con indicatore numerico delle richieste in sospeso da gestire) , stampa del catalogo libri , stampa dei libri attualmente in prestito (con indicatore numerico di riferimento).

```
oresteleone@oresteleone-X540LA:~/Documenti/LASD/Progetto1/LASDN86001980$ ./executable
Inizializzazione programma...
Catalogo libri:
[1984] |delitto e castigo| |guerra e pace| |harry_potter| |hunger_games| |i_miserabili| |i_tre_moschettieri| |il_codice_da_vinci| |il_signore_degli_ane
lli| |il_silmarillion| |il_trono_di_spade| |moby_dick| |notre_dame_de_paris| |orgoglio_e_pregiudizio| |uno_nessuno_e_centomila|
***Fine Elenco***
***** Menu *****
1. Aggiungere una richiesta
2. Gestire le richieste (0)
3. Stampa catalogo libri
4. Stampa libri attualmente in prestito (0)
0. Terminare
input: []
```

Durante l'inserimento di una richiesta il programma ci chiederà l'inserimento di un valore numerico corrispondente ad una richiesta di Prestito o Restituzione. Nel caso della restituzione sarà necessario inserire il nome del libro. Per un prestito, il programma necessita dei dati per la creazione dello studente.

**\*\*\*NOTA BENE:** nell'immissione dati attenersi all'utilizzo di "\_" <underscore> al posto di " " <spazio> e ricordarsi che le stringhe sono **CASE SENSITIVE** (attenzione all'utilizzo di maiuscole e minuscole) come indicato alla fine del punto 2. a) \*\*\*

```
4. Stampa libri attualmente in prestito (0)
0. Terminare
input: 1

    Aggiungere una richiesta:

Inserisci tipo richiesta (1.Prestito//0.Restituzione) 1

Inserisci dati studente
Nome: Oreste_Leone
Matricola: 86001980

Inserisci nome libro richiesto in prestito: harry_potter

Elaborazione dati
Operazione effettuata con successo! Gestiremo la richiesta il prima possibile...

***Fine Operazione***

***** Menu *****
1. Aggiungere una richiesta
2. Gestire le richieste (1)
3. Stampa catalogo libri
4. Stampa libri attualmente in prestito (0)
0. Terminare
input: []
```

A richiesta inserita ritorneremo nel menù precedente, dove sarà aggiornato l'indicatore della gestione per sollecitare la memoria dell'utente utilizzatore nel gestire le richieste. Potremmo eseguire un'ulteriore aggiunta di richiesta o effettuare una delle altre operazioni, ma a scopo puramente illustrativo mostriamo l'iter per un'unica richiesta senza errori e consideriamo ora la gestione delle richieste.

In seguito alla gestione torneremo nuovamente al menù principale dove saranno nuovamente aggiornati gli indicatori per la gestione e per l'elenco dei libri in prestito. Effettuiamo dunque una stampa dei libri in prestito per far verificare al lettore l'effettiva presenza del prestito indicata dal nome del libro e le generalità dello studente che ha in prestito quel libro.

```
input: 2

Gestire le richieste:

Coda prima della gestione :
prestito      86001980      harry_potter

Coda dopo la gestione :

Non ci sono ulteriori richieste!

***Fine Operazione***

**** Menu ****
1. Aggiungere una richiesta
2. Gestire le richieste (0)
3. Stampa catalogo libri
4. Stampa libri attualmente in prestito (1)
0. Terminare
input: 4

Libri attualmente in prestito: 1
harry_potter ->Matricola: 86001980 Oreste_Leone

***Fine Elenco***

**** Menu ****
1. Aggiungere una richiesta
2. Gestire le richieste (0)
3. Stampa catalogo libri
4. Stampa libri attualmente in prestito (1)
0. Terminare
input: 1
```

Effettuiamo dunque una richiesta di restituzione inserendo il nome del libro che vogliamo restituire. Ritorneremo al menù che avrà di nuovo l'indice aggiornato per la gestione delle richieste. Eseguendo la gestione il libro verrà restituito e i libri in prestito saranno aggiornati. A questo punto è possibile concludere il programma poiché non ci sono richieste in sospeso e non ci sono libri attualmente in prestito.

```
DEBUG CONSOLE  PROBLEMS  OUTPUT  TERMINAL  1:executable

Inserisci tipo richiesta (1.Prestito//0.Restituzione) 0

Inserisci nome libro da restituire: harry_potter

Elaborazione dati
Operazione effettuata con successo! Gestiremo la richiesta il prima possibile...

***Fine Operazione***

**** Menu ****
1. Aggiungere una richiesta
2. Gestire le richieste (1)
3. Stampa catalogo libri
4. Stampa libri attualmente in prestito (1)
0. Terminare
input: 2

Gestire le richieste:

Coda prima della gestione :
restituzione  86001980      harry_potter

Coda dopo la gestione :

Non ci sono ulteriori richieste!

***Fine Operazione***

**** Menu ****
1. Aggiungere una richiesta
2. Gestire le richieste (0)
3. Stampa catalogo libri
4. Stampa libri attualmente in prestito (0)
0. Terminare
```

Alcuni esempi di esecuzione interessanti:

-Gestione richieste sospese per libro in prestito ad un altro studente

1(aggiungere una richiesta); 1(prestito); oreste\_leone (nome); 86001980(matricola); harry\_potter(libro);

1(aggiungere una richiesta); 1(prestito); stefania (nome); 86002022(matricola); harry\_potter(libro);

1(aggiungere una richiesta); 1(prestito); marco (nome); 86001060(matricola); il\_trono\_di\_spade(libro);

2(gestire le richieste); // prestito effettuato e coda aggiornata con [prestito 86002022 harry\_potter] → [prestito 86001060 il\_trono\_di\_spade]

2(gestire le richieste); **Messaggio di avviso e richiesta allo studente**

Inserendo 1 avremo una coda aggiornata con [prestito 86001060 il\_trono\_di\_spade] → [prestito 86002022 harry\_potter]

Inserendo 0 la richiesta per il prestito di harry\_potter da parte di stefania verrà cancellata e potrà effettuare una nuova richiesta di prestito.