Simulasi UTP Data Structures

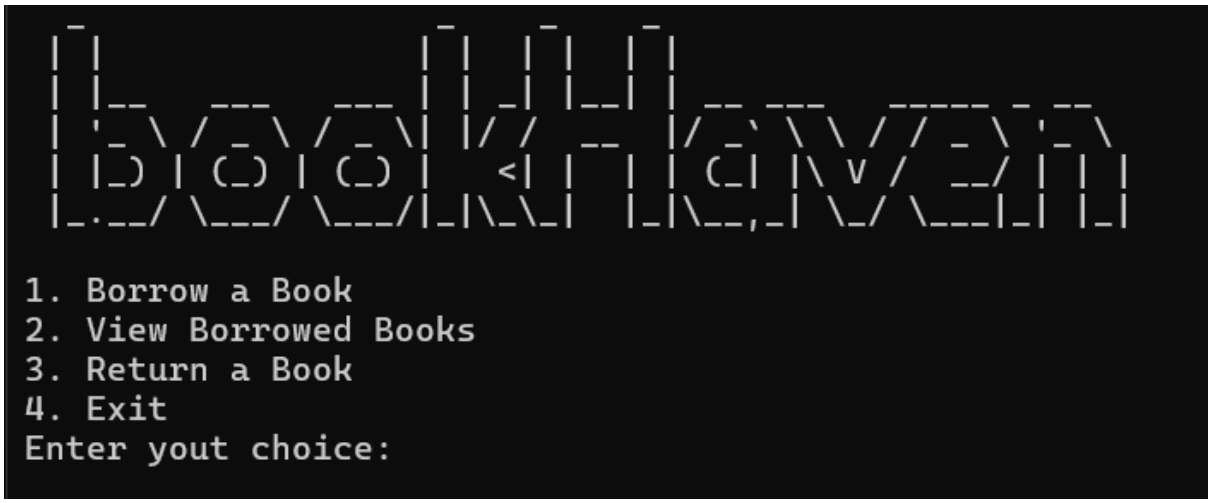
Soal

BookHaven

A newly established digital library, **BookHaven**, wants to develop a system for managing book borrowing and returning efficiently. You, as an experienced programmer, are tasked with building a program in **C language** using a **hash table with chaining method** to handle book transactions.

The program should have the following functionalities:

1. **Borrow a Book**
2. **View Borrowed Books**
3. **Return a Book**
4. **Exit**



1. Borrow a Book

If the user selects this option, the program should:

- Prompt the user to enter their **full name** (3-30 characters).
- Prompt the user to enter their **library ID** (must start with “LIB-” followed by exactly 5 digits).
- Prompt the user to enter the **book title** (3-50 characters).
- Prompt the user to enter the **borrowing duration** (1-30 days).

The system should generate a **unique Borrowing ID** using the following format:
BBXXX

- **BB** = First two characters of the book title (uppercase).
- **XXX** = A random number between 000 – 999.

Example:
Book Title: *Harry Potter* → Borrowing ID: **HA440**

Hash Table Key Calculation (Mid-Square Division Method)

1. Take the **last three digits** of the Borrowing ID as the number (**N**).
2. Square the number (N^2).
3. Extract the **middle digits** of the squared result.
4. Compute the **hash key** as follows:

Key = (Middle Digits of (N^2)) % Table Size

Table Size = 100

Example Calculation:

- Borrowing ID: **HA440** → Take **440**
- Square it: **440^2 = 193600**
- Extract the middle digits: **36**
- Compute key: **36 % 100 = 36**

The book entry is stored at index **36** in the hash table.

After storing the data, display a success message.

```
Enter your full name: a
Input must be between 3 and 30 characters
Enter your full name: Azimuth
Enter your library ID (format: LIB-xxxxx): LIB-2abcd
Input must start with "LIB-" followed by exactly 5 digits
Enter your library ID (format: LIB-xxxxx): LIB-12345
Enter the book title: Harry Potter
Enter borrowing duration (1-30 days): 2
```

```
Book Borrowed Successfully!
Borrowing ID : HA440
Borrower Name : Azimuth
Library ID : LIB-12345
Book Title : Harry Potter
Duration : 2
Press ENTER To Continue...
```

2. View Borrowed Books

- If no books are borrowed, display a message:
  "No books borrowed yet"

```
No books borrowed yet
Press ENTER To Continue...
```

- Otherwise, display all borrowed books.

```
Borrowing ID : HA440
Borrower Name : Azimuth
Library ID : LIB-12345
Book Title : Harry Potter
Duration : 2
============================
Press ENTER To Continue...
```

3. Return a Book

- If no books are borrowed, display a message:
  "No books found"

```
No books found
Press ENTER To Continue...
```

- Otherwise, show all borrowed books and prompt the user to enter the **Borrowing ID**.

```
Borrowing ID : HA440
Borrower Name : Azimuth
Library ID : LIB-12345
Book Title : Harry Potter
Duration : 2
============================

Enter Borrowing ID to return:
```

- If the entered ID does not exist, display an error message.

```
No books found
Press ENTER To Continue...
```

- If found, remove the book from the system and show a success message.

```
Book Returned Succesfully!
Borrowing ID : HA440 has been removed
Press ENTER To Continue...
```

4. Exit

- Close the program.

```
Thank you for using our program...
Press ENTER To Continue...
```