

PERPUSTAKAAN “YU EN I ES EI”



Sumber: [brewminate.com](https://www.brewminate.com)

Deskripsi Praktikum

Pada pengerjaan tugas kali ini kalian akan menerapkan 3 bentuk data struktur yaitu *Linked List*, *Queue*, dan *Stack*. Untuk pengumpulan akan menggunakan google classroom dalam bentuk zip dan progress pengerjaan akan di cek secara berkala setiap minggu. Jadi setiap minggu kalian akan melakukan *update* terhadap fitur *project* kalian berdasarkan perintah yang diberikan.

Tugas dikerjakan secara individu dan untuk bahasa pemrograman menggunakan *low-level programming language* (Pastikan terdapat konsep *pointer* dan *reference*). Program yang dikerjakan dalam bentuk *CLI* namun bila diterapkan dengan *GUI* (*Graphical User Interface*) maka bisa menjadi nilai tambahan.

Pada proses pembuatan juga perlu diperhatikan bahwa dalam proses pembuatan struktur data harus dilakukan tanpa bantuan *library* (*from scratch*) bila pada bahasa pemrograman C++ menggunakan konsep *pointer* dan *struct*.

 **Plagiasi dan segala bentuk kecurangan lainnya akan berakibat pada pengurangan nilai !**

Inisiasi Project

Pada awal pengerjaan, buatlah sistem untuk membagi hak akses aplikasi dengan role **pegawai** dan **pengunjung**. Sehingga, tampilan awal program akan menjadi seperti berikut:

```
Unset
=====
                SELAMAT DATANG !!
=====
|1. Masuk sebagai Pegawai
|2. Masuk sebagai Pengunjung
>
```

Pegawai dan pengunjung akan memiliki fungsi yang berbeda sesuai dengan privileges yang dimilikinya. Berikut adalah Dashboard pegawai:

```
Unset
=====
                DASHBOARD ADMIN
=====
|1. Menambah Data Buku
|2. Tampilkan Data Buku
|3. Mengubah Data Buku
|4. Menghapus Data Buku
|0. Keluar
>
```

Pada dashboard pengunjung, akan langsung menampilkan data buku untuk melakukan peminjaman (Untuk lebih lengkapnya akan dibahas pada Praktikum 1)

```
Unset
=====
                DASHBOARD PENGUNJUNG
=====

Manakah yang ingin kamu pinjam?
-----
| Judul    : Bumi Manusia
| Id       : 31180
| Penulis  : Pramoedya Ananta Toer
```

```
| Tahun   : 1980  
| Genre   : Fiksi
```

```
-----
```

```
| 1. Selanjutnya  
| 2. Sebelumnya  
| 3. Pinjam  
| 0. Keluar  
>
```

Praktikum 1 - Linked List

Pada minggu ini, kita akan menerapkan *linked list* dan *basic operation* pada programnya. Program *Linked list* menyimpan data buku yang ada di perpustakaan. Berikut adalah struktur nodenya:

```
C/C++
Data Buku {
    Id      : String
    Judul   : String
    Penulis : String
    Tahun   : Number
    Genre   : String
    Tersedia : Boolean
}
```

Hak Akses Pegawai

Pada sisi pegawai, dapat menambahkan, melihat, mengubah, dan menghapus data buku. Untuk menambah data buku adalah seperti berikut:

```
C/C++
Judul   : > (Masukkan judul)
Penulis : > (Masukkan nama penulis)
Tahun   : > (Masukkan tahun terbit)
Genre   : > (Masukkan genre)

Buku berhasil ditambahkan !
```

Untuk Id akan ter-generate oleh program dengan pola seperti berikut :

- Id terdiri dari 5 digit angka
- 2 digit pertama merupakan huruf terakhir yang dikurangi huruf pertama dari judul buku dan bernilai mutlak. Contoh judul “Bumi Manusia” memiliki huruf akhir “a” dan huruf depan “B”, maka 2 digit pertamanya adalah $97 - 66 = 31$
- Digit ketiga dan keempat merupakan hasil penjumlahan dari semua digit tahun terbitnya. Misalnya, buku Bumi Manusia terbit pada tahun 1980, jadi digit ketiga dan keempatnya adalah $1 + 9 + 8 + 0 = 18$. Apabila hasil penjumlahan kurang dari 10 (1 digit), maka tambahkan 0 di-depannya.

- Digit kelima akan bernilai 0 apabila tidak ada Id yang sama. Namun, jika terdapat duplikasi Id, digit kelima akan terus di-increment. Bila nilai digit sudah mencapai angka 9 maka digitnya akan bertambah 1 yang semula 5 digit menjadi 6 digit. Hal ini akan terus dilakukan selama terjadi duplikasi Id.

Pada saat ditambahkan, ketersediaan buku akan menjadi true secara default yang menandakan bahwa buku tersedia.

Buku yang sudah ditambahkan, dapat ditampilkan pada menu Tampilkan Data Buku. Berikut ini adalah menu Tampilkan Data Buku:

```
Unset
-----
| Judul      : Bumi Manusia
| Id         : 31180
| Penulis    : Pramoedya Ananta Toer
| Tahun      : 1980
| Genre      : Fiksi
| Ketersediaan : Tersedia
-----

| 1. Selanjutnya
| 2. Sebelumnya
| 0. Keluar
>
```

Pada kasus ini, diharapkan pegawai mampu melakukan iterasi terhadap data buku, arah iterasi dikontrol menggunakan command 'Selanjutnya' dan 'Sebelumnya' untuk mengakses data. Data awal dimulai pada index ke-0. Apabila pada saat menampilkan data buku telah mencapai index terakhir, maka saat ingin melanjutkan akan kembali ke daftar buku pada index ke-0.

Untuk mengubah data buku, pegawai dapat melakukan perubahan terhadap semua data buku kecuali Id dan ketersediaan. Apabila ada data yang mempengaruhi Id buku, maka Id buku tersebut akan di-generate kembali. Proses pengubahan data buku dapat dilakukan berdasarkan Id buku seperti berikut:

```
Unset
Masukkan Id Buku ingin diubah > 31180
```

```
- Mengubah buku dengan Id 31180 -
|1. Ubah judul
|2. Ubah penulis
|3. Ubah tahun
|4. Ubah genre
|0. Kembali
> 1
```

Masukkan judul buku baru > BUMI MANUSIA

Data Id 02130 telah berubah !

```
-----
| Judul      : BUMI MANUSIA
| Id         : 01180
| Penulis    : Pramoedya Ananta Toer
| Tahun      : 1980
| Genre      : Fiksi
| Ketersediaan : Tersedia
-----
```

Untuk menghapus data buku, pegawai dapat melakukannya dengan mengakses buku melalui id buku. Interface dapat berupa tampilan seperti berikut :

```
Unset
Masukkan Id Buku yang ingin diubah > 31180

- Detail Buku -
-----
| Judul      : BUMI MANUSIA
| Id         : 01180
| Penulis    : Pramoedya Ananta Toer
| Tahun      : 1980
| Genre      : Fiksi
-----

Yakin ingin menghapus buku ? (y/n) >
```

Hak Akses Pengunjung

Pada dashboard pengunjung, akan ditampilkan data buku yang telah tersedia. Data buku ditampilkan seperti pada dashboard admin.

```
Unset
-----
| Judul      : BUMI MANUSIA
| Id         : 01180
| Penulis    : Pramoedya Ananta Toer
| Tahun      : 1980
| Genre      : Fiksi
| Ketersediaan : Tersedia
-----

| 1. Selanjutnya
| 2. Sebelumnya
| 3. Pinjam
| 0. Keluar
>
```

Pada kasus ini, diharapkan pengunjung mampu melakukan iterasi terhadap data buku yang tersedia, arah iterasi dikontrol menggunakan command 'Selanjutnya' dan 'Sebelumnya' untuk mengakses data. Data awal dimulai pada index ke-0. Apabila pada saat menampilkan data buku telah mencapai index terakhir, maka saat ingin melanjutkan akan kembali ke daftar buku pada index ke-0.

Opsi pinjam buku diberikan agar pengunjung dapat melakukan peminjaman buku (Untuk lebih lengkapnya, akan ada di praktikum selanjutnya. Sementara, fungsi ini tidak perlu melakukan peminjaman).

Praktikum 2 - Queue

Pada praktikum ke 2, kalian akan menerapkan struktur data queue pada project sebelumnya. Masih ingatkah kalian dengan fungsi pinjam buku pada dashboard pengguna? Pada praktikum ini kalian akan menerapkan queue pada fungsi pinjam buku. Berikut ini adalah struktur node untuk data pinjaman buku:

```
C/C++
Data Pinjaman {
    Id      : String
    Judul   : String
    Nama    : String
    NIM     : Integer
    Tanggal : Datetime
}
```

Hak Akses Pengguna

Pengguna mampu melakukan permintaan pinjaman buku. Nantinya, permintaan akan diproses oleh pegawai dimulai dari pinjaman yang pertama masuk. Contoh kasusnya seperti ini :

```
Unset
-----
| Judul      : BUMI MANUSIA
| Id         : 01180
| Penulis    : Pramoedya Ananta Toer
| Tahun      : 1980
| Genre      : Fiksi
| Ketersediaan : Tersedia
-----

| 1. Selanjutnya
| 2. Sebelumnya
| 3. Pinjam
| 0. Keluar
> 3

Masukkan nama peminjam > Azril
Masukkan NIM peminjam > 24051204000
```


Permintaan pinjaman telah berhasil dibuat !!

```
-----
| Id           : 31180f25az03
| Judul        : Bumi Manusia
| Nama         : Azril
| NIM          : 24051204000
| Tanggal      : 8 Maret 2025 08:50
|-----
```

Menunggu persetujuan pegawai

Terdapat aturan dalam pembuatan Id pinjaman, berikut merupakan aturan dalam pembuatan Id pinjaman :

- 5 digit pertama: ID buku yang dipinjam.
- 1 huruf (digit ke-6): Inisial kategori buku, misalnya:
F → Fiksi
N → Non-Fiksi
S → Sains
- 2 digit berikutnya (ke-7 & ke-8): Dua digit terakhir dari tahun peminjaman.
- 2 huruf berikutnya (ke-9 & ke-10): Inisial nama peminjam (2 huruf pertama dari nama).
- 2 digit terakhir (ke-11 & ke-12): Nomor urut transaksi peminjaman pada hari itu.

Contoh :

- ID Buku: **31180**
- Kategori **Buku**: Fiksi (**f**)
- Tahun Peminjaman: 2025 (**25**)
- Nama Peminjam: Azril (**az**)
- Nomor Urut Peminjaman: **03**

Maka Id pinjaman yang terbentuk adalah:

31180f25az03

Semua permintaan pinjaman yang telah dilakukan oleh pengguna akan disimpan didalam queue untuk diproses lebih lanjut pada dashboard pegawai.

Hak Akses Pegawai

Pada dashboard pegawai, akan ada opsi baru yang ditambahkan yaitu opsi untuk memproses permintaan pinjaman. Berikut contohnya:

```

Unset
=====
                DASHBOARD PEGAWAI
=====
|1. Menambah Data Buku
|2. Tampilkan Data Buku
|3. Mengubah Data Buku
|4. Menghapus Data Buku
|5. Proses Permintaan Pinjaman
|0. Keluar
>

```

Pada opsi “Proses Permintaan Pinjaman”, akan menampilkan permintaan pinjaman yang telah dibuat oleh pengguna sebelumnya. Akan tetapi, pada menu opsi ini tidak terdapat “Selanjutnya” dan “Sebelumnya”. Opsi tersebut nantinya akan diganti dengan “Setujui” dan “Tolak”. Untuk lebih detailnya perhatikan contoh berikut :

```

Unset
=====
                PINJAMAN KE-1
=====

-----
| Id           : 31180f25az03
| Judul        : Bumi Manusia
| Nama         : Azril
| NIM          : 24051204000
| Tanggal      : 8 Maret 2025 08:50
-----

| 1. Setujui
| 2. Tolak
| 0. Keluar
> 1

Pinjaman dengan id 31180f25az03 telah diproses !!

=====
                PINJAMAN KE-2
=====

-----
| Id           : 31180f25az03
| Judul        : Jago Coding C++

```

```
| Nama      : Amba
| NIM       : 24051204999
| Tanggal   : 8 Maret 2025 11:35
-----
```

```
| 1. Setujui
| 2. Tolak
| 0. Keluar
> 2
```

Pinjaman yang muncul pertama kali adalah pinjaman yang pertama kali dibuat oleh user. Pada saat pegawai menyetujui atau menolak maka pinjaman akan dihapus dari antrian dan akan lanjut ke pinjaman selanjutnya sesuai konsep antrian. Dan apabila pinjaman disetujui, buat ketersediaan buku menjadi False. Pada sisi pegawai bila tidak ada pinjaman sama sekali maka tampilkan tulisan tidak ada pinjaman dan kembali ke dashboard pegawai.

```
Unset
```

```
=====
                DASHBOARD PEGAWAI
=====
|1. Menambah Data Buku
|2. Tampilkan Data Buku
|3. Mengubah Data Buku
|4. Menghapus Data Buku
|5. Proses Permintaan Pinjaman
|0. Keluar
> 5
```

```
Tidak ada pinjaman untuk diproses !!
```

```
=====
                DASHBOARD PEGAWAI
=====
|1. Menambah Data Buku
|2. Tampilkan Data Buku
|3. Mengubah Data Buku
|4. Menghapus Data Buku
|5. Proses Permintaan Pinjaman
|0. Keluar
>
```

Praktikum 3 - Stack

Pada praktikum ke 3, kalian akan menerapkan struktur data stack pada project sebelumnya. Masih ingatkah kalian dengan fungsi pinjam buku di praktikum sebelumnya? Ketika pinjaman telah disetujui oleh pegawai, data pinjaman akan dihapus dari queue. Pada praktikum ini, kamu diminta untuk menambahkan sebuah fungsi baru, yaitu menyimpan data peminjaman yang telah disetujui ke dalam struktur data linked list. Fungsi ini bertujuan untuk mendukung fitur selanjutnya.

Hak Akses Pengguna

Pada dashboard pengguna, akan ada opsi baru yang ditambahkan yaitu opsi untuk melihat riwayat pinjaman. Berikut contohnya:

```
Unset
=====
                DASHBOARD PENGGUNA
=====
|1. Daftar Buku
|2. Riwayat Pinjaman
|0. Keluar
>
```

Opsi “Daftar Buku” menampilkan daftar buku seperti yang telah dibuat sebelumnya. Sementara itu, menu “Riwayat Pinjaman” akan menampilkan data riwayat pinjaman yang telah disimpan dalam linked list. Riwayat pinjaman akan ditampilkan satu per satu, dengan menyediakan opsi navigasi seperti "Selanjutnya", "Sebelumnya", dan "Kembalikan Buku". Berikut adalah contohnya:

```
Unset
-----
| Id           : 31180f25az03
| Judul        : Bumi Manusia
| Nama         : Azril
| NIM          : 24051204000
| Tanggal     : 8 Maret 2025 08:50
-----
```

```
| 1. Selanjutnya
| 2. Sebelumnya
| 3. Kembalikan Buku
| 0. Keluar
>
```

Ketika pengguna memilih opsi "Kembalikan Buku" dari menu navigasi riwayat pinjaman, proses pengembalian buku akan dimulai. Setelah pengguna mengonfirmasi pengembalian, data pinjaman akan dihapus dari linked list riwayat pinjaman. Kemudian, data akan disimpan dalam struktur data stack pengembalian dan menunggu diproses oleh pegawai. Berikut ini adalah contohnya:

```
Unset
-----

| Id           : 31180f25az03
| Judul        : Bumi Manusia
| Nama         : Azril
| NIM          : 24051204000
| Tanggal      : 8 Maret 2025 08:50
-----

| 1. Selanjutnya
| 2. Sebelumnya
| 3. Kembalikan Buku
| 0. Keluar
> 3

Apakah Anda yakin ingin mengembalikan buku "Bumi Manusia"?
| 1. Ya
| 2. Tidak
> 1

Buku "Bumi Manusia" telah dikembalikan !!
```

Hak Akses Pegawai

Pada dashboard pegawai, akan ada opsi baru yang ditambahkan yaitu opsi untuk memproses pengembalian buku. Berikut contohnya:

```

Unset
=====
                DASHBOARD PEGAWAI
=====
|1. Menambah Data Buku
|2. Tampilkan Data Buku
|3. Mengubah Data Buku
|4. Menghapus Data Buku
|5. Proses Permintaan Pinjaman
|6. Proses Pengembalian Buku
|0. Keluar
>

```

Pada opsi “Proses Pengembalian Buku”, akan menampilkan buku-buku yang telah dikembalikan sebelumnya. Akan tetapi, pada menu opsi ini tidak terdapat “Selanjutnya” dan “Sebelumnya”. Opsi tersebut nantinya akan diganti dengan “Proses”. Untuk lebih detailnya perhatikan contoh berikut :

```

Unset
=====
                PENGEMBALIAN KE-1
=====

-----
| Id           : 31180f25az03
| Judul        : Bumi Manusia
| Nama         : Azril
| NIM          : 24051204000
| Tanggal      : 8 Maret 2025 08:50
-----

| 1. Proses
| 0. Keluar
> 1

Buku "Bumi Manusia" telah dikembalikan !!

=====
                PENGEMBALIAN KE-2
=====

-----
| Id           : 31180f25az03
| Judul        : Jago Coding C++
| Nama         : Amba

```

```

| NIM          : 24051204999
| Tanggal      : 8 Maret 2025 11:35
-----

| 1. Proses
| 0. Keluar
> 1

Buku "Jago Coding C++" telah dikembalikan !!

```

Pengembalian yang muncul pertama kali adalah pengembalian yang terakhir kali dibuat oleh user. Pada saat pegawai memproses pengembalian, data pengembalian akan dihapus dari stack pengembalian dan akan lanjut ke pengembalian selanjutnya sesuai konsep stack. Setelah pengembalian diproses, ubah ketersediaan buku menjadi True agar bisa dipinjam kembali. Apabila tidak ada pengembalian yang diproses (stack pengembalian kosong), tampilkan pesan bahwa tidak ada pengembalian yang perlu diproses dan kembali ke dashboard pegawai.

```

Unset
=====
                DASHBOARD PEGAWAI
=====
|1. Menambah Data Buku
|2. Tampilkan Data Buku
|3. Mengubah Data Buku
|4. Menghapus Data Buku
|5. Proses Permintaan Pinjaman
|6. Proses Pengembalian Buku
|0. Keluar
> 6

Tidak ada pengembalian untuk diproses !!

=====
                DASHBOARD PEGAWAI
=====
|1. Menambah Data Buku
|2. Tampilkan Data Buku
|3. Mengubah Data Buku
|4. Menghapus Data Buku
|5. Proses Permintaan Pinjaman
|6. Proses Pengembalian Buku

```

```
|0. Keluar  
>
```

Catatan

Kamu dapat mengembangkan project ini lebih lanjut sesuai dengan kreativitas kamu, seperti menyimpan data ke dalam file teks (.txt), membuat antarmuka pengguna berbasis GUI (Graphical User Interface), atau menambahkan fitur-fitur lainnya. Namun, perlu diingat bahwa semua pengembangan tersebut harus tetap memenuhi kriteria dan persyaratan yang telah ditentukan. Pengembangan yang telah kamu buat dapat menjadi nilai tambahan.

 Plagiasi dan segala bentuk kecurangan lainnya akan berakibat pada pengurangan nilai !