

# **PROJECT STRUKTUR DATA DAN ALGORITMA**

## **“ Program Gudang Penyimpanan “**

**Disusun Untuk Memenuhi Tugas Mata Kuliah Struktur Data dan Algoritma**

**Dosen Pengampu : Jefril Rahmadoni, M.Kom**



**Disusun Oleh:**

**Kelas 02**

**Kelompok 1**

**Anggota :**

<b>Muhammad Harley Adinama</b>	<b>1811521004</b>
<b>Tri Ayunia Patma Lubis</b>	<b>2011522010</b>
<b>Deyola Fadwa Shifana</b>	<b>2011522032</b>
<b>Iqbal Firmansyah</b>	<b>2111521014</b>
<b>Muhammad Satria Gemilang Lubis</b>	<b>2111522008</b>
<b>Syahniah Putri hendry</b>	<b>2111522020</b>
<b>Nazhiva Darma Fitri</b>	<b>2111523004</b>
<b>Aldo Agustio</b>	<b>2111523016</b>

**PROGRAM STUDI S1 SISTEM INFORMASI**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS ANDALAS**

**2022**

## KATA PENGANTAR

Assalamu'alaikum warahmatullahi wabarakatuh.

Segala puji bagi Allah SWT yang telah memberikan penulis kemudahan sehingga penulis dapat menyelesaikan laporan ini dengan tepat waktu. Tanpa pertolongan-Nya tentunya penulis tidak akan sanggup untuk menyelesaikan laporan ini dengan baik. Shalawat serta salam semoga terlimpah curahkan kepada baginda tercinta kita yaitu Nabi Muhammad SAW yang kita nanti-nantikan syafa'atnya di akhirat nanti.

Penulis mengucapkan syukur kepada Allah SWT atas limpahan nikmat sehat-Nya, baik itu berupa sehat fisik maupun akal pikiran, sehingga penulis mampu untuk menyelesaikan pembuatan laporan sebagai tugas dari mata kuliah Struktur Data dan Algoritma dengan judul “ **Program Gudang Penyimpanan**”.

Penulis tentu menyadari bahwa laporan ini masih jauh dari kata sempurna dan masih banyak terdapat kesalahan serta kekurangan di dalamnya. Untuk itu, penulis mengharapkan kritik serta saran dari pembaca untuk laporan ini, supaya laporan ini nantinya dapat menjadi laporan yang lebih baik lagi. Kemudian apabila terdapat banyak kesalahan pada laporan ini penulis mohon maaf yang sebesar-besarnya.

Penulis juga mengucapkan terima kasih kepada semua pihak khususnya kepada dosen mata kuliah Struktur Data dan Algoritma yaitu Bapak **Jeofil Rahmadoni, M.Kom** yang telah membimbing, mengajarkan serta membantu penulis dalam menyelesaikan tugas besar ini.

Demikian, semoga laporan ini dapat bermanfaat. Terima kasih.

Padang, 22 Maret 2022

Penulis

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	<b>2</b>
<b>DAFTAR ISI.....</b>	<b>3</b>
<b>BAB I PENDAHULUAN.....</b>	<b>4</b>
<b>A. Latar Belakang Masalah .....</b>	<b>4</b>
<b>B. Rumusan Masalah .....</b>	<b>4</b>
<b>C. Tujuan.....</b>	<b>4</b>
<b>D. Manfaat.....</b>	<b>5</b>
<b>E. Sistematika Laporan.....</b>	<b>5</b>
<b>BAB II PEMBAHASAN.....</b>	<b>7</b>
<b>A. Profil Studi Kasus .....</b>	<b>7</b>
<b>B. Flowchart Program.....</b>	<b>8</b>
<b>C. Codingan dan Hasil Compile .....</b>	<b>11</b>
<b>BAB III PENUTUP.....</b>	<b>25</b>
<b>A. Kesimpulan.....</b>	<b>25</b>
<b>B. Saran .....</b>	<b>25</b>
<b>REFERENSI.....</b>	<b>26</b>

## **BAB I PENDAHULUAN**

### **A. Latar Belakang Masalah**

Gudang merupakan bagian penting dalam sebuah perusahaan. Gudang merupakan bangunan untuk menyimpan barang, sedangkan pergudangan adalah kegiatan atau aktivitas menyimpan barang. Sistem pergudangan yang kurang baik dapat menyebabkan adanya barang kadaluarsa, kehilangan barang dan lain sebagainya yang pada akhirnya mengurangi pendapatan perusahaan.

Pergudangan yang baik adalah pergudangan yang memiliki sistem pelayanan yang baik. Sistem pelayanan yang baik mencakup adanya jaminan keamanan, kemudahan akses informasi keluar, informasi masuk, dan penyimpanan barang, serta kesesuaian kondisi lingkungan fisik bagi barang yang disimpan (Warman, 2004). Tidak terkecuali untuk gudang penyimpanan alat-alat olahraga dimana biasanya gudang tersebut menyimpan banyak alat olahraga seperti shuttlecock, matras, dan rompi futsal.

Dengan laju pesatnya perkembangan teknologi informasi dan sistem informasi, tidak menutup kemungkinan gudang penyimpanan seperti gudang penyimpanan alat-alat olahraga memanfaatkan perkembangan teknologi ini dalam proses kerjanya untuk mempermudah pekerjaan karyawan. Program akan sangat membantu dalam pendataan alat yang tersedia, yang diambil, dan dikembalikan menggunakan array, linkedlist, stack, queue, dan binary tree.

Berdasarkan uraian masalah di atas, penulis memutuskan untuk membuat “**Program Gudang Penyimpanan**” yang awalnya dilakukan dengan sistem manual dapat menjadi teromatisasi. Selain itu, hal-hal mengenai kegiatan pada Gudang Penyimpanan dilakukan dengan meinputkan stok yang dibutuhkan pada pengambilan dan pengembalian alat-alat yang ada.

### **B. Rumusan Masalah**

Berdasarkan uraian latar belakang di atas, dapat disimpulkan rumusan masalahnya sebagai berikut :

1. Bagaimanakah cara kerja program Gudang Penyimpanan tersebut?
2. Bagaimanakah penerapan array, linkedlist, stack, queue, dan binary tree pada program?

### **C. Tujuan**

Berdasarkan latar belakang masalah dan rumusan masalah di atas dapat dirumuskan tujuan sebagai berikut :

1. Pembuatan Program
  - A. Memenuhi tugas Struktur Data dan Algoritma
  - B. Menghasilkan program Gudang Penyimpanan yang terdiri dari beberapa sub program untuk melakukan peletakkan dan pengambilan barang dengan metode array, linkedlist, stack, dan queue.

## 2. Pembuatan Laporan

- A. Memenuhi tugas Struktur Data dan Algoritma.
- B. Membantu pembaca dalam memahami materi mengenai array, linkedlist, stack, queue, dan binary tree pada proses peletakkan dan pengambilan barang.
- C. Membantu pembaca dalam memahami program dan jalannya program.

## D. Manfaat

Berdasarkan latar belakang di atas, manfaat dari pembuatan laporan ini adalah:

- A. Menerapkan ilmu yang telah di dapat yaitu mengenai array, linkedlist, stack, queue, dan binary tree
- B. Mengimplementasikan pembelajaran mengenai array, linkedlist, stack, queue, dan binary tree pada Visual Studio Code.
- C. Memenuhi tugas mata kuliah Struktur Data dan Algoritma.

## E. Sistematika Laporan

Berdasarkan latar belakang masalah, rumusan masalah, manfaat dan tujuan, sistematika penulisan laporan dapat dirumuskan sebagai berikut :

### 1. Bagian awal terdiri dari beberapa unsur penting sebagai berikut :

- A. Lembar judul dan cover adalah identitas yang memberikan gambaran mengenai isi laporan
- B. Kata pengantar berisikan ucapan syukur kepada Allah SWT dan ucapan terima kasih kepada pihak-pihak yang membantu pembuatan laporan
- C. Daftar isi adalah suatu daftar yang membuat gambaran isi karya tulis secara menyeluruh

### 2. BAB I PENDAHULUAN

Pada bab I ini akan berisi :

- A. Latar belakang permasalahan adalah fenomena permasalahan dalam lingkungan yang diamati.
- B. Masalah atau pokok permasalahan merupakan identifikasi dari latar belakang permasalahan.
- C. Tujuan dan manfaat penulisan laporan adalah uraian tujuan dan hal yang ingin dicapai mengenai penulisan karya tulis.

### 3. BAB II PEMBAHASAN

Pada bab II ini akan berisi :

- A. Akan dijelaskan mengenai profil studi kasus dari program beserta dengan flowchart.

- B. Penjelasan program akan dimuat gambar, tampilan dan penjelasan program serta penjelasan dari kode baris per baris.

#### **4. BAB III PENUTUP**

Pada bab III ini akan dijelaskan kesimpulan dan saran dari tugas yang dibuat.

- 5. Pada bagian akhir** akan berisi daftar pustaka berisi sumber bacaan ilmiah yang digunakan.

## **BAB II PEMBAHASAN**

### **A. Profil Studi Kasus**

Terdapat sebuah gudang penyimpanan alat olahraga yang menyimpan beberapa barang barang olahraga yaitu shuttlecock, matras, dan rompi futsal. Sistem gudang ini dapat melakukan penambahan stok barang, melihat stok yang tersedia, pengambilan barang dan juga pengembalian barang. Tiap tiap barang yang dikenai aksi memiliki mekanisme yang berbeda.

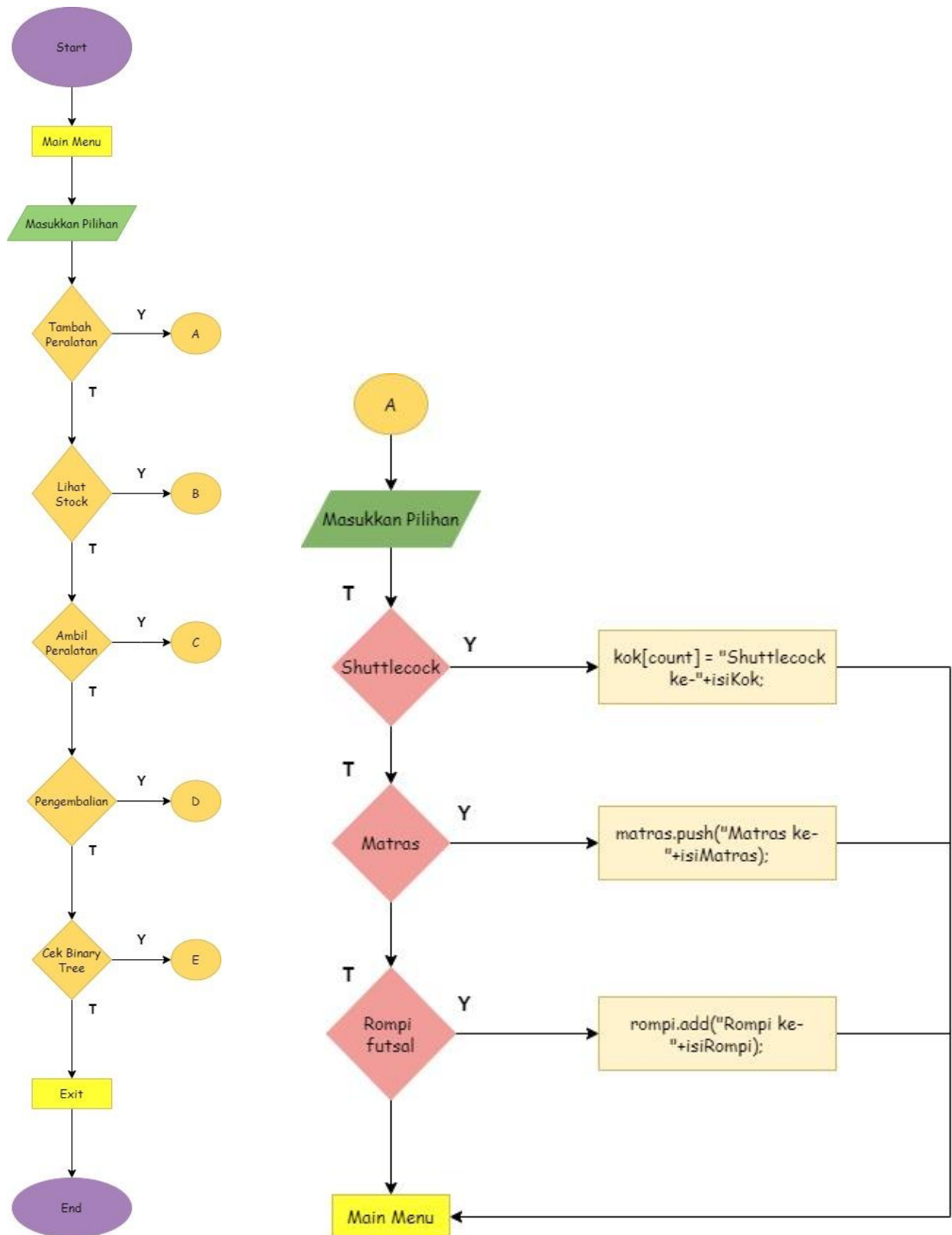
Ketika shuttlecock dikenai aksi, maka sistem akan memperlakukannya sebagai antrian atau queue. Maka ketika ada penambahan shuttlecock, barang yang baru masuk akan diletakkan di paling belakang lalu ketika ada pengambilan barang maka shuttlecock yang paling awal dimasukkan akan hilang dari antrian. Stok shuttlecock dibatasi hanya sebanyak 10 buah. Ketika melihat berapa stok shuttlecock, maka sistem akan menampilkan urutannya berdasarkan yang pertama kali diinputkan dan bila ada slot kosong pada antrian shuttlecock maka sistem akan menampilkannya sebagai slot null. Ketika dikenai aksi pengembalian, maka shuttlecock yang baru masuk akan berada di antrian paling belakang. Urutan pengembalian shuttlecock berdasarkan urutan meminjamannya.

Selanjutnya ketika matras dikenai aksi, maka sistem akan memperlakukannya sebagai tumpukan atau stack. Ketika ada penambahan stok pada matras, maka stok yang baru akan diletakkan disebelah kanan matras yang lama. Lalu ketika ada pengambilan maka matras yang terakhir dimasukkan akan hilang dari tumpukan. Kemudian ketika ada pengembalian maka matras yang baru datang akan diletakkan di sebelah kanan matras lainnya. Urutan pengembalian matras berdasarkan urutan meminjamannya. Ketika melihat stok matras, maka matras yang paling kanan adalah matras yang baru saja dimasukkan ke tumpukkan. Ketika stok matras kosong maka sistem akan menampilkannya sebagai list yang kosong.

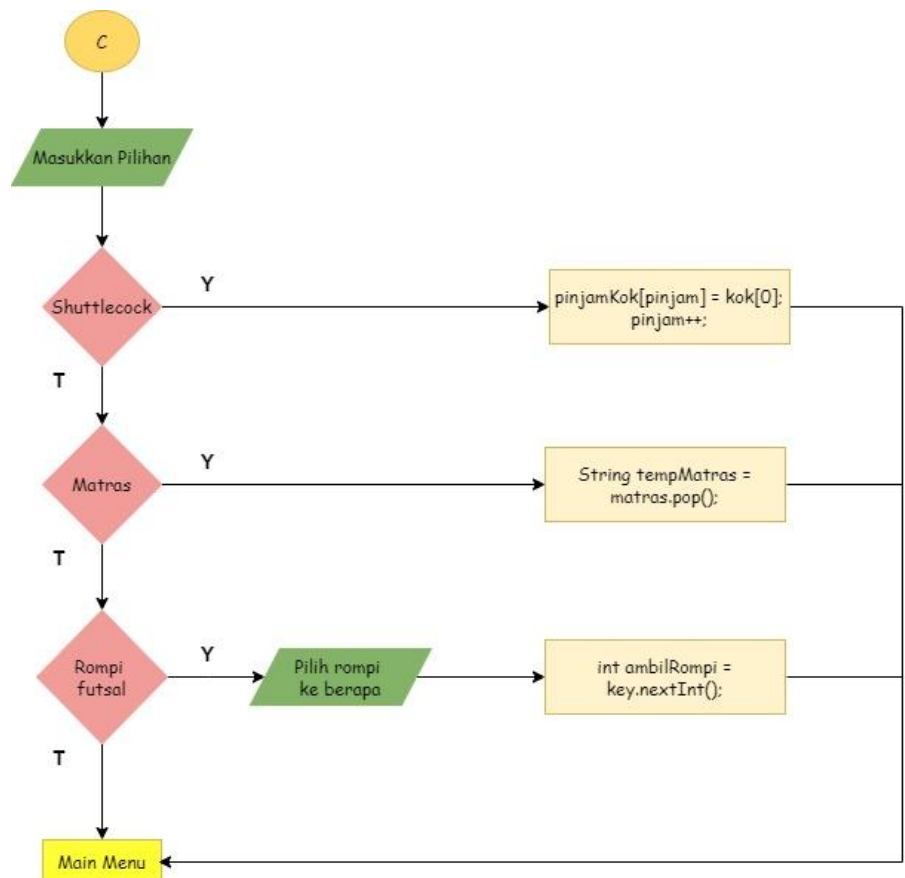
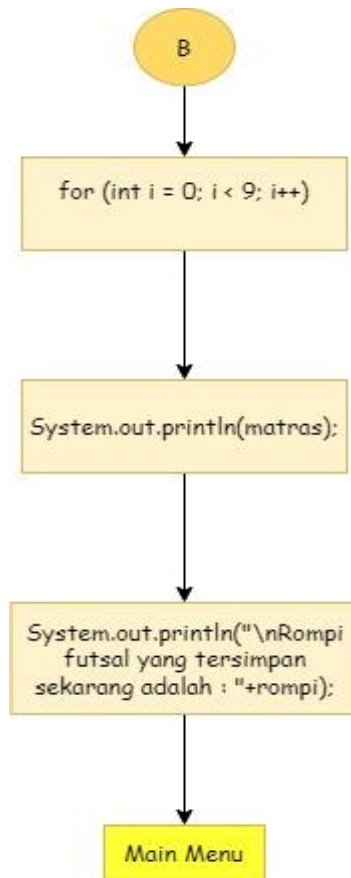
Kemudian ketika rompi futsal dikenai aksi, maka sistem akan memperlakukannya sebagai linked list. Karena diperlakukan sebagai linked list, maka user dapat mengambil rompi dengan urutan yang acak. Ketika ada penambahan stok rompi, maka stok baru tersebut akan diletakkan dibelakang stok lama. Lalu ketika ada pengambilan maka user akan ditanya ingin mengambil rompi urutan seberapa karena pengambilan dapat dilakukan secara acak. Ketika ada pengembalian maka user juga akan ditanya ingin meletakkan rompi di urutan seberapa lalu rompi tersebut akan diletakkan di urutan yang diinginkan user. Saat melihat stok rompi, maka sistem akan menampilkan urutannya sesuai kondisi terakhir yang dimodifikasi user. Bisa saja rompi yang terakhir dimasukkan sudah berada di awal list.

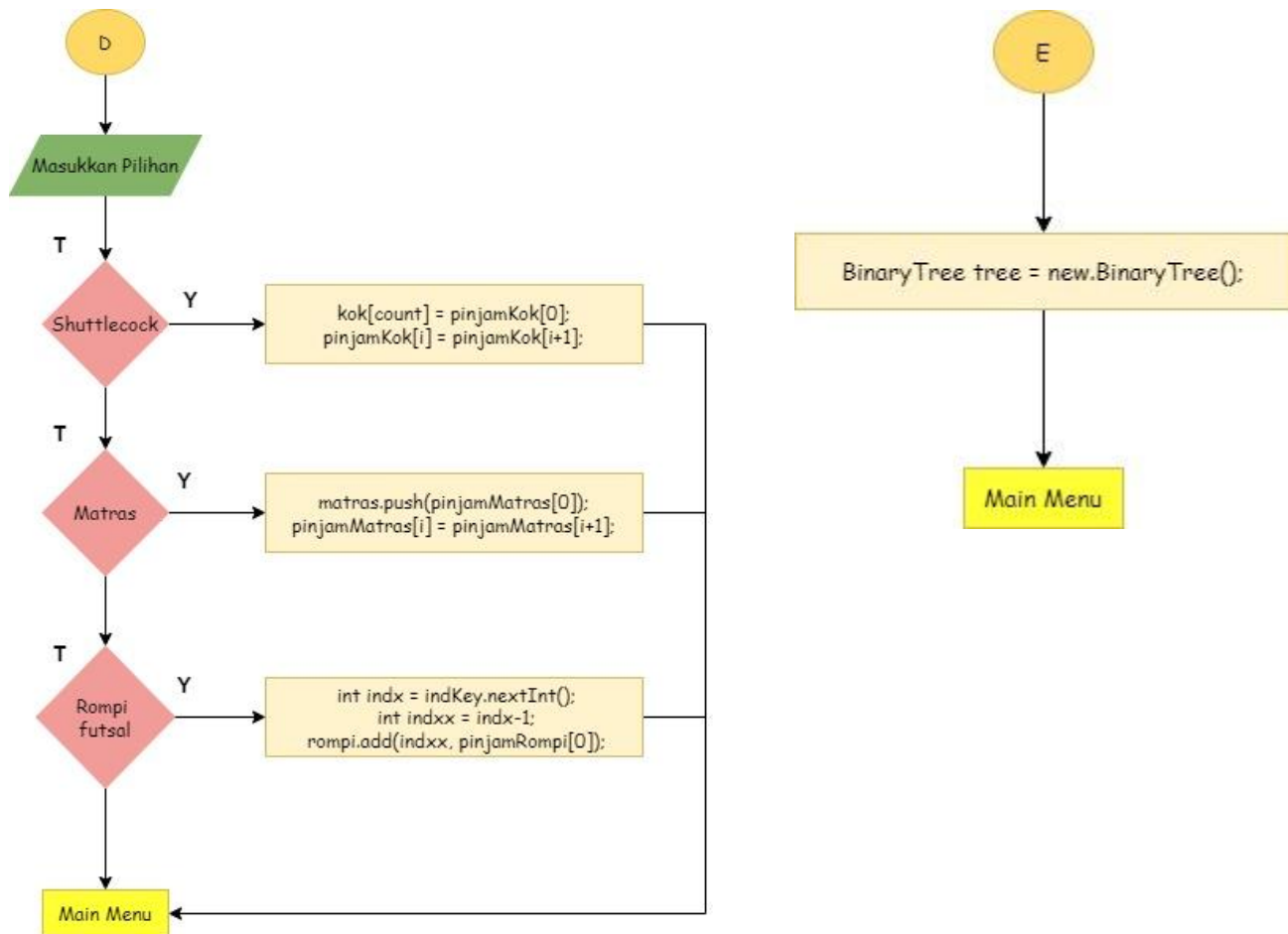
Dalam gudang penyimpanan alat olahraga ini, binary tree digunakan untuk mengasumsikan berapa banyak biasanya peminjaman di gudang ini. Untuk shuttlecock menggunakan pre order, matras menggunakan in order, dan rompi futsal menggunakan post order.

## B. Flowchart Program









Gambar flowchart di atas merupakan flowchart dari program Gudang Penyimpanan. Pada bagian tampilan utama terdapat pilihan beberapa menu dengan fungsi yang berbeda. Penggunaan simbol-simbol flowchart di atas menjabarkan sebuah langkah yang dapat dilakukan oleh user, berikut penjajarannya.

Hal pertama yang dapat dilakukan oleh seorang user ialah dengan memulainya, hal ini ditunjukkan dengan simbol “start”, dimana artinya sistem dimulai. Sebuah flowchart senantiasa diawali dengan “start” atau “mulai” serta di akhiri dengan “end” atau “selesai”. Saat pengguna memulainya, maka akan masuk pada halaman utama, hal tersebut digambarkan dengan persegi panjang yang artinya terdapat sebuah proses. Halaman utama menampilkan beberapa pilihan menu, diantaranya ialah tambah peralatan, lihat stok, ambil peralatan, pengembalian dan cek binary tree. Apabila pengguna memilih menu tambah peralatan maka akan menuju kepada shuttle cock, matress, dan rompi futsal. Disaat pengguna memilih menu stok, maka proses akan menuju ke size. Jika pengguna memilih ambil peralatan maka akan memproses kepada shuttle cock, matress, dan rompi futsal. Jika pengguna memilih menu pengembalian, maka akan menuju proses shuttle cock, matress, dan rompi futsal. Apabila pengguna tidak memilih menu apa-apa, itu artinya tidak akan terjadi sebuah proses. Pengguna dapat memilih Exit untuk mengakhiri program.

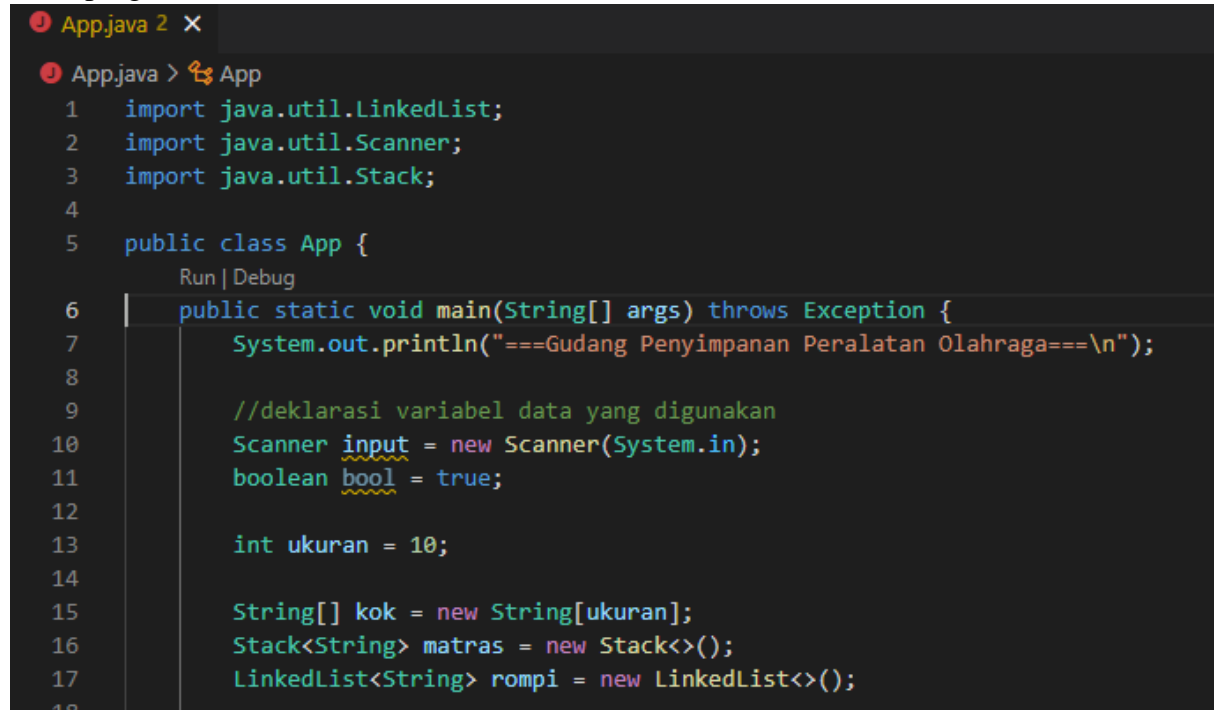
## C. Codingan dan Hasil Compile

### i. Codingan

#### 1. Pendeklarasian variabel

Pada program Gudang Penyimpanan Peralatan Olahraga ini, diperlukan pendeklarasian variabel guna komputer bisa memahami dengan pasti tipe data yang akan digunakan dalam variabel dan tipe data yang diwakili.

Code program :



```
App.java 2 X
App.java > App
1  import java.util.LinkedList;
2  import java.util.Scanner;
3  import java.util.Stack;
4
5  public class App {
    Run | Debug
6      public static void main(String[] args) throws Exception {
7          System.out.println("===Gudang Penyimpanan Peralatan Olahraga===\n");
8
9          //deklarasi variabel data yang digunakan
10         Scanner input = new Scanner(System.in);
11         boolean bool = true;
12
13         int ukuran = 10;
14
15         String[] kok = new String[ukuran];
16         Stack<String> matras = new Stack<>();
17         LinkedList<String> rompi = new LinkedList<>();
18     }
```

Pada kode diatas, dijelaskan :

- 1) input dengan tipe data scanner berfungsi sebagai masukan (input).
- 2) bool dengan tipe data boolean berfungsi sebagai perulangan yang apabila pilihan yang dimasukkan bernilai benar maka program akan berulang
- 3) ukuran dengan tipe data integer bernilai 10
- 4) kok dengan tipe data string berukuran 10 sebagai fungsi queue (antrian)
- 5) matras dengan tipe data string menggunakan fungsi stack (tumpukan)
- 6) rompi dengan tipe data string menggunakan fungsi linked list

Code program :

```
19 //variabel antrian
20 int count = 0;
21 int isiKok = 1;
22 int pinjam = 0;
23 String[] pinjamKok = new String[10];
24
25 //variabel stack
26 int isiMatras = 1;
27 int countt = 0;
28 String[] pinjamMatras = new String[10];
29
30 //variabel linkedlist
31 int isiRompi = 1;
32 int counttt = 0;
33 String[] pinjamRompi = new String[10];
```

Pada kode diatas, dijelaskan variabel antrian :

- 1) count dengan tipe data integer bernilai 0 berfungsi untuk menjadi index variabel string tempat menyimpan elemen yang dipinjam
- 2) isikok dengan tipe data integer bernilai 1 berfungsi untuk penunjang auto increment
- 3) pinjam dengan tipe data integer bernilai 0
- 4) pinjamkok dengan tipe data string bernilai 10 untuk menjadi index pengambilan elemen

## 2. Menu

Pada program menu terdapat pilihan yang terdiri dari menu tambah peralatan, lihat stok, ambil peralatan, pengembalian peralatan yang dipinjam, cek binary tree dan exit (keluar). Program ini menggunakan fungsi perulangan do while dimana dengan memindahkan posisi pemeriksaan ke akhir perulangan dengan kondisi menggunakan kondisi switch case. Artinya, lakukan dahulu sebuah perulangan, baru periksa apakah kondisi variabel counter sudah terpenuhi atau belum di akhir perulangan.

Code program:

```
37 //membuat menu
38 do {
39     System.out.println("\nPilihan Menu Tersedia : ");
40     System.out.println("1. Tambah Peralatan");
41     System.out.println("2. Lihat Stok");
42     System.out.println("3. Ambil Peralatan");
43     System.out.println("4. Pengembalian");
44     System.out.println("5. Cek Binary Tree");
45     System.out.println("6. Exit");
46
47     System.out.printf("Pilih Menu [1-6] : ");
48     int menu = input.nextInt();
49
50     System.out.println("=====\n");
```

Pada program diatas terdapat pilihan menu yang terdiri dari :

1) Tambah peralatan

Menu ini akan beroperasi pada perintah pilih menu 1-6 apabila input = 1 (switch case dengan kondisi case = 1). Menu ini berfungsi untuk menambah peralatan di Gudang olahraga.

Code program :

```
52         switch (menu) {
53             case 1:
54                 System.out.println("\nPilihan Barang : ");
55                 System.out.println("1. Shuttlecock");
56                 System.out.println("2. Matras");
57                 System.out.println("3. Rompi Futsal");
58
59                 System.out.printf("Pilih Menu [1-3] : ");
60                 int barang = input.nextInt();
61                 switch (barang) {
62                     case 1:
63                         kok[count] = "Shuttlecock ke-"+isiKok;
64
65                         System.out.println("\nShuttlecock ke-"+isiKok+" sudah ditambahkan ke gudang");
66                         isiKok++;
67                         count++;
68
69                         break;
70
71                     case 2:
72                         matras.push("Matras ke-"+isiMatras);
73
74                         System.out.println("\nMatras ke-"+isiMatras+" sudah ditambahkan ke gudang");
75                         isiMatras++;
76
77                         break;
78
79                     case 3:
80                         rompi.add("Rompi ke-"+isiRompi);
81
82                         System.out.println("\nRompi ke-"+isiRompi+" sudah ditambahkan ke gudang");
83                         isiRompi++;
84
85                         break;
86
87                     default:
88                         break;
89                 }
90                 break;
```

Program diatas akan menampilkan menu tambah peralatan yang terdiri dari shuttlecock, matras, dan rompi dengan kembali menggunakan kondisi Switch case.

- Case = 1 , maka shuttlecock akan ditambahkan ke Gudang pada antrian (queue) "kok". Count berfungsi untuk menentukan index. Untuk memasukkan elemen agar terjadi pengurutan dari nilai terkecil hingga terbesar dibutuhkan variabel "isikok". Isikok akan menjalankan operasi increment sehingga kok untuk indeks ke 0 adalah shuttlecock ke 1. Kemudian akan ditampilkan pesan bahwa shuttlecock ke 1 sudah dimasukkan ke Gudang. Begitu seterusnya hingga shuttlecock ke 10.
- Case = 2 , maka matras akan ditambahkan ke gudang pada tumpukan (stack) "matras". Variabel "isimatras" akan menjalankan operasi increment sehingga tumpukan akan tersusun.

- c. Case = 3 , maka rompi futsal akan ditambahkan ke gudang pada linkedlist “rompi”. Variabel “isirompi” akan menjalankan operasi increment sehingga tumpukan akan tersusun.
- 2) Lihat stok
- Menu ini akan beroperasi pada perintah pilih menu 1-6 apabila input = 2 (switch case dengan kondisi case = 2).

Code program :

```
92      case 2:
93          System.out.println("===Daftar Barang===");
94
95          System.out.println("Shuttlecock");
96          for (int i = 0; i<10; i++) {
97              System.out.println(kok[i]);
98          }
99
100         System.out.println("\nMatras");
101         System.out.println(matras);
102
103         System.out.println("\nRompi futsal yang tersimpan sekarang adalah : "+rompi);
104         break;
```

Pada program diatas , akan menampilkan seluruh daftar barang shuttlecock, matras, dan rompi futsal sesuai urutan setelah di tambahkan.

- 3) Ambil peralatan
- Menu ini akan beroperasi pada perintah pilih menu 1-6 apabila input = 3 (switch case dengan kondisi case = 3). menu ini berfungsi untuk meminjam peralatan yang ada di Gudang olahraga.

Code program :

```
107      case 3:
108          System.out.println("\nPilihan Barang : ");
109          System.out.println("1. Shuttlecock");
110          System.out.println("2. Matras");
111          System.out.println("3. Rompi Futsal");
112
113          System.out.printf("Pilih Menu [1-3] : ");
114          int barangg = input.nextInt();
115          switch (barangg) {
116              case 1:
117                  pinjamKok[pinjam] = kok[0];
118                  pinjam++;
119                  System.out.println("\n"+kok[0]+" sudah diambil dari gudang");
120                  for (int i = 0; i < ukuran-1; i++) {
121                      kok[i] = kok[i+1];
122                  }
123                  count--;
124
125                  break;
```

```

125         break;
126
127     case 2:
128         String tempMatras = matras.pop();
129         System.out.println("\n"+tempMatras+" sudah diambil dari gudang");
130
131         pinjamMatras[countt] = tempMatras;
132         countt++;
133         break;
134
135     case 3:
136         Scanner key = new Scanner(System.in);
137         System.out.printf("\nRompi ke-berapa yang ingin diambil[Angka] : ");
138         int ambilRompi = key.nextInt();
139         int index = ambilRompi-1;
140         System.out.printf("\nRompi ke-"+ambilRompi+" sudah diambil dari gudang");
141
142         pinjamRompi[counttt] = rompi.remove(index);
143         counttt++;
144
145         break;
146
147     default:
148         break;
149 }
150 break;

```

Program diatas akan menampilkan menu mengambil peralatan yang terdiri dari shuttlecock, matras, dan rompi dengan kembali menggunakan kondisi Switch case.

- a. Case = 1, maka shuttlecock akan diambil dari antrian (queue) “kok”. Karena konsep queue menggunakan konsep FIFO (*First In First Out*) dimana shuttlecock ke 1 sebagai indeks ke 0 akan diambil. Shuttlecock yang dipinjam akan disimpan pada variabel “simpankok”.
- b. Case = 2, maka matras akan diambil dari tumpukan (stack) “matras”. Karena konsep stack menggunakan konsep LIFO (*Last In First Out*) dimana matras yang berada pada tumpukan teratas akan terlebih dahulu diambil. Matras yang dipinjam akan disimpan pada variabel “pinjamMatras”.
- c. Case = 3, maka rompi akan diambil dari linkedList dengan variabel “rompi”. Letak rompi yang diambil dapat dipilih secara acak. rompi yang dipinjam akan disimpan pada variabel “pinjamRompi”.

#### 4) Pengembalian

Menu ini akan beroperasi pada perintah pilih menu 1-6 apabila input = 4 (switch case dengan kondisi case = 4). Menu ini berfungsi untuk mengembalikan peralatan yang di pinjam.

### Code Program :

```
152 case 4:
153 System.out.println("\nPilihan Barang : ");
154 System.out.println("1. Shuttlecock");
155 System.out.println("2. Matras");
156 System.out.println("3. Rompi Futsal");
157
158 System.out.printf("Pilih Menu [1-3] : ");
159 int baranggg = input.nextInt();
160 switch (baranggg) {
161     case 1:
162         kok[count] = pinjamKok[0];
163         System.out.println("\n"+kok[count]+" sudah dikembalikan ke gudang");
164         count++;
165         for (int i = 0; i < ukuran-1; i++) {
166             pinjamKok[i] = pinjamKok[i+1];
167         }
168         pinjam--;
169         break;
170
171     case 2:
172         matras.push(pinjamMatras[0]);
173         System.out.println("\n"+pinjamMatras[0]+" sudah dikembalikan ke gudang");
174         for (int i = 0; i < 9; i++) {
175             pinjamMatras[i] = pinjamMatras[i+1];
176         }
177         break;
178
179     case 3:
180         Scanner indKey = new Scanner(System.in);
181         System.out.printf("\nIngin meletakkan rompi pada urutan[1-10] : ");
182         int indx = indKey.nextInt();
183         int indxx = indx-1;
184
185         rompi.add(indxx, pinjamRompi[0]);
186
187         for (int i = 0; i < 9; i++) {
188             pinjamRompi[i] = pinjamRompi[i+1];
189         }
190
191         break;
192
193     default:
194         break;
195 }
196
197 break;
198
```

Program diatas akan menampilkan menu mengembalikan peralatan yang terdiri dari shuttlecock, matras, dan rompi dengan kembali menggunakan kondisi Switch case.

- Case = 1, maka shuttlecock yang dipinjam dan disimpan pada variabel “pinjamKok” akan dikembalikan ke antrian (queue) “kok”. Shuttlecock yang dikembalikan akan menjadi antrian terbawah.
- Case = 2, maka matras yang dipinjam dan disimpan pada variabel “pinjamMatras” akan dikembalikan ke tumpukan (stack) “matras”. Shuttlecock yang dikembalikan akan menjadi tumpukan teratas.
- Case = 3, maka rompi futsal yang dipinjam dan disimpan pada variabel “pinjamRompi” akan dikembalikan ke linkedlist dengan variabel “rompi”.

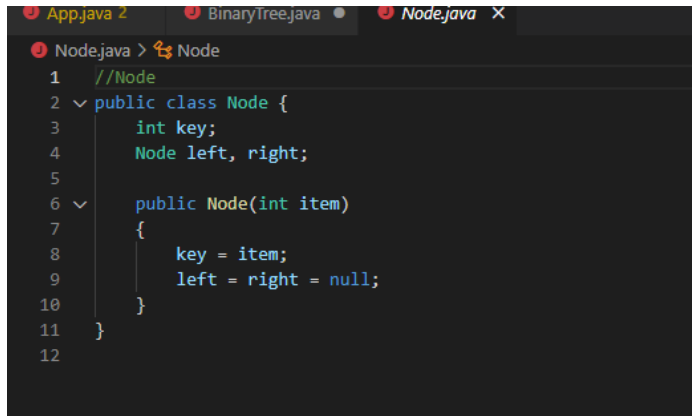


5) Cek Binary tree

Menu ini akan beroperasi pada perintah pilih menu 1-6 apabila input = 5 (switch case dengan kondisi case = 5).

a. Pendeklarasian node

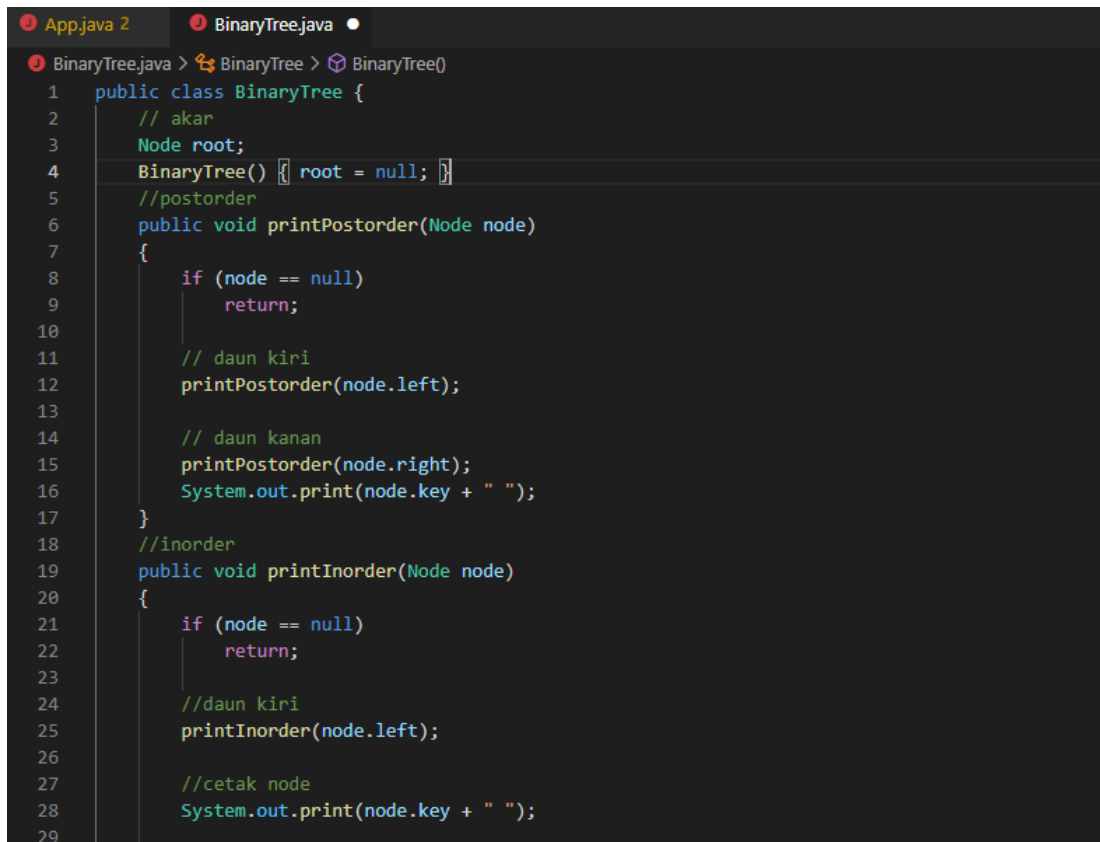
Node ini hanya untuk pendeklarasian yang akan digunakan dalam file binary tree  
Code Program :

A screenshot of a code editor showing the Node.java file. The code defines a Node class with attributes key, left, and right, and a constructor Node(int item) that initializes these attributes.

```
1 //Node
2 public class Node {
3     int key;
4     Node left, right;
5
6     public Node(int item)
7     {
8         key = item;
9         left = right = null;
10    }
11 }
12
```

b. Binary tree

Code Program :

A screenshot of a code editor showing the BinaryTree.java file. The code defines a BinaryTree class with a root attribute and methods for postorder and inorder traversal.

```
1 public class BinaryTree {
2     // akar
3     Node root;
4     BinaryTree() { root = null; }
5
6     //postorder
7     public void printPostorder(Node node)
8     {
9         if (node == null)
10            return;
11
12        // daun kiri
13        printPostorder(node.left);
14
15        // daun kanan
16        printPostorder(node.right);
17        System.out.print(node.key + " ");
18    }
19
20    //inorder
21    public void printInorder(Node node)
22    {
23        if (node == null)
24            return;
25
26        //daun kiri
27        printInorder(node.left);
28
29        //cetak node
30        System.out.print(node.key + " ");
31    }
32 }
```

```

30         //daun kanan
31         printInorder(node.right);
32     }
33
34     //preorder
35     public void printPreorder(Node node)
36     {
37         if (node == null)
38             return;
39
40         //cetak data awal
41         System.out.print(node.key + " ");
42         printPreorder(node.left);
43         printPreorder(node.right);
44     }
45     public void printPostorder() { printPostorder(root); }
46     public void printInorder() { printInorder(root); }
47     public void printPreorder() { printPreorder(root); }
48 }

```

Pada kasus ini jenis binary tree yang diterapkan adalah binary Tree Traversal, dimana nilai nilai yang diletakkan pada masing masing node merupakan data berapa banyak sebuah barang di pinjam dari masing masing variabel shuttlecock, matras dan rompi futsal.

#### 6) Exit

Menu ini akan beroperasi pada perintah pilih menu 1-6 apabila input = 6 (switch case dengan kondisi case = 6). Menu ini berfungsi untuk mengakhiri program yang dijalankan.

Code Program :

```

223         case 6:
224             System.exit(0);
225             break;
226
227         default:
228             System.out.println("Masukan inputan menu yang tersedia pada pilihan!");
229             break;
230     }
231 }while(bool=true);
232 }
233 }

```

## ii. Hasil Run Program

### 1. Menu Tambah Peralatan

#### 1) Shuttlecock sebanyak 4 buah

```
===Gudang Penyimpanan Peralatan Olahraga===

Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 1
=====

Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 1

Shuttlecock ke-1 sudah ditambahkan ke gudang

Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 1
=====

Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 1

Shuttlecock ke-2 sudah ditambahkan ke gudang

Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 1
=====
```

```
Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 1

Shuttlecock ke-3 sudah ditambahkan ke gudang

Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 1
=====

Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 1

Shuttlecock ke-4 sudah ditambahkan ke gudang
```

2) Matras sebanyak 4 buah

```
Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 1
=====

Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 2

Matras ke-1 sudah ditambahkan ke gudang

Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 1
=====

Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 2

Matras ke-2 sudah ditambahkan ke gudang

Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 1
=====

Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 2
```

```
Matras ke-3 sudah ditambahkan ke gudang

Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 1
=====

Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 2

Matras ke-4 sudah ditambahkan ke gudang
```

### 3) Rompi Futsal sebanyak 4 buah

```
Pilihan Barang :  
1. Shuttlecock  
2. Matras  
3. Rompi Futsal  
Pilih Menu [1-3] : 3
```

Rompi ke-1 sudah ditambahkan ke gudang

```
Pilihan Menu Tersedia :  
1. Tambah Peralatan  
2. Lihat Stok  
3. Ambil Peralatan  
4. Pengembalian  
5. Cek Binary Tree  
6. Exit  
Pilih Menu [1-6] : 1  
=====
```

```
Pilihan Barang :  
1. Shuttlecock  
2. Matras  
3. Rompi Futsal  
Pilih Menu [1-3] : 3
```

Rompi ke-2 sudah ditambahkan ke gudang

```
Pilihan Menu Tersedia :  
1. Tambah Peralatan  
2. Lihat Stok  
3. Ambil Peralatan  
4. Pengembalian  
5. Cek Binary Tree  
6. Exit  
Pilih Menu [1-6] : 1  
=====
```

```
Pilihan Barang :  
1. Shuttlecock  
2. Matras  
3. Rompi Futsal  
Pilih Menu [1-3] : 3
```

Rompi ke-3 sudah ditambahkan ke gudang

```
Pilihan Menu Tersedia :  
1. Tambah Peralatan  
2. Lihat Stok  
3. Ambil Peralatan  
4. Pengembalian  
5. Cek Binary Tree  
6. Exit  
Pilih Menu [1-6] : 1  
=====
```

```
Pilihan Barang :  
1. Shuttlecock  
2. Matras  
3. Rompi Futsal  
Pilih Menu [1-3] : 3
```

Rompi ke-4 sudah ditambahkan ke gudang

## 2. Menu lihat stok

Menu ini akan menampilkan antrian shuttlecock, tumpukan matras dan linkedlist dari rompi futsal setelah ditambahkan sebelumnya.

```
Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 2
=====

===Daftar Barang===
Shuttlecock
Shuttlecock ke-1
Shuttlecock ke-2
Shuttlecock ke-3
Shuttlecock ke-4
null
null
null
null
null
null

Matras
[Matras ke-1, Matras ke-2, Matras ke-3, Matras ke-4]

Rompi futsal yang tersimpan sekarang adalah : [Rompi ke-1, Rompi ke-2, Rompi ke-3, Rompi ke-4]
```

## 3. Menu ambil Peralatan

1) Ambil 1 shuttlecock, yang berarti bahwa shuttlecock pertama yang akan diambil

```
Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 3
=====

Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 1

Shuttlecock ke-1 sudah diambil dari gudang
```

- 2) Ambil 1 matras, yang berarti bahwa matras paling atas yang akan diambil

```
Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 3
=====

Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 2

Matras ke-4 sudah diambil dari gudang
```

- 3) Ambil 1 rompi futsal yang terletak pada urutan ke 2.

```
Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 3
=====

Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 3

Rompi ke-berapa yang ingin diambil[Angka] : 2

Rompi ke-2 sudah diambil dari gudang
```

#### 4. Menu pengembalian

- 1) Mengembalikan shuttlecock ke-1 yang telah dipinjam ke Gudang

```
Rompi ke-2 sudah diambil dari gudang
Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 4
=====

Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 1

Shuttlecock ke-1 sudah dikembalikan ke gudang
```

- 2) Mengembalikan matras ke-4 yang telah dipinjam ke Gudang

```
Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 4
=====

Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 2

Matras ke-4 sudah dikembalikan ke gudang
```

- 3) Mengembalikan rompi futsal ke-2 yang telah dipinjam ke Gudang pada urutan ke-2

```
Pilihan Barang :
1. Shuttlecock
2. Matras
3. Rompi Futsal
Pilih Menu [1-3] : 3

Ingin meletakkan rompi pada urutan[1-10] : 2
```

5. Menu cek Binary tree

```
Pilihan Menu Tersedia :
1. Tambah Peralatan
2. Lihat Stok
3. Ambil Peralatan
4. Pengembalian
5. Cek Binary Tree
6. Exit
Pilih Menu [1-6] : 5
=====

PreOrder  [Shuttlecock]    : 1 2 4 5 3 6 7
inOrder   [Matras]         : 4 2 5 1 6 3 7
PostOrder [Rompi futsal]   : 4 5 2 6 7 3 1
```



## **BAB III PENUTUP**

### **A. Kesimpulan**

Program Gudang Penyimpanan ini dibuat dalam rangka memudahkan dan mengifisienkan pekerjaan manusia yang awalnya melalui proses manual menjadi lebih teromatisasi, sehingga segala sesuatu yang dilakukan dapat menghemat waktu serta biaya operasional. Dalam hal pengambilan dan pengembalian barang, kita dapat mendata dengan berbagai metode seperti mengambil barang dari posisi awal, dari posisi akhir, secara terurut dari yang pertama dikembalikan, dan juga terurut dari yang terakhir dikembalikan.

### **B. Saran**

Penulis menyarankan untuk mengembangkan berbagai metode pengambilan dan pengembalian barang, sehingga dalam mengambil dan mengembalikan barang untuk memenuhi kebutuhan peminjam, gudang memiliki berbagai metode lainnya yang dapat dioperasikan.

## **REFERENSI**

Bab I pendahuluan 1.1 Latar Belakang - Universitas diponegoro. (n.d.). Retrieved March 29, 2022, from [http://eprints.undip.ac.id/51297/2/Bab\\_I.pdf](http://eprints.undip.ac.id/51297/2/Bab_I.pdf)