

# LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA



2022

**Praktikan**

2141720183

RIDWAN CAESAR RIZQI KARISMA BIWARNI

TI 1C



## Daftar Isi [harus diupdate setiap ada perubahan]

<b>PRAKTIKUM 1 .....</b>	<b>4</b>
LANGKAH 1.....	4
LANGKAH 2.....	4
LANGKAH 3.....	4
LANGKAH 4.....	5
LANGKAH 5.....	5
LANGKAH 6.....	6
LANGKAH 7.....	6
LANGKAH 8.....	6
LANGKAH 9.....	6
LANGKAH 10.....	7
LANGKAH 11.....	7
LANGKAH 12.....	7
LANGKAH 13.....	7
LANGKAH 14.....	8
LANGKAH 15.....	8
LANGKAH 16.....	8
LANGKAH 17.....	8
VERIFIKASI HASIL PERCOBAAN .....	9
<i>Pertanyaan.....</i>	<i>9</i>
<i>Jawaban.....</i>	<i>9</i>
<b>PRAKTIKUM 2 .....</b>	<b>12</b>
LANGKAH 1.....	12
LANGKAH 2.....	12
LANGKAH 3.....	12
LANGKAH 4.....	13
LANGKAH 5.....	13
LANGKAH 6.....	14
LANGKAH 7.....	14
LANGKAH 8.....	15
LANGKAH 9.....	15
LANGKAH 10.....	16
LANGKAH 11.....	16
LANGKAH 12.....	16



## PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

VERIFIKASI HASIL PERCOBAAN .....	16
<i>Pertanyaan</i> .....	17
<i>Jawaban</i> .....	17
<b>TUGAS .....</b>	<b>18</b>
SOAL NOMOR 1.....	18
SOAL NOMOR 2.....	18
JAWABAN NOMOR 1.....	18
<i>Source code yang dimodifikasi pada class StackRidwan</i> .....	18
<i>Source code yang dimodifikasi pada class StackMainRidwan</i> .....	19
<i>Output</i> .....	20
JAWABAN NOMOR 2.....	21
<i>Source Code StrukBelanjaRidwan</i> .....	21
<i>Source Code StackStrukRidwan</i> .....	21
<i>Source Code StrukMain</i> .....	23
<i>Output</i> .....	24

## Praktikum 1

Pada percobaan ini, kita akan membuat program yang mengilustrasikan tumpukan pakaian yang disimpan ke dalam stack. Karena sebuah pakaian mempunyai beberapa informasi, maka implementasi Stack dilakukan dengan menggunakan array of object untuk mewakili setiap elemennya.

### Langkah 1

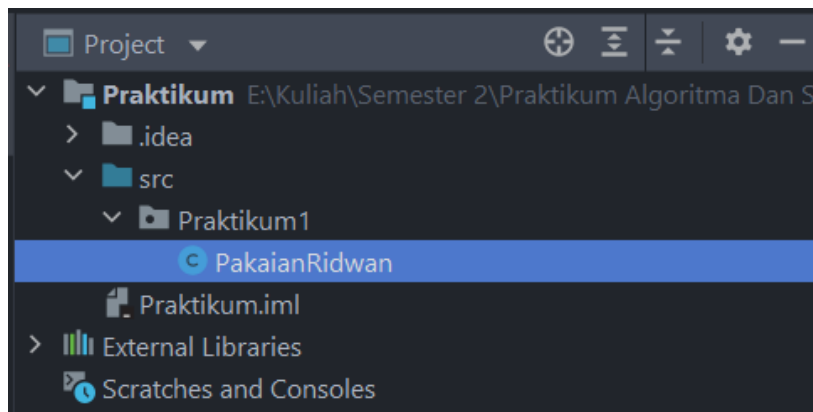
Perhatikan Diagram Class Pakaian berikut ini:

Pakaian
jenis: String warna: String merk: String ukuran: String harga: double
Pakaian(jenis: String, warna: String, merk: String, ukuran: String, harga: double)

Berdasarkan diagram class tersebut, akan dibuat program class Pakaian dalam Jawa

### Langkah 2

Buat package dengan nama Praktikum1, kemudian buat class baru dengan nama Pakaian.



### Langkah 3

Tambahkan atribut-atribut Pakaian seperti pada Class Diagram Pakaian, kemudian tambahkan pula konstruktornya seperti gambar berikut ini.

```
1 package Praktikum1;
2
3 public class PakaianRidwan {
4     String jenis, warna, merk, ukuran;
5     double harga;
6
7     PakaianRidwan(String jenis, String warna, String merk, String ukuran, double harga) {
8         this.jenis = jenis;
9         this.warna = warna;
10        this.merk = merk;
11        this.ukuran = ukuran;
12        this.harga = harga;
13    }
14 }
```

## Langkah 4

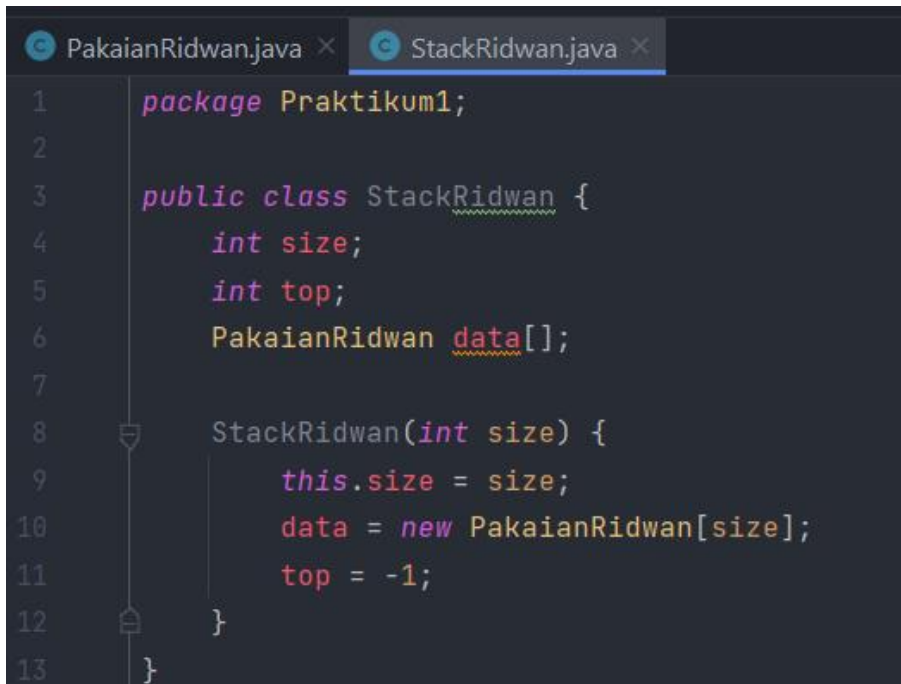
Setelah membuat class Pakaian, selanjutnya perlu dibuat class Stack yang berisi atribut dan method sesuai diagram Class Stack berikut ini:

Stack
size: int
top: int
data[]: Pakaian
Stack(size: int)
IsEmpty(): boolean
IsFull(): boolean
push(): void
pop(): void
peek(): void
print(): void
clear(): void

Keterangan: Tipe data pada variabel data menyesuaikan dengan data yang akan disimpan di dalam Stack. Pada praktikum ini, data yang akan disimpan merupakan array of object dari Pakaian, sehingga tipe data yang digunakan adalah Pakaian.

## Langkah 5

Buat class baru dengan nama Stack. Kemudian tambahkan atribut dan konstruktor seperti gambar berikut ini.



```

1  package Praktikum1;
2
3  public class StackRidwan {
4      int size;
5      int top;
6      PakaianRidwan data[];
7
8      StackRidwan(int size) {
9          this.size = size;
10         data = new PakaianRidwan[size];
11         top = -1;
12     }
13 }

```

### Langkah 6

Buat method `IsEmpty` bertipe boolean yang digunakan untuk mengecek apakah stack kosong.

```
14      public boolean IsEmpty() {  
15          if (top == -1) {  
16              return true;  
17          } else {  
18              return false;  
19          }  
20      }
```

### Langkah 7

Buat method `IsFull` bertipe boolean yang digunakan untuk mengecek apakah stack sudah terisi penuh.

```
22      public boolean IsFull() {  
23          if (top == size - 1) {  
24              return true;  
25          } else {  
26              return false;  
27          }  
28      }
```

### Langkah 8

Buat method `push` bertipe void untuk menambahkan isi elemen stack dengan parameter `pkn` yang berupa object `Pakaian`

```
30      public void push(PakaianRidwan pkn) {  
31          if (!IsFull()) {  
32              top++;  
33              data[top] = pkn;  
34          } else {  
35              System.out.println("Isi stack penuh!");  
36          }  
37      }
```

### Langkah 9

Buat method `Pop` bertipe void untuk mengeluarkan isi elemen stack. Karena satu elemen stack terdiri dari beberapa informasi (jenis, warna, merk, ukuran, dan harga), maka ketika mencetak data juga perlu ditampilkan semua informasi tersebut

```

39     public void pop() {
40         if (!IsEmpty()) {
41             PakaianRidwan x = data[top];
42             top--;
43             System.out.println("Data yang keluar: " + x.jenis + " " + x.warna +
44                 " " + x.merk + " " + x.ukuran + " " + x.harga);
45         } else {
46             System.out.println("Stack masih kosong");
47         }
48     }

```

### Langkah 10

Buat method peek bertipe void untuk memeriksa elemen stack pada posisi paling atas

```

50     public void peek() {
51         System.out.println("Elemen teratas: " + data[top].jenis + " " + data[top].warna +
52             " " + data[top].merk + " " + data[top].ukuran + " " + data[top].harga);
53     }

```

### Langkah 11

Buat method print bertipe void untuk menampilkan seluruh elemen pada stack

```

55     public void print() {
56         System.out.println("Isi stack: ");
57         for (int i = top; i >= 0; i--) {
58             System.out.println(data[i].jenis + " " + data[i].warna +
59                 " " + data[i].merk + " " + data[i].ukuran + " " + data[i].harga);
60         }
61     }

```

### Langkah 12

Buat method clear bertipe void untuk menghapus seluruh isi stack.

```

63     public void clear() {
64         if (!IsEmpty()) {
65             for (int i = top; i >= 0; i--) {
66                 top--;
67             }
68             System.out.println("Stack sudah dikosongkan");
69         } else {
70             System.out.println("Stack masih kosong");
71         }
72     }

```

### Langkah 13

Selanjutnya, buat class baru dengan nama StackMain. Buat fungsi main, kemudian lakukan instansiasi objek dari class Stack dengan nama stk dan nilai parameternya adalah 5.

```

5     public class StackMainRidwan {
6         public static void main(String[] args) {
7             StackRidwan stk = new StackRidwan( size: 5);

```

## Langkah 14

Deklarasikan Scanner dengan nama sc

```
7      StackRidwan stk = new StackRidwan( size: 5);
8      Scanner sc = new Scanner(System.in);
9  }
10 }
```

## Langkah 15

Tambahkan kode berikut ini untuk menerima input data Pakaian, kemudian semua informasi tersebut dimasukkan ke dalam stack

```
10      char pilih;
11      do {
12          System.out.print("Jenis: ");
13          String jenis = sc.nextLine();
14          System.out.print("Warna: ");
15          String warna = sc.nextLine();
16          System.out.print("Merk: ");
17          String merk = sc.nextLine();
18          System.out.print("Ukuran: ");
19          String ukuran = sc.nextLine();
20          System.out.print("Harga: ");
21          double harga = sc.nextDouble();
22
23          PakaianRidwan p = new PakaianRidwan(jenis, warna, merk, ukuran, harga);
24          System.out.print("Apakah anda akan menambahkan data baru ke stack (y/n) ? ");
25          pilih = sc.next().charAt(0);
26          sc.nextLine();
27          stk.push(p);
28      } while (pilih == 'y');
```

Catatan: sintaks `sc.nextLine()` sebelum sintaks `stk.push(p)` digunakan untuk mengabaikan karakter new line

## Langkah 16

Lakukan pemanggilan method print, method pop, dan method peek dengan urutan sebagai berikut.

```
29      stk.print();
30      stk.pop();
31      stk.peek();
32      stk.print();
```

## Langkah 17

Compile dan jalankan class StackMain, kemudian amati hasilnya.



## Verifikasi Hasil Percobaan

```

"C:\Users\Asus TUF DT\.jdk\openjdk-17.0.2\bin\java.exe" "-javaagent:E:\Da
Jenis: Kaos
Warna: Hitam
Merk: Nevada
Ukuran: M
Harga: 85000
Apakah anda akan menambahkan data baru ke stack (y/n) ? y
Jenis: Kemeja
Warna: Putih
Merk: Styves
Ukuran: XL
Harga: 127000
Apakah anda akan menambahkan data baru ke stack (y/n) ? y
Jenis: Celana
Warna: Biru
Merk: Levis
Ukuran: L
Harga: 189500
Apakah anda akan menambahkan data baru ke stack (y/n) ? n
Isi stack:
Celana Biru Levis L 189500.0
Kemeja Putih Styves XL 127000.0
Kaos Hitam Nevada M 85000.0
Data yang keluar: Celana Biru Levis L 189500.0
Elemen teratas: Kemeja Putih Styves XL 127000.0
Isi stack:
Kemeja Putih Styves XL 127000.0
Kaos Hitam Nevada M 85000.0

```

## Pertanyaan

1. Berapa banyak data pakaian yang dapat ditampung di dalam stack? Tunjukkan potongan kode program untuk mendukung jawaban Anda tersebut!
2. Perhatikan class StackMain, pada saat memanggil fungsi push, parameter yang dikirimkan adalah p. Data apa yang tersimpan pada variabel p tersebut?

```
stk.push(p);
```

3. Apakah fungsi penggunaan do-while yang terdapat pada class StackMain?
4. Modifikasi kode program pada class StackMain sehingga pengguna dapat memilih operasi-operasi pada stack (push, pop, peek, atau print) melalui pilihan menu program dengan memanfaatkan kondisi IF-ELSE atau SWITCH-CASE!

## Jawaban

1. Banyak data pakaian yang disimpan pada stack ada 5.  
Pada kode dibawah ini memanggil konstruktor StackRidwan dengan parameter bernilai 5.

```

6 public static void main(String[] args) {
7     StackRidwan stk = new StackRidwan( size: 5);

```

2. Pada variabel data p disimpan data dengan tipe data PakaianRidwan yang memiliki attribute Jenis, Warna, Merk, Ukuran, Harga.
3. Penggunaan do while agar perulangan setidaknya dilakukan minimal satu kali.
4. Source code class StackMainRidwan

```

10      while (true) {
11          System.out.println("=====");
12          System.out.println("| [1] push      |");
13          System.out.println("| [2] pop       |");
14          System.out.println("| [3] peek      |");
15          System.out.println("| [4] print     |");
16          System.out.println("| [5] clear     |");
17          System.out.println("=====");
18          System.out.print("Masukkan angka menu yang anda pilih : ");
19          int menu = input.nextInt();
20          switch (menu) {
21              case 1 -> {
22                  char pilih;
23                  do {
24                      System.out.print("Jenis: ");
25                      String jenis = sc.nextLine();
26                      System.out.print("Warna: ");
27                      String warna = sc.nextLine();
28                      System.out.print("Merk: ");
29                      String merk = sc.nextLine();
30                      System.out.print("Ukuran: ");
31                      String ukuran = sc.nextLine();
32                      System.out.print("Harga: ");
33                      double harga = sc.nextDouble();
34
35                      PakaianRidwan p = new PakaianRidwan(jenis, warna, merk, ukuran, harga);
36                      System.out.print("Apakah anda akan menambahkan data baru ke stack (y/n) ? ");
37                      pilih = sc.next().charAt(0);
38                      sc.nextLine();
39                      stk.push(p);
40                  } while (pilih == 'y');
41              }
42
43              case 2 -> stk.pop();
44              case 3 -> stk.peek();
45              case 4 -> stk.print();
46              case 5 -> stk.clear();
47              default -> System.out.println("Masukkan salah");
48          }
49          System.out.println("apakah ada keperluan lagi? (y/n) ");
50          char pilih = sc.next().charAt(0);
51          sc.nextLine();
52          if (pilih != 'y') {
53              break;
54          }
55      }

```

Output

```
Run: StackMainRidwan x
"C:\Users\Asus TUF DT\.jdk\openjdk-17.0.2\bin\java.exe"
=====
| [1] push      |
| [2] pop       |
| [3] peek      |
| [4] print     |
| [5] clear     |
=====
Masukkan angka menu yang anda pilih : 1
Jenis: kaos
Warna: hitam
Merk: nevada
Ukuran: m
Harga: 85000
Apakah anda akan menambahkan data baru ke stack (y/n) ?
apakah ada keperluan lagi? (y/n)
y
=====
| [1] push      |
| [2] pop       |
| [3] peek      |
| [4] print     |
| [5] clear     |
=====
Masukkan angka menu yang anda pilih : 3
Elemen teratas: kaos hitam nevada m 85000.0

apakah ada keperluan lagi? (y/n)
y
=====
| [1] push      |
| [2] pop       |
| [3] peek      |
| [4] print     |
| [5] clear     |
=====
Masukkan angka menu yang anda pilih : 2
Data yang keluar: kaos hitam nevada m 85000.0
apakah ada keperluan lagi? (y/n)
y
=====
| [1] push      |
| [2] pop       |
| [3] peek      |
| [4] print     |
| [5] clear     |
=====
Masukkan angka menu yang anda pilih : 4
Isi stack:
apakah ada keperluan lagi? (y/n)
```

## Praktikum 2

Pada percobaan ini, kita akan membuat program untuk melakukan konversi notasi infix menjadi notasi postfix.

### Langkah 1

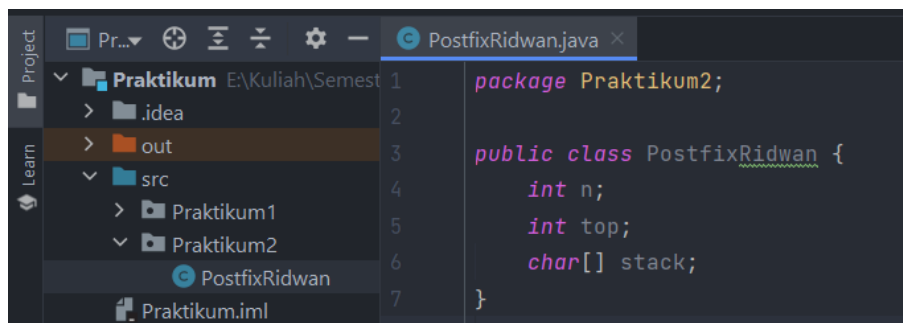
Perhatikan Diagram Class berikut ini:

Postfix
n: int top: int stack: char[]
Postfix(total: int) push(c: char): void pop(): void IsOperand(c: char): boolean IsOperator(c: char): boolean derajat(c: char): int konversi(Q: String): string

Berdasarkan diagram class tersebut, akan dibuat program class Postfix dalam Java.

### Langkah 2

Buat package dengan nama Praktikum2, kemudian buat class baru dengan nama Postfix. Tambahkan atribut n, top, dan stack sesuai diagram class Postfix tersebut.



### Langkah 3

Tambahkan pula konstruktor berparameter seperti gambar berikut ini.

```
PostfixRidwan.java x
1  package Praktikum2;
2
3  public class PostfixRidwan {
4      int n;
5      int top;
6      char[] stack;
7
8      public PostfixRidwan(int total) {
9          n = total;
10         top = -1;
11         stack = new char[n];
12         push('(');
13     }
14 }
```

#### Langkah 4

Buat method push dan pop bertipe void

```
15     public void push(char c) {
16         top++;
17         stack[top] = c;
18     }
19
20     public char pop() {
21         char item = stack[top];
22         top--;
23         return item;
24     }
```

#### Langkah 5

Buat method IsOperand dengan tipe boolean yang digunakan untuk mengecek apakah elemen data berupa operand.

```
26 public boolean IsOperand(char c) {  
27     if ((c >= 'A' && c <= 'Z') || (c >= 'a' && c <= 'z') ||  
28         (c >= '0' && c <= '9') || (c == ' ') || c == '.') {  
29         return true;  
30     } else {  
31         return false;  
32     }  
33 }  
34 }
```

### Langkah 6

Buat method IsOperator dengan tipe boolean yang digunakan untuk mengecek apakah elemen data berupa operator.

```
36 public boolean IsOperator(char c) {  
37     if (c == '^' || c == '%' || c == '/' || c == '*' || c == '-' ||  
38         c == '+') {  
39         return true;  
40     } else {  
41         return false;  
42     }  
43 }
```

### Langkah 7

Buat method derajat yang mempunyai nilai kembalian integer untuk menentukan derajat operator.

```
45 public int derajat(char c) {  
46     switch (c) {  
47         case '^':  
48             return 3;  
49         case '%':  
50             return 2;  
51         case '/':  
52             return 2;  
53         case '*':  
54             return 2;  
55         case '-':  
56             return 1;  
57         case '+':  
58             return 1;  
59         default:  
60             return 0;  
61     }  
62 }
```

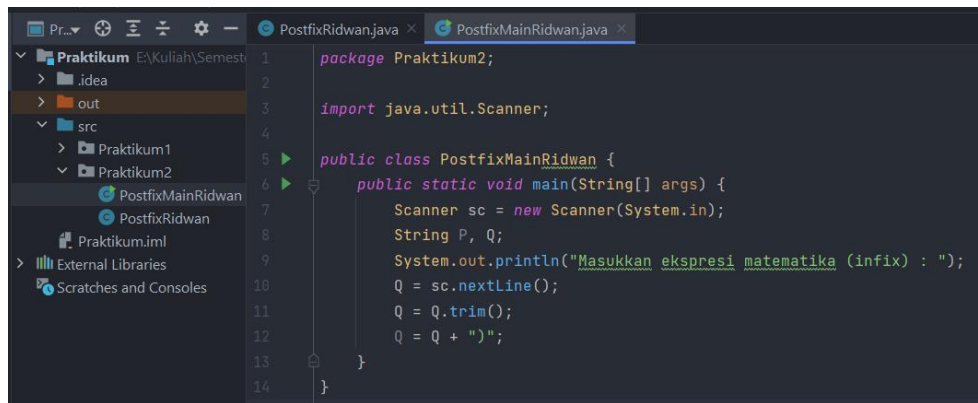
## Langkah 8

Buat method konversi untuk melakukan konversi notasi infix menjadi notasi postfix dengan cara mengecek satu persatu elemen data pada String Q sebagai parameter masukan.

```
64 public String konversi(String Q) {
65     String P = "";
66     char c;
67     for (int i = 0; i < n; i++) {
68         c = Q.charAt(i);
69         if (IsOperand(c)) {
70             P = P + c;
71         }
72         if (c == '(') {
73             push(c);
74         }
75         if (c == ')') {
76             while (stack[top] != '(') {
77                 P = P + pop();
78             }
79             pop();
80         }
81         if (IsOperator(c)) {
82             while (derajat(stack[top]) >= derajat(c)) {
83                 P = P + pop();
84             }
85             push(c);
86         }
87     }
88     return P;
89 }
```

## Langkah 9

Selanjutnya, buat class baru dengan nama PostfixMain tetap pada package Praktikum2. Buat class main, kemudian buat variabel P dan Q. Variabel P digunakan untuk menyimpan hasil akhir notasi postfix setelah dikonversi, sedangkan variabel Q digunakan untuk menyimpan masukan dari pengguna berupa ekspresi matematika dengan notasi infix. Deklarasikan variabel Scanner dengan nama sc, kemudian panggil fungsi built-in trim yang digunakan untuk menghapus adanya spasi di depan atau di belakang teks dari teks persamaan yang dimasukkan oleh pengguna.



```

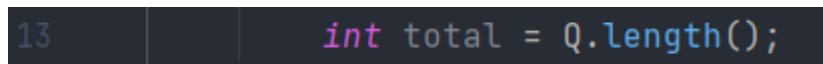
1 package Praktikum2;
2
3 import java.util.Scanner;
4
5 public class PostfixMainRidwan {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         String P, Q;
9         System.out.println("Masukkan ekspresi matematika (infix) : ");
10        Q = sc.nextLine();
11        Q = Q.trim();
12        Q = Q + " ";
13    }
14 }

```

Penambahan string “)” digunakan untuk memastikan semua simbol/karakter yang masih berada di stack setelah semua persamaan terbaca, akan dikeluarkan dan dipindahkan ke postfix.

## Langkah 10

Buat variabel total untuk menghitung banyaknya karakter pada variabel Q.



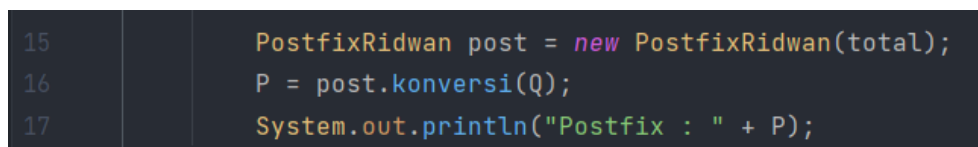
```

13        int total = Q.length();

```

## Langkah 11

Lakukan instansiasi objek dengan nama post dan nilai parameternya adalah total. Kemudian panggil method konversi untuk melakukan konversi notasi infix Q menjadi notasi postfix P.



```

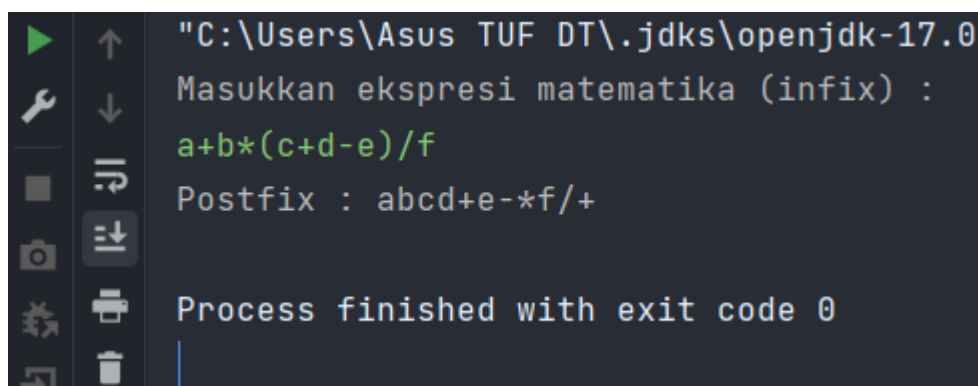
15        PostfixRidwan post = new PostfixRidwan(total);
16        P = post.konversi(Q);
17        System.out.println("Postfix : " + P);

```

## Langkah 12

Compile dan jalankan class PostfixMain dan amati hasilnya

## Verifikasi Hasil Percobaan



```

"C:\Users\Asus TUF DT\.jdk\openjdk-17.0.
Masukkan ekspresi matematika (infix) :
a+b*(c+d-e)/f
Postfix : abcd+e-*f/+
Process finished with exit code 0

```



## Pertanyaan

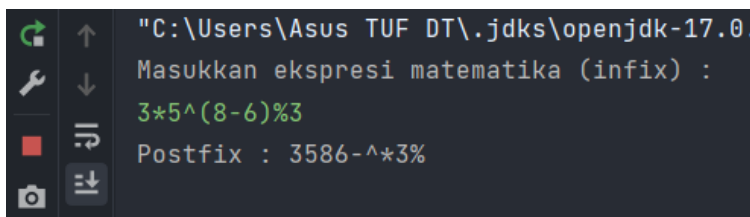
1. Perhatikan class Postfix, jelaskan alur kerja method derajat!
2. Apa fungsi kode program berikut?

```
c = Q.charAt(i);
```

3. Jalankan kembali program tersebut, masukkan ekspresi  $3*5^{(8-6)}\%3$ . Tampilkan hasilnya!
4. Pada soal nomor 3, mengapa tanda kurung tidak ditampilkan pada hasil konversi? Jelaskan!

## Jawaban

1. Pada method derajat memiliki parameter c yaitu tipe data char kemudian melakukan switch c. dari nilai c tersebut ditentukan jika nilai c adalah tanda '^' maka mengembalikan nilai derajat 3. Jika nilai c adalah tanda '%' atau '/' atau '\*' maka mengembalikan nilai derajat 2. Dan jika nilai c adalah tanda '-' atau '+' maka mengembalikan nilai 1, dan memiliki nilai kembalian default 0.
2. Kode program tersebut berfungsi untuk mengambil character dari string Q dengan indeks sesuai dengan iterasi i.
3. Output



```
"C:\Users\Asus TUF DT\.jdk\openjdk-17.0.
Masukkan ekspresi matematika (infix) :
3*5^(8-6)%3
Postfix : 3586-^*3%
```

4. Karena jika nilai c = '(' maka akan dilakukan push ke stack saja, dan jika nilai c = ')' maka akan dilakukan pop dan memasukkan nilai yang di pop ke String P proses ini diulang hingga menemukan tanda '(' dan tanda '(' dihapus.

## Tugas

### Soal Nomor 1

Perhatikan dan gunakan kembali kode program pada Praktikum 1. Tambahkan method getMin pada class Stack yang digunakan untuk mencari dan menampilkan data pakaian dengan harga terendah dari semua data pakaian yang tersimpan di dalam stack!

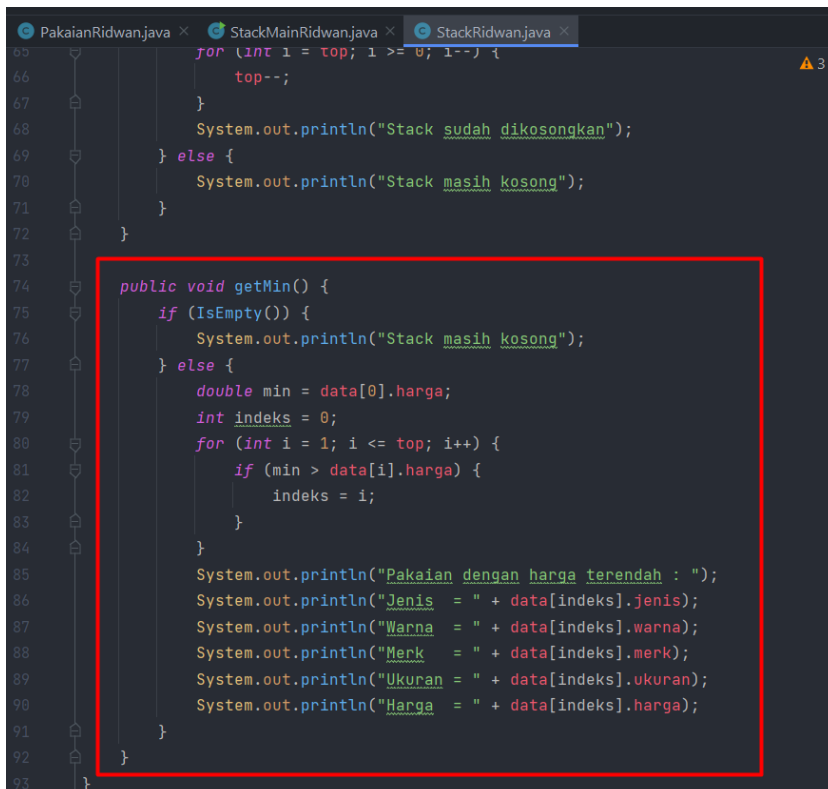
### Soal Nomor 2

Setiap hari Minggu, Dewi pergi berbelanja ke salah satu supermarket yang berada di area rumahnya. Setiap kali selesai berbelanja, Dewi menyimpan struk belanjanya di dalam laci. Setelah dua bulan, ternyata Dewi sudah mempunyai delapan struk belanja. Dewi berencana mengambil lima struk belanja untuk ditukarkan dengan voucher belanja. Buat sebuah program stack untuk menyimpan data struk belanja Dewi, kemudian lakukan juga proses pengambilan data struk belanja sesuai dengan jumlah struk yang akan ditukarkan dengan voucher. Informasi yang tersimpan pada struk belanja terdiri dari:

- Nomor transaksi
- Tanggal pembelian
- Jumlah barang yang dibeli
- Total harga bayar

### Jawaban Nomor 1

Source code yang dimodifikasi pada class StackRidwan



```
65     for (int i = top; i >= 0; i--) {
66         top--;
67     }
68     System.out.println("Stack sudah dikosongkan");
69 } else {
70     System.out.println("Stack masih kosong");
71 }
72 }
73
74 public void getMin() {
75     if (IsEmpty()) {
76         System.out.println("Stack masih kosong");
77     } else {
78         double min = data[0].harga;
79         int indeks = 0;
80         for (int i = 1; i <= top; i++) {
81             if (min > data[i].harga) {
82                 indeks = i;
83             }
84         }
85         System.out.println("Pakaian dengan harga terendah : ");
86         System.out.println("Jenis = " + data[indeks].jenis);
87         System.out.println("Warna = " + data[indeks].warna);
88         System.out.println("Merk = " + data[indeks].merk);
89         System.out.println("Ukuran = " + data[indeks].ukuran);
90         System.out.println("Harga = " + data[indeks].harga);
91     }
92 }
93 }
```

Source code yang dimodifikasi pada class StackMainRidwan

```
PakaianRidwan.java × StackMainRidwan.java × StackRidwan.java ×
17 System.out.println("| [6] Harga terendah |");
18 System.out.println("=====");
19 System.out.print("Masukkan angka menu yang anda pilih : ");
20 int menu = input.nextInt();
21 switch (menu) {
22     case 1 -> {
23         char pilih;
24         do {
25             System.out.print("Jenis: ");
26             String jenis = sc.nextLine();
27             System.out.print("Warna: ");
28             String warna = sc.nextLine();
29             System.out.print("Merk: ");
30             String merk = sc.nextLine();
31             System.out.print("Ukuran: ");
32             String ukuran = sc.nextLine();
33             System.out.print("Harga: ");
34             double harga = sc.nextDouble();
35
36             PakaianRidwan p = new PakaianRidwan(jenis, warna, merk, ukuran, harga);
37             System.out.print("Apakah anda akan menambahkan data baru ke stack? ");
38             pilih = sc.next().charAt(0);
39             sc.nextLine();
40             stk.push(p);
41         } while (pilih == 'y');
42     }
43     case 2 -> stk.pop();
44     case 3 -> stk.peek();
45     case 4 -> stk.print();
46     case 5 -> stk.clear();
47     case 6 -> stk.getMin();
48     default -> System.out.println("Masukkan salah");
```

## Output

```
"C:\Users\Asus TUF DT\.jdk\openjdk-17.0.2\bin\java.exe" "-ja
=====
| [1] push          |
| [2] pop           |
| [3] peek          |
| [4] print         |
| [5] clear         |
| [6] Harga terendah |
=====
Masukkan angka menu yang anda pilih : 1
Jenis: kaos
Warna: hitam
Merk: nevada
Ukuran: M
Harga: 85000
Apakah anda akan menambahkan data baru ke stack (y/n) ? y
Jenis: kemeja
Warna: putih
Merk: styves
Ukuran: XL
Harga: 127000
Apakah anda akan menambahkan data baru ke stack (y/n) ? y
Jenis: celana
Warna: biru
Merk: jeans
Ukuran: L
Harga: 189500
Apakah anda akan menambahkan data baru ke stack (y/n) ? n
apakah ada keperluan lagi? (y/n)
```

```
y
=====
| [1] push          |
| [2] pop           |
| [3] peek          |
| [4] print         |
| [5] clear         |
| [6] Harga terendah |
=====
Masukkan angka menu yang anda pilih : 6
Pakaian dengan harga terendah :
Jenis  = kaos
Warna  = hitam
Merk   = nevada
Ukuran = M
Harga  = 85000.0
```

## Jawaban Nomor 2

## Source Code StrukBelanjaRidwan

```
StrukBelanjaRidwan.java x StackStrukRidwan.java x StrukMain.java x
1 package Tugas;
2
3 public class StrukBelanjaRidwan {
4     int noTransaksi;
5     String tanggal;
6     int jumlahBarang;
7     int totalHarga;
8
9     StrukBelanjaRidwan(int noTransaksi, String tanggal, int jumlahBarang, int totalHarga) {
10         this.noTransaksi = noTransaksi;
11         this.tanggal = tanggal;
12         this.jumlahBarang = jumlahBarang;
13         this.totalHarga = totalHarga;
14     }
15 }
```

## Source Code StackStrukRidwan

```
StrukBelanjaRidwan.java x StackStrukRidwan.java x StrukMain.java x
1 package Tugas;
2
3 public class StackStrukRidwan {
4     StrukBelanjaRidwan[] data;
5     int top;
6     int size;
7
8     StackStrukRidwan(int size) {
9         this.size = size;
10        data = new StrukBelanjaRidwan[size];
11        top = -1;
12    }
13
14    boolean isEmpty() {
15        if (top == -1) {
16            return true;
17        } else {
18            return false;
19        }
20    }
21
22    boolean IsFull() {
23        if (top == size - 1) {
24            return true;
25        } else {
26            return false;
27        }
28    }
29 }
```

```
30 void push(StrukBelanjaRidwan struk) {
31     if (IsFull()) {
32         System.out.println("Stack sudah penuh");
33     } else {
34         top++;
35         data[top] = struk;
36     }
37 }
38
39 void print() {
40     if (IsEmpty()) {
41         System.out.println("Data masih kosong");
42     } else {
43         for (int i = top; i >= 0; i--) {
44             System.out.println("===== Data struk ke " + (i + 1) + " =====");
45             System.out.println("No Transaksi : " + data[i].noTransaksi);
46             System.out.println("Tgl Pembelian : " + data[i].tanggal);
47             System.out.println("Jumlah Barang : " + data[i].jumlahBarang);
48             System.out.println("Total Harga : " + data[i].totalHarga);
49             System.out.println("=====");
50             System.out.println();
51         }
52     }
53 }
```

```
54
55 void pop() {
56     if (IsEmpty()) {
57         System.out.println("Stack masih kosong");
58     } else {
59         StrukBelanjaRidwan keluar = data[top];
60         top--;
61         System.out.println("Data yang keluar :");
62         System.out.println("No Transaksi : " + keluar.noTransaksi);
63         System.out.println("Tgl Pembelian : " + keluar.tanggal);
64         System.out.println("Jumlah Barang : " + keluar.jumlahBarang);
65         System.out.println("Total Harga : " + keluar.totalHarga);
66         System.out.println();
67     }
68 }
69 }
```

## Source Code StrukMain

```
StrukBelanjaRidwan.java x StackStrukRidwan.java x StrukMain.java x
1 package Tugas;
2
3 public class StrukMain {
4     public static void main(String[] args) {
5         StackStrukRidwan stack = new StackStrukRidwan( size: 8);
6         StrukBelanjaRidwan struk1 = new StrukBelanjaRidwan(
7             noTransaksi: 101, tanggal: "6 Februari 2022", jumlahBarang: 2, totalHarga: 20000);
8         stack.push(struk1);
9         StrukBelanjaRidwan struk2 = new StrukBelanjaRidwan(
10            noTransaksi: 102, tanggal: "13 Februari 2022", jumlahBarang: 4, totalHarga: 40000);
11        stack.push(struk2);
12        StrukBelanjaRidwan struk3 = new StrukBelanjaRidwan(
13            noTransaksi: 103, tanggal: "20 Februari 2022", jumlahBarang: 6, totalHarga: 60000);
14        stack.push(struk3);
15        StrukBelanjaRidwan struk4 = new StrukBelanjaRidwan(
16            noTransaksi: 104, tanggal: "27 Februari 2022", jumlahBarang: 8, totalHarga: 80000);
17        stack.push(struk4);
18        StrukBelanjaRidwan struk5 = new StrukBelanjaRidwan(
19            noTransaksi: 105, tanggal: "6 Maret 2022", jumlahBarang: 10, totalHarga: 100000);
20        stack.push(struk5);
21        StrukBelanjaRidwan struk6 = new StrukBelanjaRidwan(
22            noTransaksi: 106, tanggal: "13 Maret 2022", jumlahBarang: 12, totalHarga: 120000);
23        stack.push(struk6);
24        StrukBelanjaRidwan struk7 = new StrukBelanjaRidwan(
25            noTransaksi: 107, tanggal: "20 Maret 2022", jumlahBarang: 14, totalHarga: 140000);
26        stack.push(struk7);
27        StrukBelanjaRidwan struk8 = new StrukBelanjaRidwan(
28            noTransaksi: 108, tanggal: "27 Maret 2022", jumlahBarang: 16, totalHarga: 160000);
29        stack.push(struk8);
30
31        System.out.println("===== Data Semua Struk Belanja Sebelum Ditukarkan =====");
32        stack.print();
33
34        System.out.println("Pengambilan 5 struk untuk ditukarkan");
35        for (int i = 0; i < 5; i++) {
36            stack.pop();
37        }
38        System.out.println("===== Data Struk yang masih di dalam stack =====");
39        stack.print();
40    }
41 }
```

## Output

```
Run: StackMainRidwan x StrukMain x
"C:\Users\Asus TUF DT\.jdk\openjdk-17.0.2\bin\java.exe" "-
===== Data Semua Struk Belanja Sebelum Ditukarkan =====
===== Data struk ke 8=====
No Transaksi : 108
Tgl Pembelian : 27 Maret 2022
Jumlah Barang : 16
Total Harga : 160000
=====

===== Data struk ke 7=====
No Transaksi : 107
Tgl Pembelian : 20 Maret 2022
Jumlah Barang : 14
Total Harga : 140000
=====

===== Data struk ke 6=====
No Transaksi : 106
Tgl Pembelian : 13 Maret 2022
Jumlah Barang : 12
Total Harga : 120000
=====

===== Data struk ke 5=====
No Transaksi : 105
Tgl Pembelian : 6 Maret 2022
Jumlah Barang : 10
Total Harga : 100000
=====
```

```
===== Data struk ke 4=====
No Transaksi : 104
Tgl Pembelian : 27 Februari 2022
Jumlah Barang : 8
Total Harga : 80000
=====

===== Data struk ke 3=====
No Transaksi : 103
Tgl Pembelian : 20 Februari 2022
Jumlah Barang : 6
Total Harga : 60000
=====

===== Data struk ke 2=====
No Transaksi : 102
Tgl Pembelian : 13 Februari 2022
Jumlah Barang : 4
Total Harga : 40000
=====

===== Data struk ke 1=====
No Transaksi : 101
Tgl Pembelian : 6 Februari 2022
Jumlah Barang : 2
Total Harga : 20000
=====
```



```
Pengambilan 5 struk untuk ditukarkan
Data yang keluar :
No Transaksi : 108
Tgl Pembelian : 27 Maret 2022
Jumlah Barang : 16
Total Harga : 160000

Data yang keluar :
No Transaksi : 107
Tgl Pembelian : 20 Maret 2022
Jumlah Barang : 14
Total Harga : 140000

Data yang keluar :
No Transaksi : 106
Tgl Pembelian : 13 Maret 2022
Jumlah Barang : 12
Total Harga : 120000

Data yang keluar :
No Transaksi : 105
Tgl Pembelian : 6 Maret 2022
Jumlah Barang : 10
Total Harga : 100000

Data yang keluar :
No Transaksi : 104
Tgl Pembelian : 27 Februari 2022
Jumlah Barang : 8
Total Harga : 80000
```

```
===== Data Struk yang masih di dalam stack =====
===== Data struk ke 3=====
No Transaksi : 103
Tgl Pembelian : 20 Februari 2022
Jumlah Barang : 6
Total Harga : 60000
=====

===== Data struk ke 2=====
No Transaksi : 102
Tgl Pembelian : 13 Februari 2022
Jumlah Barang : 4
Total Harga : 40000
=====

===== Data struk ke 1=====
No Transaksi : 101
Tgl Pembelian : 6 Februari 2022
Jumlah Barang : 2
Total Harga : 20000
=====
```