

**LAPORAN HASIL PRAKTIKUM
ALGORITMA STRUKTUR DATA
STACK**



Oleh:

FAUZIYYAH ADELIA RAMANDA

NIM. 2341760145

SIB-1F / 10

**D-IV SISTEM INFORMASI BISNIS
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG**

7.2. Praktikum 1

7.2.1 Langkah-langkah Percobaan

Stack.java

```
public class Stack {
    int data[];
    int size;
    int top;

    public Stack(int size){
        this.size = size;
        data = new int[size];
        top = -1;
    }

    public boolean isFull(){
        if(top == size - 1){
            return true;
        }
        else {
            return false;
        }
    }

    public boolean isEmpty(){
        if(top == -1){
            return true;
        }
        else {
            return false;
        }
    }

    public void push(int dt){
        if(!isFull()){
            top++;
            data[top] = dt;
        }
        else{
            System.out.println(x:"Stack penuh");
        }
    }

    public void pop(){
        if(!isEmpty()){
            int x = data[top];
            top--;
            System.out.println("Data yang dikeluarkan dari stack: "+x);
        }
        else{
            System.out.println(x:"Stack masih kosong");
        }
    }

    public void peek(){
        System.out.println("Elemen teratas stack: "+data[top]);
    }

    public void print(){
        System.out.println(x:"Isi stack: ");
        for(int i = top; i>=0; i--){
            System.out.println(data[i]+" ");
        }
        System.out.println(x:" ");
    }

    public void clear(){
        if(!isEmpty()){
            for(int i = top; i>=0; i--){
                top--;
            }
            System.out.println(x:"Stack sudah dikosongkan");
        }
        else{
            System.out.println(x:"Stack masih kosong");
        }
    }
}
```

StackDemo.java

```
public class StackDemo {
    Run | Debug
    public static void main(String[] args){
        Stack stack = new Stack(size:10);
        stack.push(1:8);
        stack.push(1:12);
        stack.push(1:18);
        stack.print();
        stack.pop();
        stack.peek();
        stack.pop();
        stack.push(-5);
        stack.print();
    }
}
```

7.2.2 Verifikasi Hasil Percobaan

```
Isi stack:  
18  
12  
8  
  
Data yang dikeluarkan dari stack: 18  
Elemen teratas stack: 12  
Data yang dikeluarkan dari stack: 12  
Isi stack:  
-5  
8
```

7.2.3 Pertanyaan

1. Pada method pop(), mengapa diperlukan pemanggilan method isEmpty()? Apa yang terjadi jika tidak ada pemanggilan isEmpty()?

Untuk memeriksa apakah stack kosong sebelum melakukan operasi pop. Jika tidak ada pemanggilan method isEmpty(), maka program akan langsung melakukan operasi pengurangan indeks top tanpa mengambil data terlebih dahulu. Hal ini dapat menyebabkan error indeks jika method pop() dipanggil saat stack kosong

2. Jelaskan perbedaan antara method peek() dengan method pop() pada class Stack. Method peek() hanya melihat elemen teratas pada stack tanpa mengubahnya, sedangkan pop() mengambil elemen paling atas dan mengurangi indeks top hingga stack jadi lebih rendah

7.3. Praktikum 2

7.3.1 Langkah-langkah Percobaan

Pakaian.java

```
public class Pakaian {
    String jenis, warna, merk, ukuran;
    double harga;

    Pakaian(String jenis, String warna, String merk, String ukuran, double harga) {
        this.jenis = jenis;
        this.warna = warna;
        this.merk = merk;
        this.ukuran = ukuran;
        this.harga = harga;
    }
}
```

StackPakaian.java

```
import java.util.Scanner;

public class StackPakaian {
    private Stack<Pakaian> stack;
    private int top;

    public StackPakaian() {
        stack = new Stack<Pakaian>();
        top = -1;
    }

    public boolean isEmpty() {
        return top == -1;
    }

    public boolean isFull() {
        return top == stack.size() - 1;
    }

    public void push(Pakaian p) {
        if (!isFull()) {
            stack.add(p);
            top++;
        } else {
            System.out.println("Stack penuh!");
        }
    }

    public Pakaian pop() {
        if (!isEmpty()) {
            Pakaian p = stack.remove(stack.size() - 1);
            top--;
            return p;
        } else {
            System.out.println("Stack kosong!");
            return null;
        }
    }

    public Pakaian peek() {
        if (!isEmpty()) {
            return stack.get(stack.size() - 1);
        } else {
            System.out.println("Stack kosong!");
            return null;
        }
    }

    public void print() {
        System.out.println("Isi Stack:");
        for (Pakaian p : stack) {
            System.out.println(p.jenis + " | " + p.warna + " | " + p.merk + " | " + p.ukuran + " | " + p.harga);
        }
    }
}
```

StackMain.java

```
import java.util.Scanner;

public class StackMain {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        StackPakaian stk = new StackPakaian();
        char pilih;

        do {
            System.out.println("Jenis: ");
            String jenis = sc.nextLine();
            System.out.println("Warna: ");
            String warna = sc.nextLine();
            System.out.println("Merk: ");
            String merk = sc.nextLine();
            System.out.println("Ukuran: ");
            String ukuran = sc.nextLine();
            System.out.println("Harga: ");
            double harga = sc.nextDouble();

            Pakaian p = new Pakaian(jenis, warna, merk, ukuran, harga);
            System.out.print("Apakah Anda akan menambahkan data baru ke stack (y/e)?");
            pilih = sc.next().charAt(0);
            sc.nextLine();
            stk.push(p);
        } while (pilih == 'y');

        stk.print();
        stk.pop();
        stk.peek();
        stk.print();
    }
}
```

7.3.2 Verifikasi Hasil Percobaan

```
Jenis: Kaos
Warna:
Hitam
Merk: Nevada
Ukuran:
M
Harga:
85000
Apakah Anda akan menambahkan data baru ke stock (y/n)?y
Jenis: Kemeja
Warna:
Putih
Merk: Styves
Ukuran:
XL
Harga:
127000
Apakah Anda akan menambahkan data baru ke stock (y/n)?y
Jenis: Celana
Warna:
Biru
Merk: Levis
Ukuran:
L
Harga:
189000
Apakah Anda akan menambahkan data baru ke stock (y/n)?n
Isi stack:
Celana Biru Levis L 189000.0
Kemeja Putih Styves XL 127000.0
Kaos Hitam Nevada M 85000.0

Data yang keluar: Celana Biru Levis L 189000.0
Elemen teratas: Kemeja Putih Styves XL 127000.0
Isi stack:
Kemeja Putih Styves XL 127000.0
Kaos Hitam Nevada M 85000.0
```

7.3.3 Pertanyaan

1. Berapa banyak data pakaian yang dapat ditampung di dalam stack? Tunjukkan potongan kodeprogram untuk mendukung jawaban Anda tersebut!

5, berikut programnya

```
StackPakaian stk = new StackPakaian(5);
```

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);
```

Data spt pakaian jenis, warna , merk, ukuran, harga

3. Apakah fungsi penggunaan **do-while** yang terdapat pada class **StackMain**?

Fungsinya untuk memasukkan secara berulang ke dalam stack hingga mencapai kapasitas maksimumnya

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!

```
import java.util.Scanner;

public class StackMain {
    Run | Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Stack15_Pakaian stk = new Stack_Pakaian(5);

        int pilihan;

        do {
            System.out.println("\nPilih Operasi:");
            System.out.println("1. Push");
            System.out.println("2. Pop");
            System.out.println("3. Peek");
            System.out.println("4. Print");
            System.out.println("5. GetMax");
            System.out.println("6. Exit");
            System.out.print("Masukkan pilihan: ");
            pilihan = sc.nextInt();
            sc.nextLine();

            switch (pilihan) {
                case 1:
                    System.out.println();
                    System.out.print("Jenis\t: ");
                    String jenis = sc.nextLine();
                    System.out.print("Warna\t: ");
                    String warna = sc.nextLine();
                    System.out.print("Merk\t: ");
                    String merk = sc.nextLine();
                    System.out.print("Ukuran\t: ");
                    String ukuran = sc.nextLine();
                    System.out.print("Harga\t: ");
                    double harga = sc.nextDouble();

                    Pakaian15 p = new Pakaian(jenis, warna, merk, ukuran, harga);
                    stk.push(p);
                    break;
                case 2:
                    stk.pop();
                    break;
                case 3:
                    stk.peek();
                    break;
                case 4:
                    stk.print();
                    break;
                case 5:
                    stk.getMax();
                    break;
                case 6:
                    System.out.println("Program selesai.");
                    break;
                default:
                    System.out.println("Pilihan tidak valid.");
            }
        } while (pilihan != 5);

        sc.close();
    }
}
```

7.4. Praktikum 3

7.4.1 Langkah Percobaan

```
import java.util.Scanner;

public class PostfixMain {
    Run | Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String P, Q;
        System.out.println(x:"Masukkan ekspresi matematika (infix) : ");
        Q = sc.nextLine();
        Q = Q.trim();
        Q = Q + ")";

        int total = Q.length();

        Postfix post = new Postfix(total);
        P = post.konversi(Q);
        System.out.println("Postfix : " + P);
    }
}
```

7.4.2 Verifikasi Hasil Percobaan

```
PS C:\Users\USER\Documents\kuliah_uiii\SM>
Masukkan ekspresi matematika (infix) :
a+b*(c+d-e)/f
Postfix : abcd+e-*f/+
```

7.4.3 Pertanyaan

1. Perhatikan class **Postfix**, jelaskan alur kerja method **derajat**!

Method derajat menghitung derajat dari operator yang diinput, tapi a digunakan dalam method konversi untuk mengkonversi infix expression menjadi postfix expression.

2. Apa fungsi kode program berikut?

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

Fungsinya untuk mengambil karakter pada posisi i dan string Q, dan menyimpannya dalam variabel c untuk diproses selanjutnya

3. Jalankan kembali program tersebut, masukkan ekspresi **5*4^(1+2)%3**. Tampilkan hasilnya!

```
Masukkan ekspresi matematika (infix) :
5*4^(1+2)%3
Postfix : 5412+^*3%
```

4. Pada soal nomor 3, mengapa tanda kurung tidak ditampilkan pada hasil konversi? Jelaskan!
Karena tanda kurung bukan derajat operator aritmatika

7.5 Tugas

1. Perhatikan dan gunakan kembali kode program pada **Praktikum 2**. Tambahkan method **getMax** pada class **Stack** yang digunakan untuk mencari dan menampilkan data pakaian dengan harga tertinggi dari semua data pakaian yang tersimpan di dalam stack!

```
public void getMax() {
    if (isEmpty()) {
        System.out.println("Stack masih kosong");
    } else {
        Pakaian maxPakaian = data[0];

        for (int i = 1; i <= top; i++) {
            if (data[i].harga > maxPakaian.harga) {
                maxPakaian = data[i];
            }
        }

        System.out.println("Data pakaian dengan harga tertinggi: " + maxPakaian.jenis + " " + maxPakaian.warna + " " + maxPakaian.merk + " " + maxPakaian.ukuran + " " + maxPakaian.harga);
    }
}
```

```
case 5:
    stk.getMax();
    break;
```

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

Tampilkan informasi struk belanja yang masih tersimpan di dalam stack!

Struk10.java

```
Struk10.java Struk10.java StrukMain10.java
public class Struk10 {
    int noTransaksi, jumlahBarang;
    String tglBeli;
    int totalBayar;
    int size;
    int top;
    Struk10[] data;
    Struk10[] stk;

    Struk10(int no, String tgl, int jb, int tb){
        noTransaksi = no;
        tglBeli = tgl;
        jumlahBarang = jb;
        totalBayar = tb;
    }

    public Struk10(int size){
        this.size = size;
        data = new Struk10[size];
        top = -1;
    }

    public boolean isEmpty(){
        return top == -1;
    }

    public boolean isFull(){
        return top == size - 1;
    }

    public void push(Struk10 dt){
        if (!isFull()) {
            top++;
            data[top] = dt;
        } else {
            System.out.println("Isi Stack Penuh!");
        }
    }

    public void pop(){
        if (!isEmpty()) {
            Struk10 x = data[top];
            top--;
            System.out.println("Data yang keluar: " + x.noTransaksi + " " + x.tglBeli + " " + x.jumlahBarang + " " + x.totalBayar + " ");
        } else {
            System.out.println("Stack masih kosong");
        }
    }

    public void peek(){
        System.out.println("Elemen teratas: " + data[top].noTransaksi + " " + " " + data[top].tglBeli + " " + data[top].jumlahBarang + " " + data[top].totalBayar);
    }

    public void print(){
        System.out.println("Isi stack: ");
        for (int i = top; i >= 0; i--) {
            System.out.println(data[i].noTransaksi + " " + data[i].tglBeli + " " + data[i].jumlahBarang + " " + data[i].totalBayar + " ");
        }
        System.out.println("");
    }

    public void clear(){
        if (!isEmpty()) {
            for (int i = top; i >= 0; i--) {
                top--;
            }
            System.out.println("Stack sudah dikosongkan");
        } else {
            System.out.println("Sagati Stack masih kosong");
        }
    }
}
```

StrukMain10.java

```

StrukMain10.java / StrukMain10 / kuponStruk10
import java.util.Scanner;

public class StrukMain10 {
    Scanner sc = new Scanner(System.in);

    Run | Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Struk10 stk = new Struk10(20);
        StrukMain10 main = new StrukMain10();

        int pil;
        while (true) {
            System.out.println("\nPilih Operasi:");
            System.out.println("1. Push");
            System.out.println("2. Pop");
            System.out.println("3. Peek");
            System.out.println("4. Print");
            System.out.println("5. Kupon");
            System.out.println("6. Exit");
            System.out.print("Masukkan pilihan: ");
            pil = sc.nextInt();
            switch (pil) {
                case 1:
                    main.pushPakaian(stk);
                    break;
                case 2:
                    stk.pop();
                    break;
                case 3:
                    stk.peak();
                    break;
                case 4:
                    stk.print();
                    break;
                case 5:
                    main.kupon(stk);
                    break;
                case 6:
                    return;
                default:
                    System.out.println("Pilihan tidak tersedia.");
                    break;
            }
        }
    }

    public void pushPakaian(Struk10 stk) {
        char pilih;
        do {
            System.out.print("No. Belanja: ");
            int noTra = sc.nextInt();
            System.out.print("Tanggal (dd/mm/yyyy) : ");
            String tanggal = sc.next();
            System.out.print("Jumlah: ");
            int jumlah = sc.nextInt();
            System.out.print("Total Bayar: ");
            int total = sc.nextInt();

            Struk10 d = new Struk10(noTra, tanggal, jumlah, total);

            System.out.print("Apakah Anda akan menambahkan data baru ke stack (y/n) ? ");
            pilih = sc.next().charAt(0);
            stk.push(d);
        } while (pilih == 'y');
    }

    public void kupon(Struk10 stk){
        System.out.println("Data yang diambil");
        int i = 0;
        while (i < 5) {
            stk.pop();
            i++;
        }

        System.out.println();
        System.out.println("Data yang tersisa");
        stk.print();
    }
}

```