

Jurnal 05: Stack dan Queue

Kerjakanlah soal-soal berikut. Untuk soal pertama, buatlah juga penjelasan kode program dalam bentuk video dengan durasi maksimal 5 menit. Penjelasan pada soal kedua cukup dengan menggunakan narasi yang dituliskan pada Word/ sejenisnya. Diharapkan penulisan kode program sesuai dengan Java coding style.

1. Buatlah suatu program untuk mengubah notasi *infix* menjadi *postfix* (*Reverse-Polish notation*). Gunakan struktur data **Stack** untuk program yang dibuat. Kode program boleh langsung mengambil dari kode yang banyak dituliskan/ dijelaskan di internet, namun harus menyertakan **sumber kode** (tuliskan nama pembuat/ judul artikel, alamat URL dan tanggal akses). Anda juga boleh mengambil sumber dari video YouTube mengenai pengubahan infix ke postfix yang disertakan pada LMS. Video penjelasan, pada sisi lainnya, **harus** menggunakan **penjelasan** dari Anda sendiri. Sertakan pula penjelasan mengenai apa itu notasi infix dan postfix pada video.

Noted:

Notasi *infix* adalah notasi dimana operator (+, -, *, /, ^) berada **di antara operand** (angka atau huruf yang akan dilakukan operasi aritmatika padanya). Contoh: $1 + 2$, $(a + b) / (c + d)$

Notasi *postfix* adalah notasi dimana operator (+, -, *, /, ^) berada **setelah/ di belakang operand** (angka atau huruf yang akan dilakukan operasi aritmatika padanya). Contoh: $1\ 2\ +$, $a\ b\ +\ c\ d\ +\ /$

Melakukan *parsing* untuk notasi *infix* dalam suatu program tidaklah terlalu efisien. Karenanya, dilakukan pengubahan notasi *infix* ke dalam notasi lainnya.

Dalam pengubahan notasi, perhatikan *precedence* dari setiap operator. Ingat, operator pangkat (^) merupakan operator dengan *precedence* tertinggi dan operator penjumlahan serta pengurangan merupakan operator dengan *precedence* terendah. Selain itu, tanda kurung akan selalu menjadi pembatas bagi ekspresi aritmatika yang dilakukan.

Contoh masukan:

$a+b*(c^d-e)^{(f+g*h)}-i$

Contoh keluaran:

$abcd^e-fgh*+^*+i-$

2. Robert lagi-lagi merasa bingung. Banyaknya tugas, kuis, belum ditambah dengan kesibukan Robert di kegiatan kemahasiswaan, telah membuat Robert seringkali melupakan hal-hal yang seharusnya Robert lakukan. Pagi ini saja Robert lupa membawa tugas ISD yang harus dikumpulkan karena salah mengira waktu pengumpulan. Bantulah Robert untuk mencatat dan mengingatkan pekerjaan apa saja yang harus dikerjakan dalam waktu dekat dengan membuat program *to-do-list* sederhana.

Fitur yang harus dimiliki aplikasi adalah sebagai berikut:

- a. Menambah nama pekerjaan yang harus dilakukan.
- b. Menampilkan nama pekerjaan yang harus diselesaikan terlebih dahulu. Pekerjaan yang dimasukkan pertama harus diselesaikan lebih dahulu, kemudian pekerjaan kedua, ketiga, dan seterusnya. Contoh, jika yang dimasukkan pertama kali adalah Jurnal 05 ISD, kemudian Kuis Sistem Basis Data, lalu Tugas Besar, maka urutan tugas yang harus diselesaikan adalah Jurnal 05 ISD – Kuis Sistem Basis Data – Tugas Besar.
- c. Menghapus pekerjaan yang sudah diselesaikan. Dari contoh pada bagian b, jika pekerjaan Jurnal 05 ISD sudah selesai dikerjakan, maka tugas tersebut harus dihapus dari list to-do.
- d. Menampilkan seluruh isi to-do list.

Tuliskan penjelasan program dan kumpulkan beserta project untuk Soal 2. Jangan lupa menuliskan NIM dan Nama Anda pada file penjelasan.

Java Coding Style

Ikuti gaya penulisan berikut di kode program agar mendapat nilai maksimal

1. Setiap nama kelas, variabel dan konstanta harus dapat menggambarkan isinya.

Misal, membuat variabel untuk menyimpan jumlah penumpang di bus. Deklarasi:

```
int penumpang = 0;           // Bagus, OK
int p = 0;                   // Tidak OK
```

2. Penulisan nama harus menggunakan huruf besar/kecil yang sesuai.

Nama kelas: UpperCamelCase. Misal: PenumpangBus

Nama variabel: lowerCamelCase. Misal: penumpangBus

Nama konstanta: CAPITAL_CASE. Misal: PENUMPANG_BUS

Nama method: lowerCamelCase. Misal: hitungPenumpangBus

3. Penggunaan tab/spasi ketika berbeda blok program, WAJIB masuk 1 tab atau 4 spasi ke dalam.

```
public class PenumpangBus {
    public static void main(String[] args) {
        int penumpang = 0;
        penumpang = penumpang + 5;
        penumpang = penumpang - 4 + 2;
        penumpang = penumpang - 1;
        penumpang = penumpang - 2 + 3;
        penumpang = penumpang - 2 + 5;
        penumpang = penumpang - 1 + 3;
        System.out.println(penumpang);
    }
}
```

4. Gunakan komentar seperlunya.

Javadoc comment (yang warna biru) hanya untuk mengomentari kelas, method dan variable

```
/** Jumlah penumpang di bus */
int penumpang = 0;

/* Ada 5 orang penumpang naik ke bus */
penumpang = penumpang + 5;

// Penumpang turun 4 orang, naik 2
penumpang = penumpang - 4 + 2;
```