

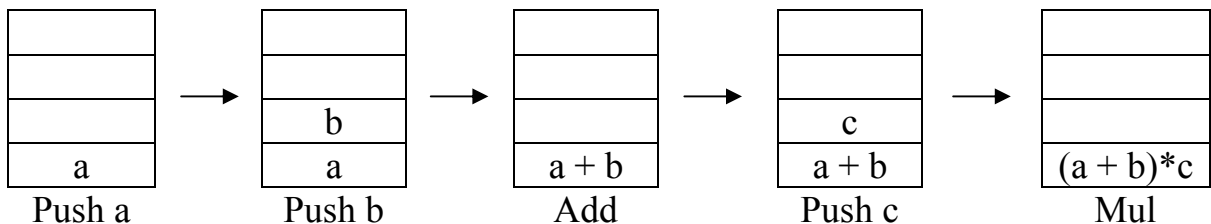
□ Format instruksi 0-alamat

| <u>Instruksi</u> | <u>Keterangan</u> |
|------------------|----------------------------------|
| PUSH X | $TOS \leftarrow [X]$ |
| POP Y | $Y \leftarrow [TOS]$ |
| MUL | $TOS \leftarrow [TOS-1] * [TOS]$ |
| DIV | $TOS \leftarrow [TOS-1] / [TOS]$ |
| ADD | $TOS \leftarrow [TOS-1] + [TOS]$ |
| SUB | $TOS \leftarrow [TOS-1] - [TOS]$ |

➤ Contoh : $(a + b) * c$

Instruksinya :

Push a
Push b
Add
Push c
Mul



- ❖ Rumus-rumus matematika biasanya diekspresikan dalam notasi infix. Dalam notasi ini, operasi biner berada di antara 2 buah operand (mis: $a + b$).
- ❖ Teknik lain adalah yang dikenal sebagai notasi reverse polish atau postfix. Pada notasi ini, operator berada setelah 2 buah operandnya.

Contoh :

infix
 $a + b$
 $a + (b * c)$
 $(a + b) * c$

Postfix
 $ab+$
 $abc*+$
 $ab+c*$

- ❖ Kelebihan postfix adalah bahwa ekspresi dalam bentuk notasi ini mudah untuk dievaluasi dengan menggunakan stack.

- ❖ Proses pengubahan ekspresi infix menjadi ekspresi postfix akan mudah sekali dilakukan apabila menggunakan algoritma Dijkstra sebagai berikut:

1. Periksa elemen berikutnya

- Jika elemen itu berupa operand, outputkan operand tersebut
- Jika elemen itu berupa tanda kurung buka, push elemen tersebut ke stack
- Jika elemen itu merupakan operator, maka :
 - Jika elemen paling atas stack berupa tanda kurung buka, maka push operator tersebut
 - Jika operator itu memiliki prioritas yang lebih tinggi dibandingkan dengan prioritas elemen paling atas stack, maka push operator tersebut
 - Jika tidak kedua-duanya, maka outputkan operator tersebut
- Jika elemen itu berupa tanda kurung tutup, pop operator-operator ke output sampai ditemukan tanda kurung buka. Pop dan buang tanda kurung buka tersebut.

2. Jika terdapat input lainnya, maka lakukan langkah 1. Jika tidak terdapat input lainnya outputkan operand-operand lainnya yang masih ada di stack

Contoh : Ubah bentuk infix berikut menjadi bentuk postfix
 $f = (a - b) / (c + d * e)$

Jawab :

1.

| |
|---|
| |
| |
| (|

2.

| |
|---|
| |
| |
| (|

Output :
a

3.

| |
|---|
| |
| - |
| (|

Output :
a

4.

| |
|---|
| |
| - |
| (|

Output :
ab

5.

| |
|--|
| |
| |
| |

Output :
ab-

6.

| |
|---|
| |
| |
| / |

Output :
ab-

7.

| |
|---|
| |
| (|
| / |

Output :
ab-

8.

| |
|---|
| |
| (|
| / |

Output :
ab-c

9.

| |
|---|
| |
| + |
| (|
| / |

Output :
ab-c

10.

| |
|---|
| |
| + |
| (|
| / |

Output :
ab-cd

11.

| |
|---|
| |
| * |
| + |
| (|
| / |

Output :
ab-cd

12.

| |
|---|
| |
| * |
| + |
| (|
| / |

Output :
ab-cde

13.

| |
|--|
| |
| |
| |
| |

Output :
ab-cde*+ /

- Dengan menggunakan notasi postfix $ab-cde*+/,$ instruksi-instruksi untuk melakukan operasi aritmetik : $f = (a-b)/(c+d*e)$ akan lebih ditentukan.

Instruksinya :

Push a
 Push b
 Sub
 Push c
 Push d
 Push e
 Mul
 Add
 Div
 Pop f

