**Laporan Teori Algoritma dan Struktur Data**

**Tugas 8 – Stack**

**Dosen Pengampu : Triana Fatmawati, S.T., M.T**



**Nama : Annisa**

**Nim : 2341760032**

**Prodi : D-IV Sistem Informasi Bisnis**

**Absen : 04**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**2023/2024**

**Latihan**

**1. Tuliskan langkah-langkah pengerjaan dari beberapa rangkaian operasi stack berikut:**

**1) Push(10)**

**2) Push(6)**

**3) Pop()**

**4) Push(8)**

**5) Push(2)**

**6) Pop**

**7) Pop**

**8) Push(4)**

**Jika kondisi kondisi awal stack kosong, berapa nilai top saat ini (setelah langkah 8)?**

* Tahap 1 : push(10)

|  |  |
| --- | --- |
| 0 | 10 |
| index | nilai |

* Tahap 2 : push (6)

|  |  |
| --- | --- |
| 1 | 6 |
| 0 | 10 |
| index | nilai |

* Tahap 3 : pop ()

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 6 | Setalah pop 🡪 |  | |
| 0 | 10 | 0 | 10 |
| index | nilai | index | nilai |

* Tahap 4 : push (8)

|  |  |
| --- | --- |
| 1 | 8 |
| 0 | 10 |
| index | nilai |

* Tahap 5 : push (2)

|  |  |
| --- | --- |
| 2 | 2 |
| 1 | 8 |
| 0 | 10 |
| index | nilai |

* Tahap 6 : pop ()

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2 | 2 | Setalah pop 🡪 |  | |
| 1 | 8 | 1 | 8 |
| 0 | 10 | 0 | 10 |
| index | nilai |  | index | nilai |

* Tahap 7 : pop ()

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 8 | Setalah pop 🡪 |  | |
| 0 | 10 | 0 | 10 |
| index | nilai | index | nilai |

* Tahap 8 : push (4)

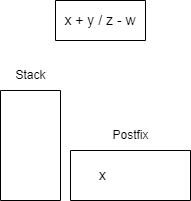
|  |  |
| --- | --- |
| 1 | 4 |
| 0 | 10 |
| index | nilai |

Hasil Akhir

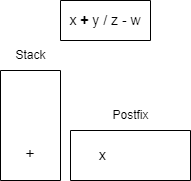
Nilai top setelah langkah 8 terdapat pada indeks ke-1 bernilai 4.

1. **Lakukan konversi notasi infix berikut menjadi notasi postfix!**
2. **x + y / z – w**

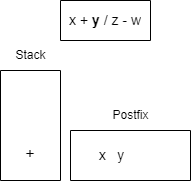
* Tahap 1: Operand x, masuk ke postfix



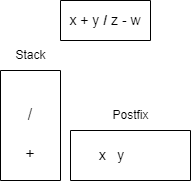
* Tahap 2: Operator ‘+’, push ke stack



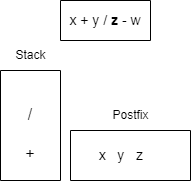
* Tahap 3: Operand ‘y’, masuk ke postfix



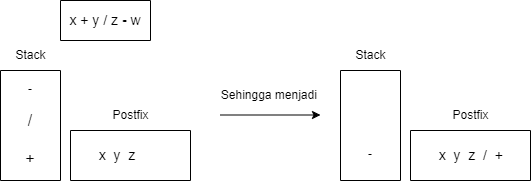
* Tahap4: Operand ‘/’, push ke stack



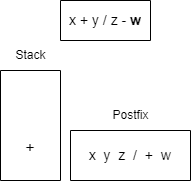
* Tahap 5: Operand ‘z’, masuk ke postfix



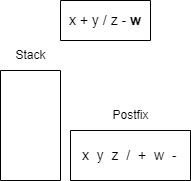
* Tahap 6: Operator
* push ke stack
* Pop ‘/’ , karena derajat ‘/’ lebih tinggi dari pada ‘–‘ dan dimasukkan ke postfix.
* Pop ‘+’, karena dejarat ‘+’ dan ‘–‘ sama, maka letakkan ke postfix terlebih dahulu yang lebih kiri.



* Tahap 7: Operand ‘w’ masuk ke postfix



* Tahap 8: Pop() Operator’-‘, lalu memasukkan ke postfix.

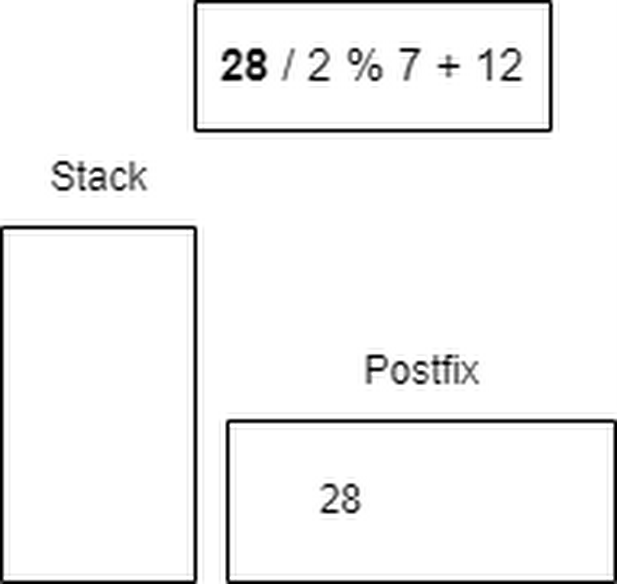


Hasil Konversi:

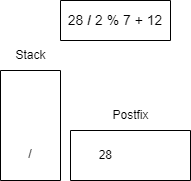
x + y / z- w 🡪 x y z / + w –

1. **28 / 2 % 7 + 12**

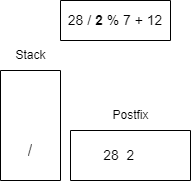
* Tahap 1: Operand ‘28’ masuk ke postfix



* Tahap 2: Operator ‘/’ push ke stack



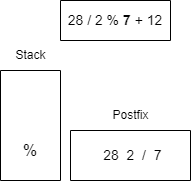
* Tahap 3: Operand ‘2’ masuk ke postfix



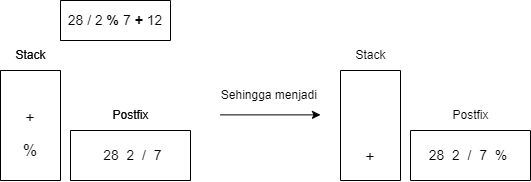
* Tahap 4
* Operator ‘%’push ke stack
* Pop() ‘/’ karena memiliki derajat yang sama dengan ‘%’. Lalu meletakkan operator ‘/’ terlebih dahulu ke postfix di bagian paling kiri.



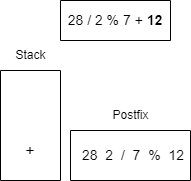
* Tahap 5: Operand ‘7’ masuk ke postfix



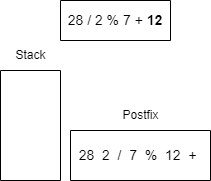
* Tahap 6
* Operator ‘+’ push ke stack
* Pop() operator ‘%’ karena , derajat ‘%’ lebih tinggi dibandingkan dengan ‘+’. Lalu meletakkan operator ‘%’ terlebih dahulu ke postfix.



* Tahap 7: Operand ‘12’ masuk ke postfix



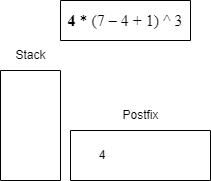
* Tahap 8: Pop() operator ‘+’ dan dimasukkan ke postfix



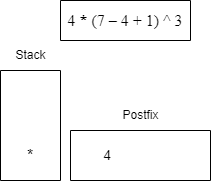
Hasil Konversi

1. 2 % 7 + 12 🡪 28 2 / 7 % 12 +
2. **4 \* (7 – 4 + 1) ^ 3**

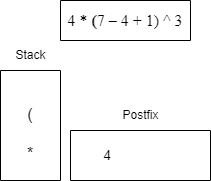
* Tahap 1: Operand ‘4’ masuk ke postfix.



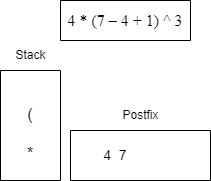
* Tahap 2: Push Operator ‘\*’ ke stack



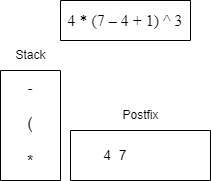
* Tahap 3: Push tanda ‘(‘ ke stack



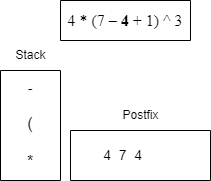
* Tahap 4: Operand ‘7’ masuk ke postfix



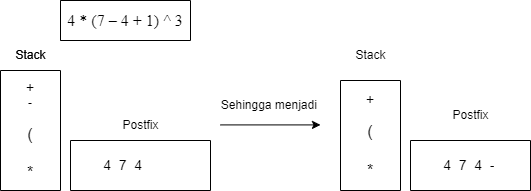
* Tahap 5: Push operator ‘-‘ ke stack



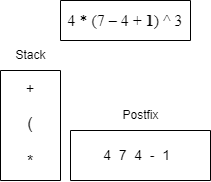
* Tahap 6: Operand ‘4’ masuk ke postfix



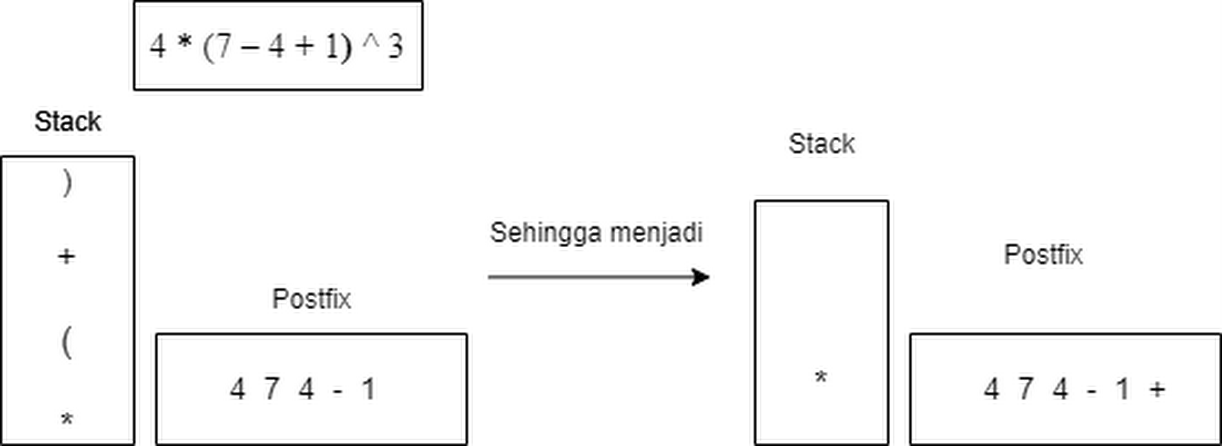
* Tahap 7
* Push operator ‘+’ ke dalam stack.
* Pop() operator ‘-‘, sebab derajatnya sama dengan ‘+’. Maka letakkan operator ‘-‘ ke postfix di bagian paling kiri.



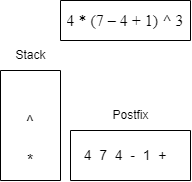
* Tahap 8: Operand ‘1’ masuk ke postfix



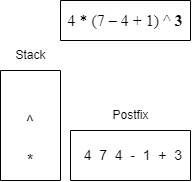
* Tahap 9
* Push tanda ‘)’ ke stack
* Pop() operator ‘+’ lalu masukkan ke dalam postfix
* Pop() tanda ‘)’ pada stack, tidak perlu dimasukkan ke postfix



* Tahap 10: Push operator ‘^’ ke dalam stack

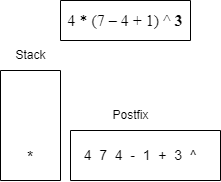


* Tahap 11: Operand ‘3’ dimasukkan ke postfix

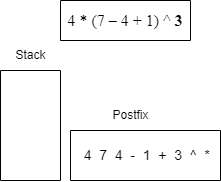


* Tahap 12:

Pop() operator ‘^’ lalu memasukkan ke dalam postfix karena jika dibandingkan dengan operator ‘\*’, operator ‘^’ memiliki derajat lebih tinggi.



* Tahap 13: Pop() operator ‘\*’ lalu memasukkannya ke dalam postfix.



Hasil Konversi

4 \* (7 – 4 + 1) ^ 3 🡪 4 7 4 – 1 + 3 ^ \*