****

**ALGORITMA DAN STRUKTUR DATA**

INF1083

**LAPORAN PRAKTIKUM 6 : Struktur Data Dasar (1)**

Oleh :

*Akhmad Qasim*

*2211102441237*

Teknik Informatika

Fakultas Sains & Teknologi

Universitas Muhammadiyah Kalimantan Timur

Samarinda, 2023

**Laporan Praktikum 6:**

**Struktur Data Dasar (1)**

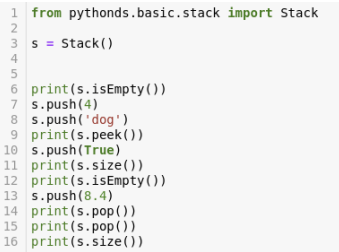
**Pokok Bahasan:**

* Stack.
* Infix, Prefix, dan Postfix.

**Tujuan Pembelajaran:**

* Memahami implementasi *Stack* pada struktur data *Python*.
* Memahami penulisan ekspresi aritmatika di *Python*.

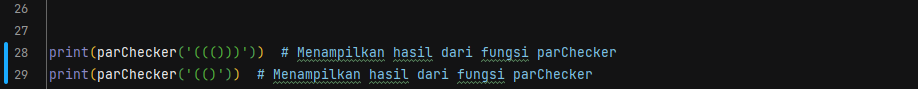
**Percobaan & Latihan:**

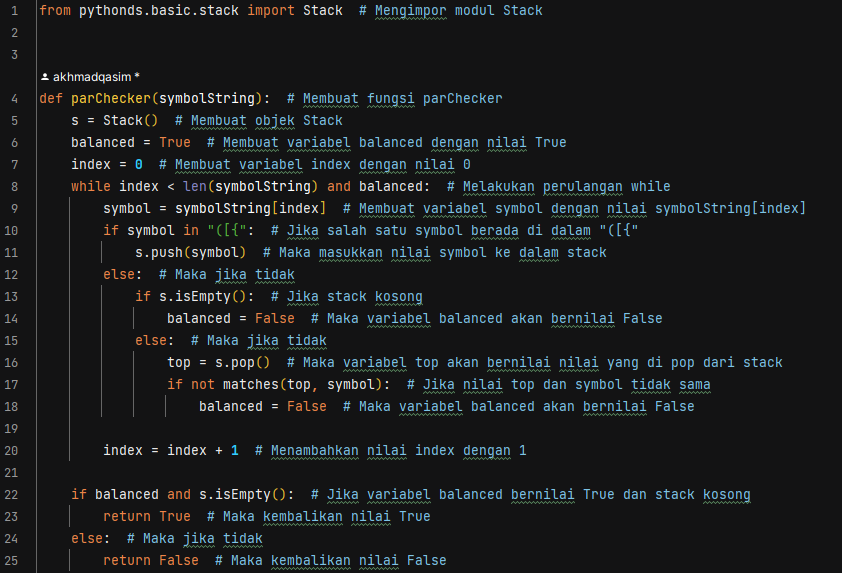
1. **Stack**  
   
   1. **Berikan tampilan output dari perintah diatas!**  
      Tampilan Output:  
      Graphical user interface, text

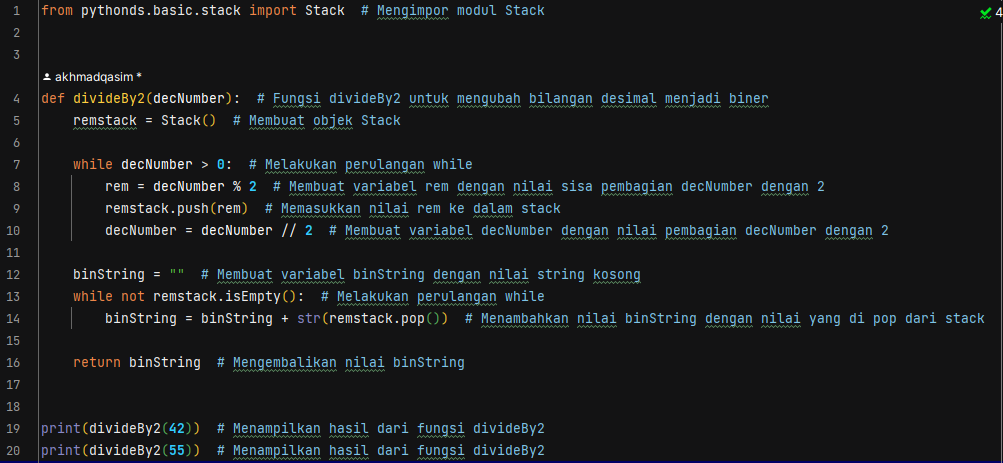
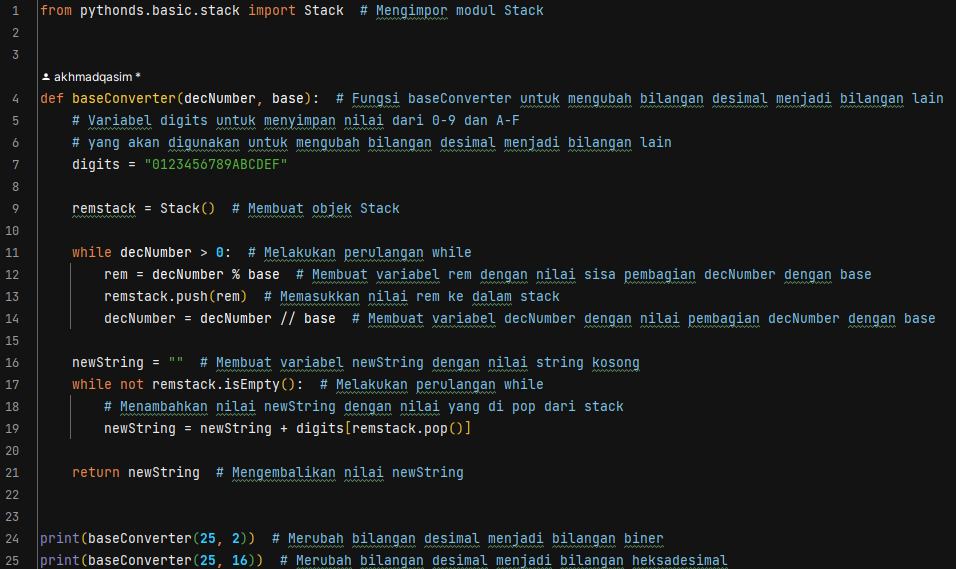
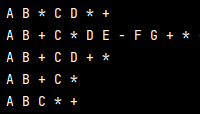
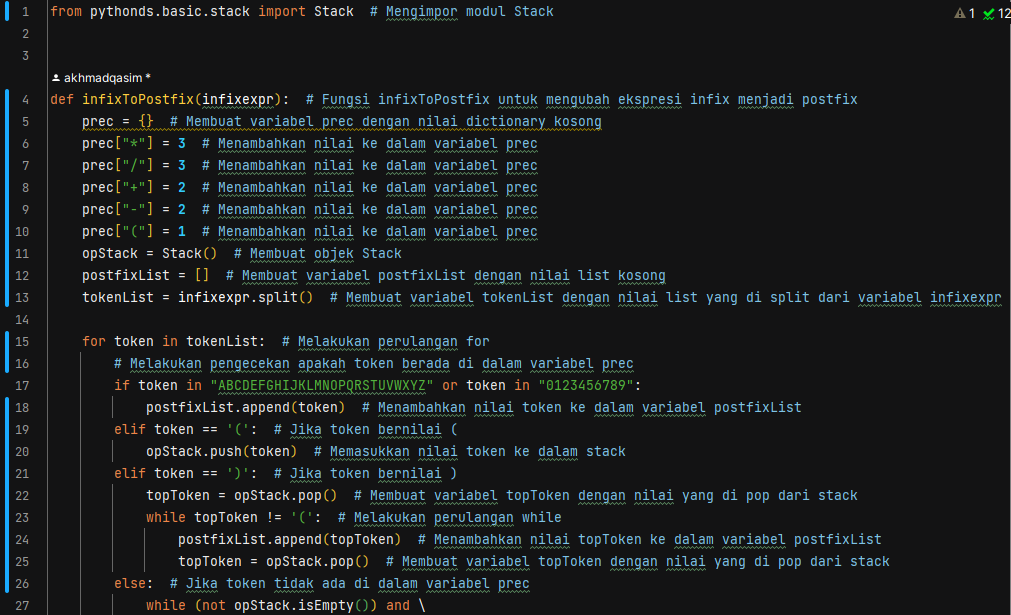
      Description automatically generated
   2. **Analisa hasil dari perintah diatas!**  
      Analisa syntax:  
      Text

      Description automatically generatedText

      Description automatically generated
2. **Stack**
   1. **Jalankan perintah berikut dan analisa hasil output.**  
        
      Tampilan Syntax:  
      Text

      Description automatically generated   
        
      Tampilan Output:  
        
        
      Nilai yang pertama bernilai True karena tanda kurung yang muncuk seimbang, sedangkan nilai yang kedua kekurangan 1 tanda kurung ‘)’ yang berarti tidak seimbang.
   2. **Jalankan perintah berikut dan berikan alasan mengapa menghasilkan ouput False!**  
        
      Tampilan Syntax:  
      Text

      Description automatically generated  
        
        
      Tampilan Output:  
        
        
      Pada output baris pertama dan baris kedua mengembalikan nilai False dikarenakan pada struktur kontrol hanya mengecek tanda kurung saja.
3. **Stack**
   1. **Berikan tampilan ouput dari perintah diatas!**  
      Tampilan Output:  
      
   2. **Jelaskan perbedaan fungsi diatas dengan fungsi yang terdapat di Percobaan & Latihan 6.2!**Tampilan syntax:  
      A screenshot of a computer

      Description automatically generated with medium confidence  
        
      Terdapat perbedaan antara fungsi pada percobaan & latihan 6.2 dengan percobaan & latihan 6.3 dimana terdapat tambahan pada struktur kontrol dan fungsi matcher yang dapat mengenali tanda kurung, kurung siku, dan kurung kurawal.
4. **Stack**
   1. **Jalankan fungsi diatas menggunakan perintah berikut!**  
        
        
      Tampilan Syntax:  
      
   2. **Analisa hasil ouput diatas!**  
      Tampilan Output:  
        
        
      Hasil output merupakan hasil operasi fungsi divideBy2 yang merubah bilangan desimal menjadi bilangan biner.
5. **Stack**
   1. **Jelaskan perbedaan ouput dari kedua perintah diatas (ouput baris 19 dan 20)!**  
      Tampilan Syntax:  
        
        
      Tampilan Output:  
        
        
      Perbedaan antara baris 24 dan 25 adalah kita menggunakan fungsi baseConverter untuk mengubah bilangan desimal menjadi bilangan lain. Pada baris 24 kita menggunakan parameter base 2 yang berarti merubah bilangan desimal menjadi bilangan biner, sedangkan pada baris 25 kita menggunakan fungsi baseConverter dengan parameter base 16 yang berarti kita akan mengubah bilangan desimal menjadi bilangan heksadesimal.
6. **Infix, Prefix, dan Postfix**
   1. **Berikan hasil output dari perintah diatas!**  
      Tampilan Output:  
      
   2. **Berikan penjelasan tiap baris pada coding diatas!**  
      Tampilan Syntax:  
       Graphical user interface, text

      Description automatically generated
7. **Infix, Prefix, dan Postfix**
   1. **Berikan hasil output dari perintah diatas!**  
      Tampilan Output:  
      
   2. **Berikan penjelasan tiap baris pada coding diatas!**  
      Tampilan Syntax:  
      Text

      Description automatically generated Text

      Description automatically generated

**Kesimpulan:**

Di dalam pemrograman, "stack" (tumpukan) adalah struktur data yang digunakan untuk menyimpan dan mengakses elemen dengan aturan Last-In-First-Out (LIFO). Artinya, elemen yang terakhir dimasukkan ke dalam stack adalah elemen yang pertama kali diambil.

Dalam matematika, "infix", "prefix" dan "postfix" adalah notasi yang digunakan untuk menulis ekspresi aritmatika. Contoh ekspresi aritmatika adalah "3 + 4 \* 2". Notasi infix adalah notasi yang paling umum digunakan, yaitu operator ditempatkan antara dua operand. Notasi prefix dan postfix adalah notasi yang operatornya ditempatkan sebelum atau sesudah operand.

Contoh notasi infix: 3 + 4 \* 2  
Contoh notasi prefix: + 3 \* 4 2  
Contoh notasi postfix: 3 4 2 \* +

Untuk mengubah notasi infix menjadi prefix atau postfix, kita perlu menggunakan struktur data stack. Dalam konversi infix ke postfix, kita menggunakan stack untuk menyimpan operator dan operand sementara sebelum menulis ulang ekspresi dalam notasi postfix. Dalam konversi infix ke prefix, kita juga menggunakan stack dengan aturan yang sama, hanya saja penulisan ulang dilakukan dalam notasi prefix.