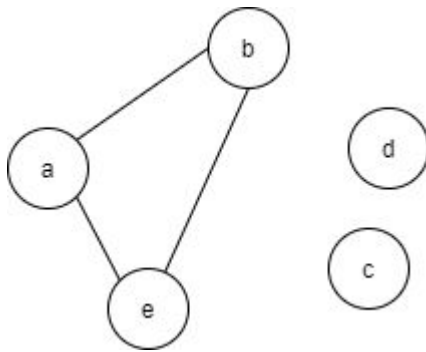


Praktikum Matematika Informatika - 11 (3)

Latihan Adjacency List

Materi:

- Perhatikan **graf** berikut:



- **Salah satu cara** merepresentasikan sebuah **sisi** adalah sebagai berikut:

```
# graf di atas dibuat menjadi adjacency list berikut:
graf1 = {
    'a': ['e','b'],
    'b': ['a','e'],
    'c': [],
    'd': [],
    'e': ['a','b']
}

# sisi a ke b bisa diwakili dengan tipe set (himpunan):
sisi_a_b = {'a', 'b'} # atau {'b', 'a'}

# begitu juga dengan sisi lainnya
sisi_e_a = {'a', 'e'} # atau {'e', 'a'}
sisi_e_b = {'e', 'b'} # atau {'b', 'e'}

# sehingga sisi-sisi dari graf tersebut bisa dinyatakan sebagai
list:
list_sisi_graf1 = [{'a', 'b'}, {'e', 'a'}, {'b', 'e'}]
```

Referensi perintah (boleh dicoba dulu sebelum mengerjakan soal-soal):

- Mengecek apakah sebuah nilai merupakan (atau bukan merupakan) elemen dari suatu list:

```
list1 = [3,4,5]

if 3 in list1:
    print('Ada nilai 3') # masuk ke sini
else:
    print('Tidak ada nilai 3')

if 6 not in list1:
    print('Tidak ada nilai 6') # masuk ke sini
else:
    print('Ada nilai 6')
```

Soal 3 (sisi)

Masih terkait soal nomor 1 (menggunakan graf yang sama). Buat sebuah program yang berisi poin-poin di bawah ini:

- a. Buat dan isi sebuah **list** yang berisi **sisi-sisi** graf tersebut. Untuk memudahkan, **sisi** dinyatakan saja sebagai sebuah **set**.

Contoh: sisi yang menghubungkan Eloy dan Paulo direpresentasikan dalam bentuk **set** {'Eloy', 'Paulo'} atau {'Paulo', 'Eloy'}

Pastikan di dalam list sisi sebuah sisi **hanya muncul 1 kali**. Misalnya: {'Igor', 'Andreas'} sama dengan {'Andreas', 'Igor'}, dan cukup muncul 1 saja.

```
# definisi graf
# ...

# inisialisasi list sisi (list kosong)
# ...

# proses mengisi list sisi
# ...
```

```
# cetak list sisi
print(list_sisi)

# contoh hasil (urutan belum tentu sama):
# [{'Igor', 'Max'}, {'Paulo', 'Max'}, {'Igor', 'Paulo'}, {'Igor',
'Andreas'}, {'Paulo', 'Andreas'}, {'Paulo', 'Derrick'}, {'Eloy',
'Paulo'}, {'Andreas', 'Derrick'}, {'Bella', 'Eloy'}, {'Roxanne',
'Gloria'}]

# cetak banyaknya sisi
print(len(list_sisi))
# 10
```

Panduan langkah:

- Lakukan **pengulangan** terhadap **key** dari graf.
- Di dalamnya (pada setiap pengulangan dari key) ada **pengulangan** lagi terhadap **value** dari key.
- Di dalam pengulangan kedua ini, buat sebuah **set** yang terdiri dari 2 elemen, yaitu (1) **simpul key**, dan (2) **simpul** salah satu **tetangga** dari simpul key.
- **Tambahkan** set (yang sudah berisi 2 elemen) **ke dalam list sisi**, hanya **jika** set tersebut **belum ada** di list.
- **Pengecekan** apakah **set** sudah **ada di list** menggunakan perintah "if ... not in ...".
- Catatan:
 - karena elemen list merupakan set, jika sudah ada elemen set {1,2} di list, elemen set {1,2} atau {2,1} tidak akan dimasukkan.
 - jika elemen list berupa tuple (atau list juga), elemen (1,2) tetap akan dimasukkan meskipun sudah ada (2,1). Boleh dicoba.