

LAPORAN PRAKTIKUM 7

Worksheet 7

Analysis Algorithm

GREEDY ALGORITHM

STUDI KASUS : ALGORITMA KRUSKAL DAN ALGORITMA PRIM



Disusun oleh :

Mohammad Dhikri

140810180075

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS PADJADJARAN
2020**

PENDAHULUAN

Minimum Spanning Tree

Minimum Spanning Tree (MST) adalah subset dari tepi-tepi dari graf tak-berbobot tepi-terhubung yang terhubung, yang menghubungkan semua simpul secara bersamaan, tanpa siklus apa pun dan dengan total bobot edge seminimum mungkin. Sebuah tree memiliki satu jalur yang menghubungkan dua simpul. Spanning tree dari graf adalah sebuah tree yang:

- Berisi semua simpul graf asal
- Edge yang dibentuk mencapai semua simpul
- Acyclic, artinya graf tidak memiliki node yang kembali ke dirinya sendiri

Algoritma Kruskal

Algoritma Kruskal adalah algoritma *minimum spanning tree* yang tujuannya menemukan *edge* dengan bobot sekecil mungkin yang menghubungkan 2(dua) *tree* di *forest*. Berikut cara kerja dari algoritma kruskal.

1. Mengurutkan *edges* dari graf menurut bobotnya.
2. T masih kosong. (: *tree*)
3. Pilih sisi *e* (*edge*) dengan bobot minimum yang tidak membentuk sirkuit di T. Masukkan *e* ke dalam T.
4. Ulangi langkah 2 sebanyak $n-1$ kali.

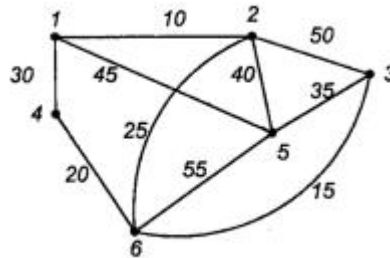
Algoritma Prim

Algoritma Prim merupakan algoritma *minimum spanning tree* yang beroperasi dengan membangun *tree* dengan satu simpul pada satu waktu, dari titik awal yang berubah – ubah, pada setiap langkah menambahkan *cost* yang sekecil mungkin dari *tree* ke titik lain. Berikut proses algoritma prim.

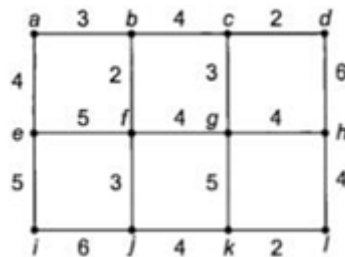
1. Menginisialisasi *tree* dengan *simpul* tunggal, dipilih secara acak dari graf.
2. Membuat *tree* dengan satu *edge*: dari *edge* yang menghubungkan *tree* ke simpul yang belum ada di *tree*, temukan *edge* dengan bobot minimum, dan pindahkan ke *tree*.
3. Mengulangi langkah 2(dua) sampai semua simpul terdapat di *tree*.

TUGAS ANDA

1. Cari minimum spanning tree pada graf di bawah dengan Algoritma Kruskal. Jelaskan langkah demi langkah sampai graf membentuk minimum spanning tree.



2. Gambarkan 3 buah minimum spanning tree yang berbeda beserta bobotnya untuk graf di bawah dengan Algoritma Prim. Jelaskan setiap langkah untuk membangun minimum spanning tree.



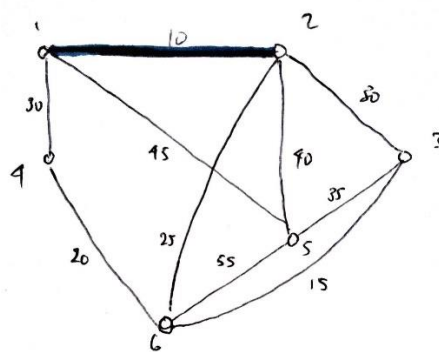
4. Apakah semua minimum spanning tree T dari graf terhubung G harus mengandung jumlah sisi yang sama? Jelaskan alasannya (bukan dengan contoh).

Jawab :

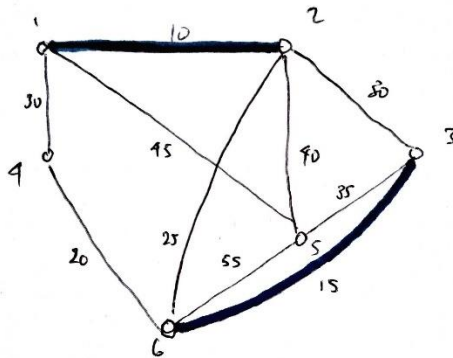
1. Cari minimum spanning tree pada graf di bawah dengan Algoritma Kruskal. Jelaskan langkah demi langkah sampai graf membentuk minimum spanning tree.
 - a. Menghapus semua loop dan parallel edges
 - b. Mengatur semua edge pada graf dari yang terkecil ke yang terbesar

Edges	Cost
1, 2	10
3, 6	15
4, 6	20
2, 6	25
1, 4	30
3, 5	35
2, 5	40
1, 5	45
2, 3	50
5, 6	55

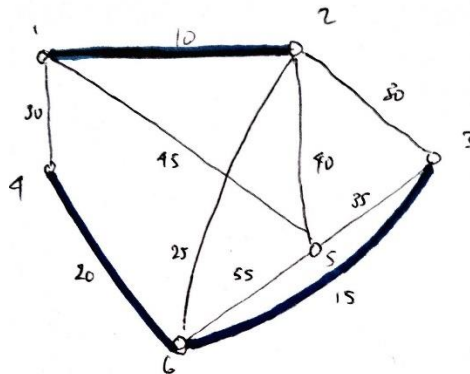
- c. Menambahkan edge dengan bobot paling kecil. 1 dan 2 terhubung dahulu karena edge cost nya hanya 10.



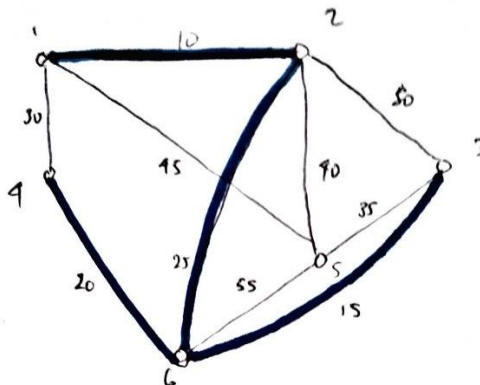
- d. Menghubungkan 3 dan 6 karena memiliki edge cost 15.



- e. Menghubungkan edges dengan cost 20

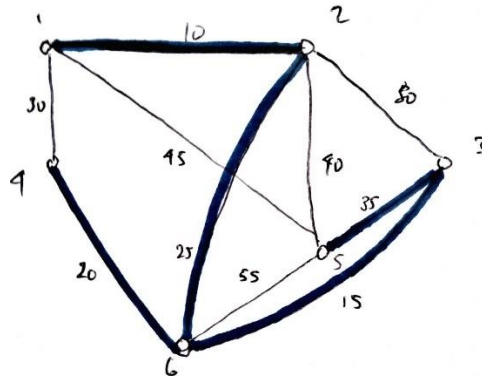


- f. Menghubungkan edges dengan cost 25

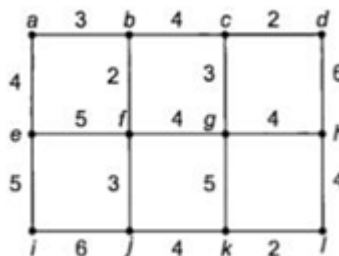


- g. Menghubungkan edges dengan cost 30, namun karena 1, 4 sudah termasuk

- dalam 1 tree maka step ini tidak menghubungkan
- h. Menghubungkan edges dengan cost 35

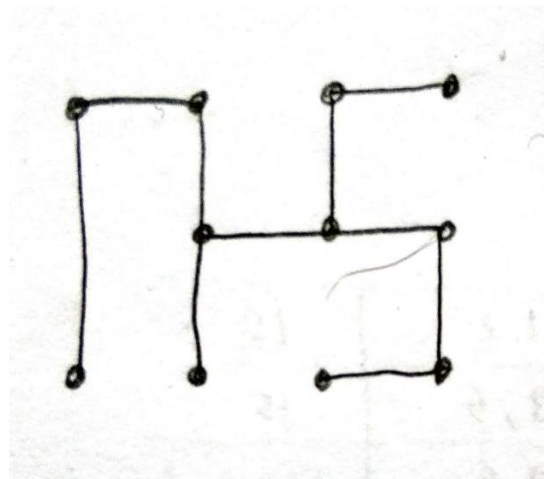
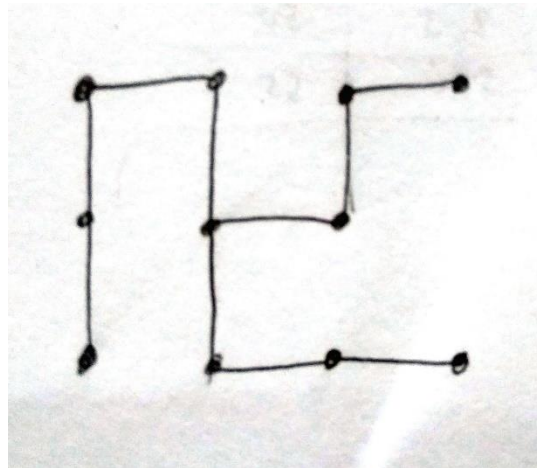


2. Algoritma Prim merupakan algoritma yang beroperasi dengan membangun tree dengan satu simpul pada satu waktu, dari titik awal yang berubah – ubah, pada setiap langkah menambah *cost* yang sekecil mungkin dari tree ke titik lain. Berikut langkah pengerjaannya :



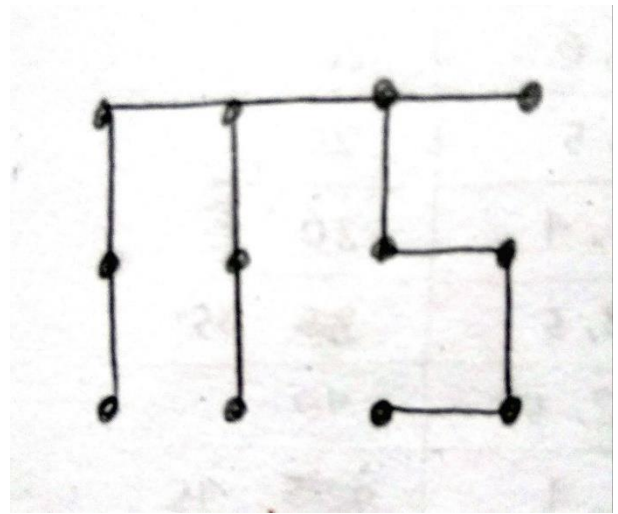
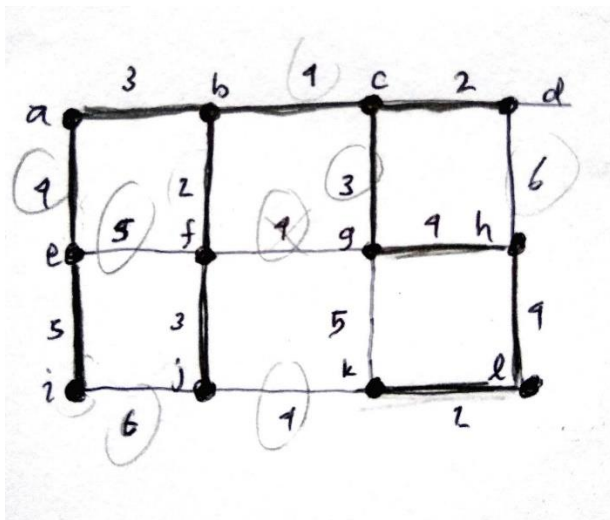
- a. Menginisialisasi tree dengan simpul tunggal, dipilih secara acak dari graf.
 - i. Diambil simpul pertama adalah a.
- b. Membuat tree dengan satu edge : dari edge yang menghubungkan tree ke simpul yang belum ada di tree, temukan edge dengan bobot minimum, pindahkan ke tree.
 - i. Simpul a terbagi 2 cabang yaitu ke simpul b, dan e. Yang terkecil adalah 3, maka garis menuju b.
- c. Mengulangi langkah 2 sampai semua simpul terdapat di tree
 - i. Membaca simpul a dan b.
 1. Simpul a \rightarrow e = 4
 2. Simpul b \rightarrow f = 2 (terkecil)
 3. Simpul b \rightarrow c = 4
 - Simpul dari b menuju f
 - ii. Membaca simpul a, b, dan f
 1. Simpul a \rightarrow e = 4
 2. Simpul b \rightarrow c = 4
 3. Simpul f \rightarrow e = 5
 4. Simpul f \rightarrow j = 3 (terkecil)
 5. Simpul f \rightarrow g = 4
 - Simpul dari f menuju j
 - iii. Membaca simpul a, b, f, dan j

1. Membaca simpul a, b, f, j, k



3. Membaca simpul a, b, c, f, j
 - a. Simpul a \rightarrow e = 4
 - b. Simpul c \rightarrow d = 2 (terkecil)
 - c. Simpul c \rightarrow g = 3
 - d. Simpul f \rightarrow g = 4
 - e. Simpul f \rightarrow e = 5
 - f. Simpul j \rightarrow I = 6
 - g. Simpul j \rightarrow k = 4
 - i. Simpul ke d

Hasil Akhir



3. Apakah semua minimum spanning tree T dari graf terhubung G harus mengandung jumlah sisi yang sama? Jelaskan alasannya (bukan dengan contoh).

Jawab : Harus mengandung jumlah sisi yang sama karena untuk membentuk spanning tree dan karena syarat spanning tree tidak boleh membentuk cycle maka pasti jumlah sisi dari tree yang terbentuk pasti sama yaitu jumlah simpul dikurangi 1.

Daftar Pustaka

Modul Praktikum 7

<https://www.youtube.com/watch?v=Uj47dxYPow8>

<https://www.youtube.com/watch?v=Yo7sddEVONg>