

LAPORAN PRAKTIKUM ANALISIS ALGORITMA



DISUSUN OLEH

SALMA ALIFIA SHAFIRA

140810180058

UNIVERSITAS PADJADJARAN

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

TEKNIK INFORMATIKA

2020

PENDAHULUAN

Stable Matching Problem (SMP) adalah problem algoritmik yang memberikan ilustrasi mengenai berbagai tema yang dipelajari di analisis algoritma ini. Algoritma ini muncul dari beberapa problem praktis. Oleh karena itu supaya problemnya jelas dan penyelesaian tepat perlu dilakukan 3 langkah berikut:

- Mencermati problem
- Memformulasikan problem
- Mendesain algoritma

Stable Matching Problem berasal, sebagian, pada tahun 1962, ketika David Gale dan Lloyd Shapley, dua matematika ekonom, mengajukan pertanyaan:

Bisakah seseorang merancang sebuah perguruan tinggi proses penerimaan, atau proses perekrutan pekerjaan, itu mandiri (otomatis)?

Inti dari proses aplikasi adalah interaksi antara dua jenis pihak yang berbeda: **perusahaan dan pelamar**. Setiap pelamar memiliki daftar preferensi perusahaan yang ingin dimasuki, dan setiap perusahaan- setelah aplikasi masuk-membentuk daftar preferensi akan pelamarnya. Berdasarkan preferensi ini, perusahaan memberikan penawaran kepada beberapa pelamar mereka, pelamar memilih penawaran mana yang akan mereka terima.

Bagaimana jika tidak dilakukan secara otomatis? Kemungkinan resiko kecurangan tinggi.

Jadi inilah pertanyaan yang diajukan Gale dan Shapley: Diberikan seperangkat preferensi di antara pemberi kerja dan pelamar, dapatkah kami menetapkan pelamar untuk pemberi kerja sehingga untuk setiap pemberi kerja E , dan setiap pelamar A yang tidak dijadwalkan bekerja untuk E , setidaknya satu dari dua hal berikut ini yang terjadi?

- (i) E lebih memilih setiap satu dari daftar pelamar yang diterima(A); atau
- (ii) A lebih suka situasinya saat ini daripada bekerja untuk pemberi kerja E

Jika ini berlaku, hasilnya stabil: kepentingan pribadi individu akan mencegah kesepakatan pemohon/pemberi kerja dibuat dibalik layar. Gale dan Shapley mengembangkan solusi algoritmik yang tajam untuk problem ini, yang akan kita pelajari.

Studi Kasus

SMP ini dapat dilihat juga sebagai problem menyusun sistem dimana setiap pria dan wanita akhirnya bisa berpasangan.

Jadi pertimbangkan satu set $M = \{N_1, \dots, N_n\}$ dari n pria, dan satu set $W = \{w_1, \dots, w_n\}$ dari n wanita. Produk kartesius $M \times W$ menunjukkan set dari semua pasangan bentuk yang mungkin dipasangkan (N, w), di mana $N \in M$ dan $w \in W$.

Matching S adalah seperangkat pasangan yang dipasangkan, masing-masing dari $M \times W$, dengan properti yang masing-masing anggota M dan setiap anggota W muncul di paling banyak satu pasangan di S .

Dipandu oleh motivasi awal kita dalam hal pemberi kerja dan pelamar, kita harus khawatir tentang situasi berikut: Ada dua pasangan (N, w) dan (N', w') dalam S (seperti yang digambarkan pada Gambar 1.1) dengan properti bahwa N lebih suka w' daripada w , dan w' lebih suka N ke N' . Dalam hal ini, tidak ada yang bisa menghentikan N dan w' meninggalkan pasangan mereka saat ini dan

pergi bersama; set pernikahan menjadi tidak self-enforcing.

Tujuan kita adalah mengembalikan serangkaian pasangan tanpa ketidakstabilan (harus stabil).

Kita akan mengatakan bahwa S stabil jika

- (1) Perfect (1 laki-laki tepat berhubungan dengan satu perempuan), dan
- (2) tidak ada ketidakstabilan sehubungan dengan S

Syarat:

- Perfect Match: semua orang dicocokkan secara monogami.
 - (1) Setiap pria mendapatkan satu wanita.
 - (2) Setiap wanita mendapatkan satu pria.
- Stable Matching: pencocokan sempurna tanpa pasangan tidak stabil.

Stable Matching Problemnya:

Dengan daftar preferensi pria dan wanita, temukan sebuah stable matching jika ada.

Contoh 1

- Pertanyaan: Jika dipasangkan X-C, Y-B, dan Z-A, apakah stabil?

	favorite ↓	1st	2nd	3rd	least favorite ↓	1st	2nd	3rd	least favorite ↓
Xavier		Amy	Bertha	Clare		Amy	Yancey	Xavier	Zeus
Yancey		Bertha	Amy	Clare		Bertha	Xavier	Yancey	Zeus
Zeus		Amy	Bertha	Clare		Clare	Xavier	Yancey	Zeus

Men's Preference Profile *Women's Preference Profile*

- Jawaban: Tidak. Bertha & Xavier akan putus

Contoh 2

- Pertanyaan: Jika dipasangkan X-A, Y-B, dan Z-C, apakah stabil?

	favorite ↓	1st	2nd	3rd	least favorite ↓	1st	2nd	3rd	least favorite ↓
Xavier		Amy	Bertha	Clare		Amy	Yancey	Xavier	Zeus
Yancey		Bertha	Amy	Clare		Bertha	Xavier	Yancey	Zeus
Zeus		Amy	Bertha	Clare		Clare	Xavier	Yancey	Zeus

Men's Preference Profile *Women's Preference Profile*

- Jawaban: Ya

Worksheet 01

Jika Anda belum mengerjakan worksheet 01 di kelas, maka Anda dapat mengerjakannya di awal praktikum. Anda diberikan waktu 30 menit untuk menyelesaikan persoalan pada worksheet 01. Bagi Anda yang sudah mengerjakan, Anda dapat langsung mengerjakan tugas praktikum dan mencocokkan hasil worksheet 01 Anda dengan tugas praktikum.

Worksheet 01

Dengan Algoritma Gale-Shapley, cari himpunan stable-matching yang sesuai dengan preference-lists berikut ini. Gunakan processor terhebat yang Anda miliki (otak) untuk mengikuti algoritma G-S dan output tidak perlu diuraikan per-looping tetapi Anda harus memahami hasil setiap looping.

Men's Preferences Profile

	0 th	1 st	2 nd	3 rd	4 th
Victor	Bertha	Amy	Diane	Erika	Clare
Wyatt	Diane	Bertha	Amy	Clare	Erika
Xavier	Bertha	Erika	Clare	Diane	Amy
Yancey	Amy	Diane	Clare	Bertha	Erika
Zeus	Bertha	Diane	Amy	Erika	Clare

Women's Preferences Profile

	0 th	1 st	2 nd	3 rd	4 th
Amy	Zeus	Victor	Wyatt	Yancey	Xavier
Bertha	Xavier	Wyatt	Yancey	Victor	Zeus
Clare	Wyatt	Xavier	Yancey	Zeus	Victor
Diane	Victor	Zeus	Yancey	Xavier	Wyatt
Erika	Yancey	Wyatt	Zeus	Xavier	Victor

Tugas Praktikum

- Ubahlah pseudocode algoritma G-S pada worksheet 01 ke dalam program menggunakan bahasa C++
- Gunakan table pria sebagai table acuan untuk memudahkan Anda menentukan pasangannya.
- Cocokkan jawaban Anda pada worksheet 01 dengan hasil program yang Anda buat
- Jika ada yang berbeda tuliskan bagian mana yang berbeda dan analisislah (Poin ini disampaikan pada bagian Analisis Algoritma) yang sudah disiapkan.

Tahap	Man	Woman	Free
1	Victor	Bertha	
2	Wyatt	Diane	
3	Xavier	Bertha	Victor
4	Victor	Amy	
5	Yancey	Amy	Yancey
6	Yancey	Diane	Wyatt
7	Wyatt	Bertha	Wyatt
8	Wyatt	Amy	Wyatt
9	Wyatt	Clare	
10	Zeus	Bertha	Zeus
11	Zeus	Diane	Yancey
12	Yancey	Clare	Yancey
13	Yancey	Bertha	Yancey
14	Yancey	Erika	

Analisis Algoritma

Jawablah pertanyaan berikut:

1. Apakah jawaban Anda di Worksheet 01 dan Program sama persis? Jika Tidak? Kenapa?

Ya.

Anda diminta untuk membuktikan algoritma G-S benar dengan menjawab pertanyaan berikut:

Fakta (1.1):

Seorang wanita tetap bertunangan dari titik di mana dia menerima proposal pertamanya; dan urutan mitra yang bertunangan dengannya menjadi lebih baik dan lebih baik lagi (hal ini sesuai dengan daftar preferensi wanita). → tidak perlu dipertanyakan

Fakta (1.2):

Urutan wanita yang dilamar pria lebih buruk dan lebih buruk lagi (hal ini sesuai dengan daftar preferensi pria). → tidak perlu dipertanyakan

Teorema (1.3):

Algoritma G-S berakhir setelah paling banyak n^2 iterasi menggunakan While Loop. Buktikan!

Pada saat menggunakan while loop memiliki kemajuan, yaitu dimana pria lajang melamar wanita berikutnya dalam daftar pilihannya, lalu seseorang yang belum pernah ia ajukan sebelumnya. Karena ada n laki-laki dan setiap daftar preferensi memiliki panjang n , ada paling banyak proposal n^2 yang dapat terjadi. Jadi jumlah dalam iterasi yang bisa terjadi paling banyak adalah n^2

Algoritma

Teorema (1.4):

Jika seorang pria bebas di beberapa titik dalam eksekusi algoritma, maka ada seorang wanita yang

belum dia ajak bertunangan.
Buktikan!

Buktinya berdasarkan kontradiksi. Misalkan ada waktu tertentu dalam pelaksanaan algoritma ketika seorang pria lajang, namun telah mengusulkan kepada setiap wanita. Ini berarti, setiap wanita telah diusulkan setidaknya satu kali. Dengan teori 1, mendapatkan bahwa setiap wanita bertunangan. Jadi, kita telah melibatkan n wanita dan karenanya n laki-laki bertunangan, yang menyiratkan bahwa m juga terlibat bertentangan dengan asumsi bahwa m adalah lajang.

Teorema (1.5):

Himpunan S yang dikembalikan saat terminasi adalah *perfect matching* Buktikan!

Karena setiap laki-laki berpasangan dengan setiap wanita

Teorema (1.6):

Sebuah eksekusi algoritma G-S mengembalikan satu set pasangan S. Set S adalah pasangan yang stabil. Buktikan!

Menunjukkan bahwa pencocokan yang dikembalikan adalah pencocokan sempurna. Buktinya dengan kontradiksi. Misalkan tidak, maka ada seorang pria yang masih lajang di akhir algoritma. Menurut teori 2, itu berarti m belum melamar beberapa wanita. Tetapi kemudian, algoritma tidak akan keluar dari pengulangan loop, menghasilkan kontradiksi yang diinginkan.

Menunjukkan bahwa pencocokan yang dikembalikan stabil. Lagi-lagi buktinya dengan kontradiksi. Misalkan ada laki-laki m dan m' dan wanita w dan w' sehingga (m, w) dan (m', w') berada di S, tetapi m lebih suka w' ke w dan w' lebih suka m ke m'. Dengan algoritma, w adalah wanita terakhir yang saya ajukan. Karena m lebih suka w' ke w, m harus sudah mengusulkan ke w' sebelum usulannya ke w. Pada saat itu, atau nanti, w' bertunangan dengan seorang pria, katakanlah m'', yang ia sukai lebih dari m. Pada akhirnya, w' bertunangan dengan m'. Oleh teori 1, menemukan bahwa w' lebih memilih m' daripada m'' dan lebih memilih m'' daripada m; ini menyiratkan bahwa w' lebih suka m' daripada m, bertentangan dengan asumsi bahwa w' lebih memilih m daripada m'.

```
C:\Users\salma\Desktop\penjodohan.exe
-----
Pria    Wanita
-----
Victor  Amy
Xavier  Bertha
Wyatt   Clare
Zeus    Diane
Yancey  Erika
-----

-----
Process exited after 0.1708 seconds with return value 0
Press any key to continue . . .
```

```

1  /*
2  Nama : Salma Alifia Shafira
3  NPM : 140810180058
4  Kelas : B
5  */
6
7  #include <iostream>
8  #include <string.h>
9  #include <stdio.h>
10
11 using namespace std;
12 #define N 5
13
14 bool chooseMen(int prioritas[2 * N][N], int w, int m, int m1)
15 {
16     for (int i = 0; i < N; i++)
17     {
18         if (prioritas[w][i] == m1)
19             return true;
20
21         if (prioritas[w][i] == m)
22             return false;
23     }
24 }
25
26 void smp(int prioritas[2 * N][N])
27 {
28     int pasanganWanita[N];
29     bool priaJomblo[N];
30
31     memset(pasanganWanita, -1, sizeof(pasanganWanita));
32     memset(priaJomblo, false, sizeof(priaJomblo));
33     int jumlahJomblo = N;
34
35     while (jumlahJomblo > 0)
36     {
37         int m;
38         for (m = 0; m < N; m++)
39             if (priaJomblo[m] == false)
40                 break;
41
42         for (int i = 0; i < N && priaJomblo[m] == false; i++)
43         {
44             int w = prioritas[m][i];
45
46             if (pasanganWanita[w - N] == -1)
47             {
48                 pasanganWanita[w - N] = m;
49                 priaJomblo[m] = true;
50                 jumlahJomblo--;
51             }
52             else
53             {
54                 int m1 = pasanganWanita[w - N];
55
56                 if (chooseMen(prioritas, w, m, m1) == false)
57                 {
58                     pasanganWanita[w - N] = m;
59                     priaJomblo[m] = true;
60                     priaJomblo[m1] = false;
61                 }
62             }
63         }
64
65         cout << "-----" << endl;
66         cout << "  Pria   Wanita  " << endl;
67         cout << "-----" << endl;
68         string man;
69         string woman;
70         for (int i = 0; i < N; i++)
71         {
72             if (i < N)
73             {
74                 if (pasanganWanita[i] == 0)
75                     man = "Victor";
76                 if (pasanganWanita[i] == 1)
77                     man = "Wyatt";
78
79                 if (pasanganWanita[i] == 2)
80                     man = "Xavier";
81                 if (pasanganWanita[i] == 3)
82                     man = "Yancey";
83                 if (pasanganWanita[i] == 4)
84                     man = "Zeus";
85
86                 if (i == 0)
87                     woman = "Amy";
88                 if (i == 1)
89                     woman = "Bertha";
90                 if (i == 2)
91                     woman = "Clare";
92                 if (i == 3)
93                     woman = "Diane";
94                 if (i == 4)
95                     woman = "Erika";
96
97                 cout << "  " << man << "\t " << woman << endl;
98             }
99             cout << "-----" << endl;
100         }
101     }
102
103 int main()
104 {
105     int prioritas[2 * N][N] = {{6, 5, 8, 9, 7},
106                                {8, 6, 5, 7, 9},
107                                {6, 9, 7, 8, 5},
108                                {5, 8, 7, 6, 9},
109                                {6, 8, 5, 9, 7},
110                                {4, 0, 1, 3, 2},
111                                {2, 1, 3, 0, 4},
112                                {1, 2, 3, 4, 0},
113                                {0, 4, 3, 2, 1},
114                                {3, 1, 4, 2, 0}};
115
116     smp(prioritas);
117
118     return 0;
119 }

```