

LAPORAN PRAKTIKUM  
STABLE MATCHING PROBLEM



Alvin  
140810180013

PROGRAM STUDI S-1 TEKNIK INFORMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS PADJADJARAN  
2020

## 1. TUJUAN PRAKTIKUM

Tujuan praktikum mata kuliah Analisis Algoritma kali ini adalah untuk mengetahui Algoritma Gale-Shapley dalam menyelesaikan masalah pencocokan ataupun otomatisasi perekrutan / perjodohan serta untuk menganalisis output atau keluaran dari Algoritma Gale - Shapley

## 2. MATERI PRAKTIKUM

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.

## Soal

### Worksheet 01

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

Hasil yang diharapkan:

Men's Preferences Profile					
	0 <sup>th</sup>	1 <sup>st</sup>	2 <sup>nd</sup>	3 <sup>rd</sup>	4 <sup>th</sup>
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 <sup>th</sup>	1 <sup>st</sup>	2 <sup>nd</sup>	3 <sup>rd</sup>	4 <sup>th</sup>
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

Dengan pseudocodenya sebagai berikut:

---

```
Initially all  $m \in M$  and  $w \in W$  are free
While there is a man  $m$  who is free and hasn't proposed to
every woman
    Choose such a man  $m$ 
    Let  $w$  be the highest-ranked woman in  $m$ 's preference list
    to whom  $m$  has not yet proposed
    If  $w$  is free then
         $(m, w)$  become engaged
    Else  $w$  is currently engaged to  $m'$ 
        If  $w$  prefers  $m'$  to  $m$  then
             $m$  remains free
        Else  $w$  prefers  $m$  to  $m'$ 
             $(m, w)$  become engaged
             $m'$  becomes free
        Endif
    Endif
Endwhile
Return the set  $S$  of engaged pairs
```

---

Pertanyaan:

1. Ubahlah pseudocode algoritma G-S pada worksheet 01 ke dalam program menggunakan bahasa C++.
2. Gunakan table pria sebagai table acuan untuk memudahkan Anda menentukan pasangannya.
3. Cocokkan jawaban Anda pada worksheet 01 dengan hasil program yang Anda buat.
4. Jika ada yang berbeda tuliskan bagian mana yang berbeda dan analisislah:
  - a. Apakah jawaban Anda di Worksheet 01 dan Program sama persis? Jika Tidak? Kenapa?
  - b. 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!
- Teorema (1.4): Jika seorang pria bebas di beberapa titik dalam eksekusi algoritma, maka ada seorang wanita yang belum dia ajak bertunangan. Buktikan!
- Teorema (1.5): Himpunan S yang dikembalikan saat terminasi adalah perfect matching Buktikan!
- Teorema (1.6): Sebuah eksekusi algoritma G-S mengembalikan satu set pasangan S. Set S adalah pasangan yang stabil. Buktikan!

### Jawaban

---

#### Men's Preferences Profile

	0 <sup>th</sup>	1 <sup>st</sup>	2 <sup>nd</sup>	3 <sup>rd</sup>	4 <sup>th</sup>
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 <sup>th</sup>	1 <sup>st</sup>	2 <sup>nd</sup>	3 <sup>rd</sup>	4 <sup>th</sup>
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

Dari tabel dapat disimpulkan proses yang terjadi adalah sebagai berikut:

- Victor dengan Bertha.
- Wyatt dengan Diane.
- Xavier dengan Bertha, karena Bertha lebih menyukai Xavier sehingga Victor dilepas.
- Yancey dengan Amy.
- Zeus dengan Bertha, namun karena Bertha sudah dengan Xavier dan lebih memilih Xavier maka Zeus tidak mendapatkan pasangan.
- Victor dengan Amy, karena Amy lebih menyukai Victor sehingga Yancey dilepas.
- Yancey dengan Diane, karena Diane lebih menyukai Yancey sehingga Wyatt dilepas.
- Zeus dengan Diane, karena Diane lebih memilih Zeus sehingga Yancey dilepas.
- Wyatt dengan Clare.
- Yancey dengan Erika.

Pasangan yang terbentuk adalah

- Victor – Amy
- Wyatt - Clare
- Xavier – Bertha
- Yancey - Erika
- Zeus – Diane

Program untuk pseudocode di atas:

```
/*
Nama      : Alvin
NPM       : 140810180013
Kelas    : A
Tugas     : Stable Matching Problem
*/

#include <iostream>
#include <string.h>
#include <stdio.h>

using namespace std;
#define N 5

bool pilihPria(int prioritas[2 * N][N], int w, int m, int m1){//mengecek prioritas
    for (int i = 0; i < N; i++)
    {
        if (prioritas[w][i] == m1)
```

```

        return true;

        if (prioritas[w][i] == m)
            return false;
    }
}

void pencocokan(int prioritas[2 * N][N]) {
    int pasanganWanita[N];

    bool priaSingle[N];

    memset(pasanganWanita, -1, sizeof(pasanganWanita));
    memset(priaSingle, false, sizeof(priaSingle));
    int jumlahSingle = N;

    while (jumlahSingle > 0){
        int m; //inisialisasi pasangan pria
        for (m = 0; m < N; m++)
            if (priaSingle[m] == false)
                break;

        for (int i = 0; i < N && priaSingle[m] == false; i++){//pria single memilih pasangan berdasarkan prioritas tertingginya
            int w = prioritas[m][i];

            if (pasanganWanita[w - N] == -1){//jika wanita belum bertunangan
                pasanganWanita[w - N] = m;
                priaSingle[m] = true;
                jumlahSingle--;
            }

            else{
                int m1 = pasanganWanita[w - N];//jika wanita sudah bertunangan

                if (pilihPria(prioritas, w, m, m1) == false){
                    pasanganWanita[w - N] = m;
                    priaSingle[m] = true;
                    priaSingle[m1] = false;
                }
            }
        }
    }

    cout << " HASIL PASANGAN  \n" << endl;
    string pria;
    string wanita;
    for (int i = 0; i < N; i++){
        if (i < N){
            if (pasanganWanita[i] == 0)
                pria = "Victor";
            if (pasanganWanita[i] == 1)
                pria = "Wyatt";
            if (pasanganWanita[i] == 2)
                pria = "Xavier";
            if (pasanganWanita[i] == 3)
                pria = "Yancey";
            if (pasanganWanita[i] == 4)
                pria = "Zeus";

            if (i == 0)
                wanita = "Amy";
            if (i == 1)
                wanita = "Bertha";

```

```

        if (i == 2)
            wanita = "Clare";
        if (i == 3)
            wanita = "Diane";
        if (i == 4)
            wanita = "Erika";
    }
    cout << pria << "\t& " << wanita << endl;
}

int main()
{
    int prioritas[2 * N][N] = {{6, 5, 8, 9, 7},{8, 6, 5, 7, 9}, {6, 9, 7, 8, 5},{5,
8, 7, 6, 9}, {6, 8, 5, 9, 7},
                                {4, 0, 1, 3, 2},{2, 1, 3, 0, 4}, {1, 2, 3, 4, 0}, {0, 4,
3, 2, 1}, {3, 1, 4, 2, 0}
                                };

    pencocokan(prioritas);

    return 0;
}

```

#### Output dari program:

```

D:\SEMESTER-4\ANALGO\PRAKTIKUM\TUGAS\tugas1_praktikum(26-02-2020)\stable_match_problem.exe
HASIL PASANGAN
Victor & Amy
Xavier & Bertha
Wyatt & Clare
Zeus & Diane
Yancey & Erika
-----
Process exited after 0.2751 seconds with return value 0
Press any key to continue . . .

```

#### Analisis:

1. Apakah jawaban Anda di Worksheet 01 dan Program sama persis? Jika Tidak? Kenapa?  
Ya, jawabannya sama persis.
2. 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!

→ Setiap pria yang ada akan memilih satu wanita yang sangat disukainya berdasarkan table preferensi yang ada, jika terdapat  $n$  pria dan  $n$  wanita maka agar semua mendapatkan pasangan maka terjadi  $n^2$  loop hingga semua berpasangan..

- Teorema (1.4): Jika seorang pria bebas di beberapa titik dalam eksekusi algoritma, maka ada seorang wanita yang belum dia ajak bertunangan. Buktikan!

→ dalam pelaksanaan algoritma G-S setiap pria akan melamar wanita yang berbeda, jika semua pasangan sudah bertunangan maka tidak mungkin ada seorang wanita yang belum pernah diajak bertunangan sama sekali.

- Teorema (1.5): Himpunan  $S$  yang dikembalikan saat terminasi adalah perfect matching. Buktikan!

→ Pria pasti hanya akan melamar apabila belum berpasangan atau pasangan sebelumnya tidak cocok. Sedangkan wanita akan selalu memilih pria dengan preferensi teratas untuk bertunangan dengannya. Baik pria dan wanita, prinsip mereka adalah melamar sesuai urutan preferensi. Dengan itu Himpunan  $S$  adalah perfect matching dikarenakan teori tersebut.

- Teorema (1.6): Sebuah eksekusi algoritma G-S mengembalikan satu set pasangan  $S$ . Set  $S$  adalah pasangan yang stabil. Buktikan!

→ Tidak ada pria yang bisa ditolak oleh semua wanita. Wanita pun harus mendapatkan satu pria dan ia hanya dapat menolak lamaran ketika ada pria yang lebih tinggi preferensinya dibandingkan pria sebelumnya. Setiap iterasi dari loop

sementara melibatkan tepat satu proposal dan pria tidak akan melamar wanita yang sama dua kali. Dalam kasus terburuk, seorang pria bisa melamar semua wanita hingga ia mendapatkan pasangannya. Dengan menggunakan algoritma G-S, dapat kita simpulkan bahwa algoritma akan berakhir setelah semua mendapat pasangan. Selain itu pasangan yang sudah dipasangkan juga harus menunjukkan kecocokan dengan preferensi masing-masing dan pencocokan algoritma G-S dianggap selalu stabil.