**Analisis Algoritma**

****

**Disusun oleh :**

**Meira Dwiana Anjani**

**140810180015**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS PADJADJARAN**

**2020**

Worksheet 01

Jika Anda belum mengerajakan 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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0th | 1st | 2nd | 3rd | 4th |
| Bertha | Amy | Diane | Erika | Clare |
| Diane | Bertha | Amy | Clare | Erika |
| Bertha | Erika | Clare | Diane | Amy |
| Amy | Diane | Clare | Bertha | Erika |
| Bertha | Diane | Amy | Erika | Clare |

Victor Wyatt Xavier Yancey Zeus

Women’s Preferences Profile

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0th | 1st | 2nd | 3rd | 4th |
| Zeus | Victor | Wyatt | Yancey | Xavier |
| Xavier | Wyatt | Yancey | Victor | Zeus |
| Wyatt | Xavier | Yancey | Zeus | Victor |
| Victor | Zeus | Yancey | Xavier | Wyatt |
| Yancey | Wyatt | Zeus | Xavier | Victor |

Amy

### 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 analisalah (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?

Iya, sama.

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 n2 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 prefensi memiliki panjang n, ada paling banyak proposal n2 ynag dapat terjadi. Jadi jumlah dalam iterasi yang bisa terjadi paling banyak adalah n2

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’.

/\*

Nama : Meira Dwiana Anjani

NPM : 140810180015

Kelas : A

Deskripsi : Stable Matching Problem

\*/

#include <iostream>

#include <string.h>

#include <stdio.h>

using namespace std;

#define N 5

// Fungsi true jika wanita lebih menyukai pria 'm1' daripada 'm'

bool pilihPria(int pilih[2\*N][N], int w, int m, int m1)

{

//Cek wanita lebih suka m daripada m1

for (int i = 0; i < N; i++)

{

if (pilih[w][i] == m1)

return true;

if (pilih[w][i] == m)

return false;

}

}

void stableMarriage(int pilih[2\*N][N])

{

int wPartner[N];

bool mFree[N];

memset(wPartner, -1, sizeof(wPartner));

memset(mFree, false, sizeof(mFree));

int freeCount = N;

// Jika pria ada yang Free

while (freeCount > 0)

{

int m;

for (m = 0; m < N; m++)

if (mFree[m] == false)

break;

for (int i = 0; i < N && mFree[m] == false; i++)

{

int w = pilih[m][i];

if (wPartner[w-N] == -1)

{

wPartner[w-N] = m;

mFree[m] = true;

freeCount--;

}

else // Jika wanita tidak free

{

int m1 = wPartner[w-N];

if (pilihPria(pilih, w, m, m1) == false)

{

wPartner[w-N] = m;

mFree[m] = true;

mFree[m1] = false;

}

}

}

}

// Memunculkan Hasilnya

cout << "Hasil Stable Matching : " << endl;

cout << "--------------------------------"<< endl;

cout << "Man \t\tWoman" << endl;

cout<<endl;

string man;

string woman;

for (int i = 0; i < N; i++) {

if(i<N){

if(wPartner[i]==0)

man="Victor";

if(wPartner[i]==1)

man="Wyatt ";

if(wPartner[i]==2)

man="Xavier";

if(wPartner[i]==3)

man="Yancey";

if(wPartner[i]==4)

man="Zeus ";

if(i==0)

woman="Amy";

if(i==1)

woman="Bertha";

if(i==2)

woman="Clare";

if(i==3)

woman="Diane";

if(i==4)

woman="Erika";

}

cout << man << "\tX\t" << woman << endl;

}

}

// Uji Main Program

int main()

{

int pilih[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}

};

stableMarriage(pilih);

return 0;

}

