**LAPORAN PRAKTIKUM 1**

**ANALISIS ALGORITMA**

****

**DISUSUN OLEH**

**ILHAM MUHARAM**

**140810170046**

**KELAS B**

**PROGRAM STUDI S-1 TEKNIK INFORMATIKA**

**UNIVERSITAS PADJADJARAN**

**JATINANGOR**

**2019**

**Worksheet 01**

Mencari himpunan stable-matching dengan algoritma Gale-Shapley.

Men’s Preferences Profile

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0th | 1st | 2nd | 3rd | 4th |
| 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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0th | 1st | 2nd | 3rd | $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 |

Jawab:

Iterasi 1

* **1. (Victor, Bertha)**
* **2. (Wyatt, Diane)**
* **3. (Xavier, Bertha)** 🡪 (Bertha, Xavier > Victor)

Sehingga ~~(Victor, Bertha)~~

* **4. (Yancey, Amy)**
* **5. (Zeus, Bertha)** 🡪 (Bertha, Xavier > Zeus)

Sehingga ~~(Zeus, Bertha)~~

**(Zeus, Diane)** 🡪 (Diane, Zeus > Wyatt)

Sehingga ~~(Wyatt, Diane)~~

Iterasi 2

* **1. (Victor, Amy)** 🡪 (Amy, Victor > Yancey)

Sehingga ~~(Yancey, Amy)~~

* **2. (Wyatt, Bertha)** 🡪 (Bertha, Xavier > Bertha)

Sehingga ~~(Wyatt, Bertha)~~

**(Wyatt, Amy)** 🡪 (Amy, Victor > Wyatt)

Sehingga ~~(Wyatt, Amy)~~

**(Wyatt, Clare)**

* **3. (Yancey, Diane)** 🡪 (Diane, Zeus > Yancey)

Sehingga ~~(Yancey, Diane)~~

**(Yancey, Clare)** 🡪 (Clare, Wyatt > Yancey)

Sehingga ~~(Yancey, Clare)~~

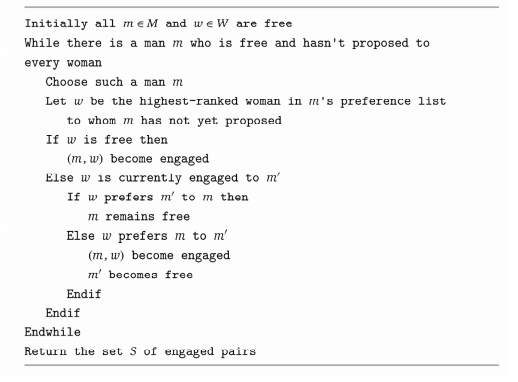
**(Yancey, Bertha)** 🡪 (Bertha, Xavier > Yancey)

Sehingga ~~(Yancey, Bertha)~~

**(Yancey, Erika)**

Hasil Akhir: (Victor, Amy), (Wyatt, Clare), (Xavier, Bertha), (Yancey, Erika), (Zeus, Diane).

**Source Code**

Base:

C++ Code:

|  |
| --- |
| #include <iostream> |
|  | using namespace std; |
|  |  |
|  | typedef int prefer[5]; |
|  |  |
|  | struct orang{ |
|  | string nama; |
|  | int pasangan; |
|  | int preferences[5]; |
|  | }; |
|  |  |
|  | orang setDataOrang(string nama, prefer prefers); |
|  | void stableMatching(orang pria[5], orang wanita[5]); |
|  |  |
|  | int main(){ |
|  | orang pria[5], wanita[5]; |
|  | prefer prefers; |
|  |  |
|  | //Pria |
|  | prefers[0] = 1; prefers[1] = 0; prefers[2] = 3; prefers[3] = 4; prefers[4] = 2; |
|  | pria[0] = setDataOrang("Victor",prefers); |
|  | prefers[0] = 3; prefers[1] = 1; prefers[2] = 0; prefers[3] = 2; prefers[4] = 4; |
|  | pria[1] = setDataOrang("Wyatt",prefers); |
|  | prefers[0] = 1; prefers[1] = 4; prefers[2] = 2; prefers[3] = 3; prefers[4] = 0; |
|  | pria[2] = setDataOrang("Xavier",prefers); |
|  | prefers[0] = 0; prefers[1] = 3; prefers[2] = 2; prefers[3] = 1; prefers[4] = 4; |
|  | pria[3] = setDataOrang("Yancey",prefers); |
|  | prefers[0] = 1; prefers[1] = 3; prefers[2] = 0; prefers[3] = 4; prefers[4] = 2; |
|  | pria[4] = setDataOrang("Zeus",prefers); |
|  |  |
|  | //Wanita |
|  | prefers[0] = 4; prefers[1] = 0; prefers[2] = 1; prefers[3] = 3; prefers[4] = 2; |
|  | wanita[0] = setDataOrang("Amy",prefers); |
|  | prefers[0] = 2; prefers[1] = 1; prefers[2] = 3; prefers[3] = 0; prefers[4] = 4; |
|  | wanita[1] = setDataOrang("Bertha",prefers); |
|  | prefers[0] = 1; prefers[1] = 2; prefers[2] = 3; prefers[3] = 4; prefers[4] = 0; |
|  | wanita[2] = setDataOrang("Clare",prefers); |
|  | prefers[0] = 0; prefers[1] = 4; prefers[2] = 3; prefers[3] = 2; prefers[4] = 1; |
|  | wanita[3] = setDataOrang("Diane",prefers); |
|  | prefers[0] = 3; prefers[1] = 1; prefers[2] = 4; prefers[3] = 2; prefers[4] = 0; |
|  | wanita[4] = setDataOrang("Erika",prefers); |
|  |  |
|  | stableMatching(pria,wanita); |
|  |  |
|  | cout << "\nHasil Stable Matching :" << endl; |
|  | cout << "-----------------" << endl |
|  | << "| Man \t Woman\t|" << endl |
|  | << "-----------------" << endl; |
|  | for(int i = 0; i < 5; i++){ |
|  | cout << "|" << pria[i].nama << "\t " << wanita[pria[i].pasangan].nama << "\t|" << endl; |
|  | } |
|  | cout << "-----------------" << endl; |
|  | } |
|  |  |
|  | orang setDataOrang(string nama, prefer prefers){ |
|  | orang org; |
|  | org.nama = nama; |
|  | org.pasangan = -1; |
|  | for(int i = 0; i < 5;i++){ |
|  | org.preferences[i] = prefers[i]; |
|  | } |
|  | return org; |
|  | } |
|  |  |
|  | void stableMatching(orang pria[5], orang wanita[5]){ |
|  | bool bebas = true; |
|  | int i=0, j=0, prefSek,prefSuk; |
|  | while(bebas){ |
|  | if(wanita[pria[i].preferences[j]].pasangan = -1){ |
|  | pria[i].pasangan = pria[i].preferences[j]; |
|  | wanita[pria[i].preferences[j]].pasangan = i; |
|  | }else{ |
|  | int k = 0; |
|  | prefSek = -1; prefSuk = -1; |
|  | while(prefSek == -1 || prefSuk == -1){ |
|  | if(i == wanita[pria[i].preferences[j]].preferences[k]){ |
|  | prefSuk = k; |
|  | } |
|  | if(wanita[pria[i].preferences[j]].pasangan == wanita[pria[i].preferences[j]].preferences[k]){ |
|  | prefSek = k; |
|  | } |
|  | k++; |
|  | } |
|  | if(prefSuk<prefSek){ |
|  | pria[wanita[pria[i].preferences[j]].pasangan].pasangan = -1; |
|  | pria[i].pasangan = pria[i].preferences[j]; |
|  | wanita[pria[i].preferences[j]].pasangan = i; |
|  | }else{ |
|  | j++; |
|  | continue; |
|  | } |
|  | } |
|  | bebas = false; |
|  | j=0; |
|  | for(int l = 0; l < 5; l++){ |
|  | if(pria[l].pasangan == -1){ |
|  | i = l; |
|  | bebas = true; |
|  | break; |
|  | } |
|  | } |
|  | } |
|  | } |