**LAPORAN PRAKTIKUM**

**ANALISIS ALGORITMA**



**DISUSUN OLEH**

Dimas Ramadhan Pradipta Bagaskara

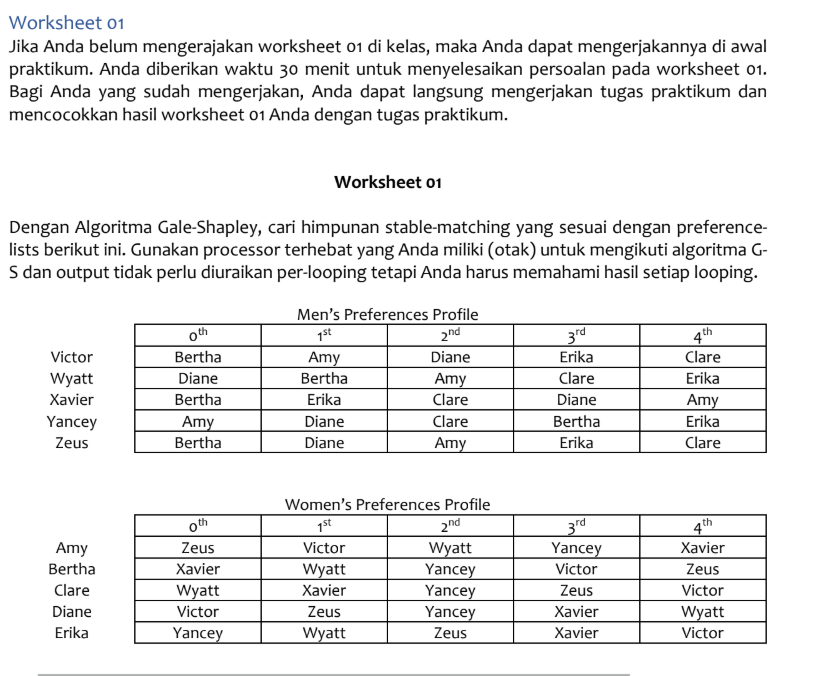
140810180072

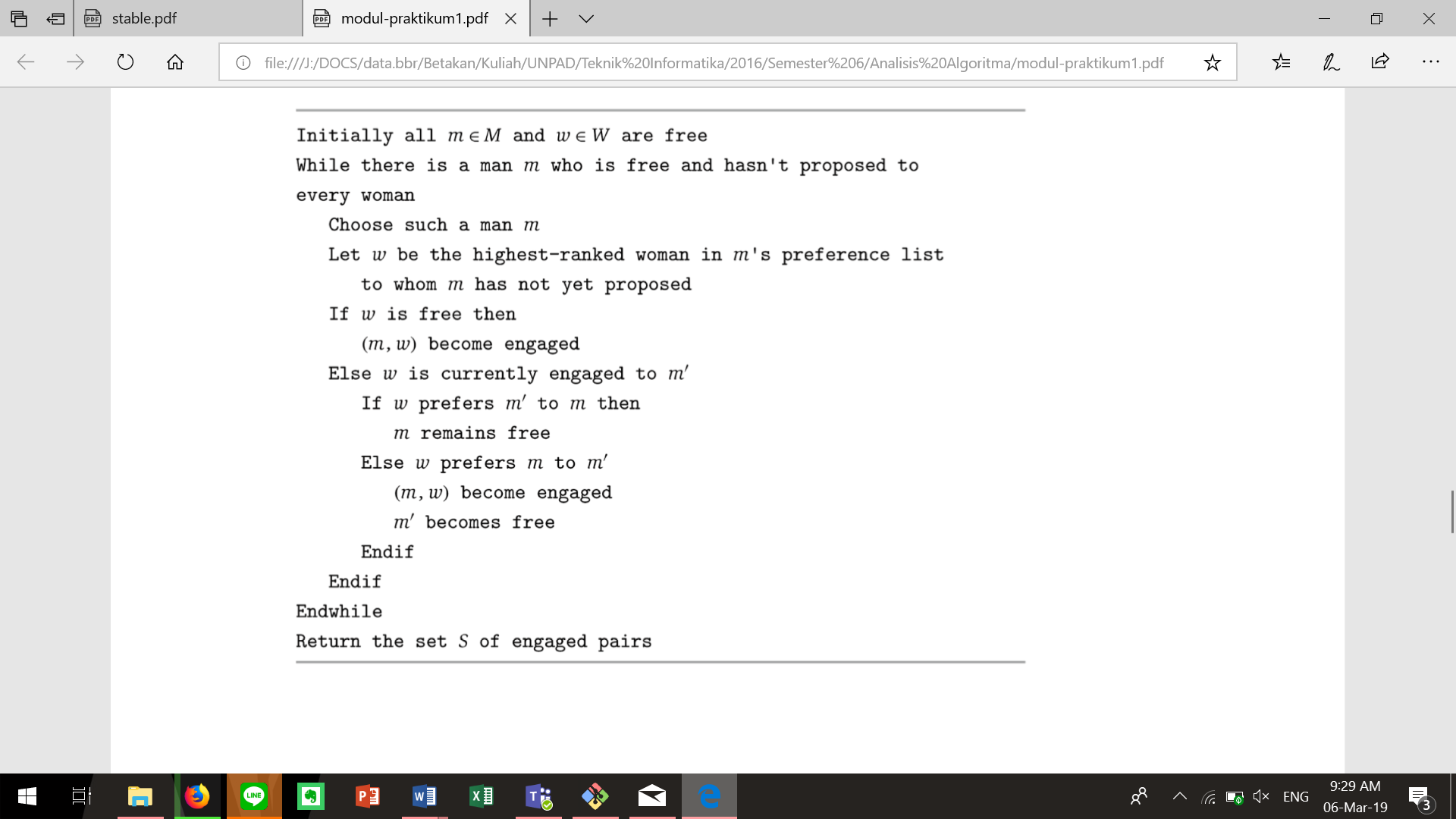
**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS PADJADJARAN**

**2020**





Iterasi 1

Victor - Bertha

Wyatt - Diane

Xavier – Bertha ||Victor Free

Yancey – Amy

Zeus - Diane ||Wyatt Free

Itearasi 2

Victor – Amy

Wyatt – Clare

Yancey – Erika

Hasil akhir :

Victor – Amy

Wyatt – Clare

Xavier - Bertha

Yancey - Erika

Zeus - Diane

{(V,A), (W,C), (X,B) , (Y,E) , (Z,D)}

Pertanyaan:

* 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:
  + Apakah jawaban Anda di Worksheet 01 dan Program sama persis? Jika Tidak? Kenapa?
  + 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!
    - 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!

1. /\*
2. Nama : Dimas Ramadhan Pradipta Bagaskara
3. NPM  : 14081080072
4. Kelas : B\*/
6. #include <iostream>
7. **using** **namespace** std;
9. #define Array 5
10. string male\_name[Array] = {"Victor","Wyatt","Xavier","Yancey","Zeus"};
11. string female\_name[Array] = {"Amy","Bertha","Clare","Diane","Erika"};
13. **int** male\_preferences[Array][Array] = { {1, 0, 3, 4, 2},{3, 1, 0, 2, 4},{1, 4, 2, 3, 0}, {0, 3, 2, 1, 4}, {1, 3, 0, 4, 2}};//berdasarkan urutan array pada string female\_name
14. **int** female\_preferences[Array][Array] = { {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}}; //berdasarkan urutan array pada string male\_name
15. **int** stableMatch[Array][2];
16. **bool** male\_match[Array];
17. **bool** female\_match[Array];
19. **int** getmaleMatch(**int** female){
20. **for** (**int** i=0; i<Array; i++){
21. **if** (stableMatch[i][1]==female){
22. **return** stableMatch[i][0];
23. **break**;
24. }
25. }
26. **return** -1;
27. }
29. **int** getPreferences(**int** female, **int** male){
30. **for** (**int** i=0; i<Array; i++){
31. **if** (female\_preferences[female][i]==male){
32. **return** i;
33. **break**;
34. }
35. }
36. **return** -1;
37. }
39. **void** matching(**int** m, **int** w){
40. stableMatch[m][0] = m;
41. stableMatch[m][1] = w;
42. male\_match[m] = **true**;
43. female\_match[w] = **true**;
44. }
46. **void** hasilMatching(){
47. cout <<"Hasil Matching\n";
48. **for** (**int** i=0;i<Array;i++){
49. cout <<"("<<male\_name[stableMatch[i][0]]<<" dengan "<<female\_name[stableMatch[i][1]]<< ")\n";
50. }
51. }
53. **int** main(){
54. **for** (**int** i=0;i<Array;i++){
55. **for** (**int** j=0;j<Array;j++){
56. **int** m = j;
57. **if** (male\_match[m]==**false**) {
58. **int** w = male\_preferences[m][i];
59. **if** (female\_match[w]==**true**){
60. **int** freeM = getmaleMatch(w);
61. **int** freeM\_pref = getPreferences(w,freeM);
62. **int** M\_pref = getPreferences(w,m);
63. **if** (M\_pref<freeM\_pref){
64. matching(m,w);
65. male\_match[freeM] = **false**;
66. stableMatch[freeM][1] = -1;
67. **continue**;
68. } **else** {
69. **continue**;
70. }
71. } **else** {
72. matching(m,w);
73. **continue**;
74. }
75. } **else** {
76. **continue**;
77. }
78. }
79. }
80. hasilMatching();
81. }
83. Apakah jawaban Anda di Worksheet 01 dan Program sama persis? Jika Tidak? Kenapa?

Jawabannya sama dengan table yang ada di worksheet 01

1. 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!

Karena pada pool terdapat n pria dan n wanita dan juga masing masing pria memiliki preferensi sebanyak n, maka jumlah iterasi maksimal yang dapat terjadi adalah n2 karena setiap pria dapat melamar wanita sebanyak n kali sampai terjadi kecocokan dan sampai semua mendapat pasangan.

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

Ketika seorang pria sudah mendapatkan pasangan maka dia tidak akan melamar wanita lain sampai wanita tersebut dilamar oleh pria lain yang preferensinya lebih tinggi dari dia. Jika hal ini terjadi maka pria tersebut akan bebas dia akan mulai melamar wanita wanita yang ada di preferensinya, pada titik ini pasti ada wanita yang belum dia lamar dalam pool.

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

Algoritma ini akan terus berjalan ketika masih ada pria yang masih single dan belum melamar semua wanita sesuai preferensinya, karena jumlah pria dan wanita sama maka dapat dipastikan pasti setiap pria dan wanita memiliki pasangan masing masing, oleh karena itu Himpunan S adalah perfect matching.

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

Karena pada setiap iterasi akan di prioritaskan preferensi dari pihak pria, dan wanita tidak dapat menolak lamaran dari pria tersebut maka akan dihasilakn satu set pasangan matching, dan juga meskipun wanita yang di lamar oleh pria lain preferensinya lebih tinggi daripada pasangan yang sekarang maka sang wanita dapat berpisah lalu Bersama pria yang baru melamar tersebut namun jika pria tidak bisa mendapatkan wanita tersebut dia akan melamar wanita lainnya, sehingga pasti setiap eksekusi akan menghasilkan satu buah pasangan.