

4. Tessa Nurwani - 2100214

s. Wendi Kardin - 2100016

6. Yesha Rahma - 2101138

## Matrices Adjacency

Rama	Ella
Rama	Bob
Rama	Katie
Ella	Bob
Bob	Tom
Katie	Lee
Katie	Swati
Lee	Bob
Lee	Sam
Lee	Zahir
Lee	Swati
Swati	Zahir
Zahir	Sam
Zahir	Arun
Sam	Tom
Sam	Arun
Bob	Sam

	0	1	2	3	4	5	6	7	8	9	10
1	0	1	1	0	0	0	1	0	0	0	
2	1	0	0	1	1	0	0	0	0	0	
3	1	0	0	0	0	0	1	0	0	0	
4	0	1	0	0	1	1	1	1	0	0	
5	0	1	0	1	0	1	0	0	0	0	
6	0	0	0	1	1	0	0	1	0	1	
7	1	0	1	1	0	0	0	1	1	0	
8	0	0	0	0	0	1	1	0	1	1	
9	0	0	0	0	0	0	1	1	0	0	
10	0	0	0	0	0	1	0	1	0	0	

## Matrices Adjacency

[illegible][illegible]

## **Tugas Struktur Data – Graph Representation Menggunakan Matriks**

Kelompok : 1

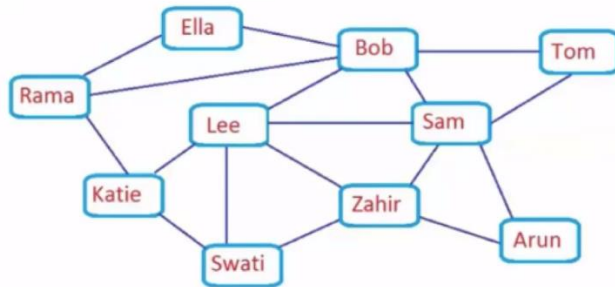
Anggota Kelompok :

1. Fajar Mustaqim (2100302)
2. Galih Lazuardi (2103670)
3. Sajida I'tikafah (2102116)
4. Tessa Nurwani (2100214)
5. Wendi Kardian (2100016)
6. Yesha Rahma (2101138)

Kelas : Pendidikan Ilmu Komputer A

## 1. Data Graph yang akan direpresentasikan kedalam bentuk matriks

### IMPLEMENTASI GRAPH



### Source Code :

```
emuan 10 > C Matriks1.c > main()
#include<stdio.h>
#include<stdlib.h>
#define vertex 17

int main(){
    // matriks ruas
    char matriksRuas[vertex][2][10] = {"Rama", "Ella"},
    {"Rama", "Bob"},
    {"Rama", "Katie"},
    {"Ella", "Bob"},
    {"Bob", "Tom"},
    {"Bob", "Sam"},
    {"Katie", "Lee"},
    {"Katie", "Swati"},
    {"Lee", "Bob"},
    {"Lee", "Sam"},
    {"Ella", "Zahir"},
    {"Lee", "Swati"},
    {"Swati", "Zahir"},
    {"Zahir", "Sam"},
    {"Zahir", "Arun"},
    {"Sam", "Tom"},
    {"Sam", "Arun"}
};

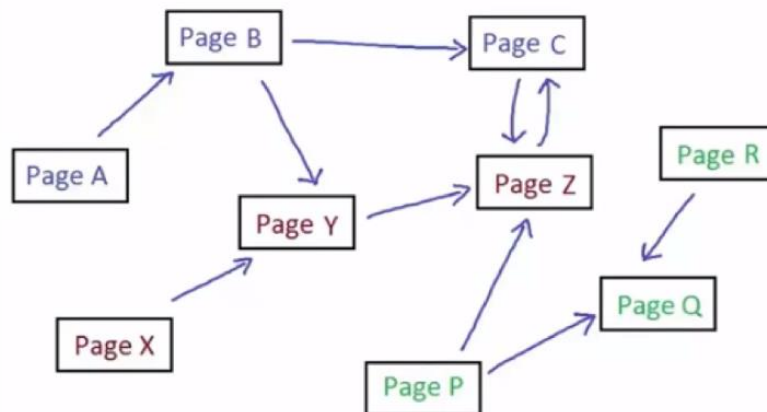
    int i, j ;
    for (i = 0; i<vertex; i++){
        for(j = 0; j<2; j++){
            printf("%s\t", matriksRuas[i][j]);
        }
        printf("\n");
    }
}
```

Output :

```
PS D:\UPI\SEMESTER 2\STRUKTUR DATA> cd "d:\UPI\SEMESTER 2\STRUKTUR DATA\Pertemuan 10\" ; if ($?) { gcc Matriks1.c -o Matriks1 } ; if ($?) { .\Matriks1 }
Rama Ella
Rama Bob
Rama Katie
Ella Bob
Bob Tom
Bob Sam
Katie Lee
Katie Swati
Lee Bob
Lee Sam
Ella Zahir
Lee Swati
Swati Zahir
Zahir Sam
Zahir Arun
Sam Tom
Sam Arun
PS D:\UPI\SEMESTER 2\STRUKTUR DATA\Pertemuan 10>
```

2. Data Graph yang akan direpresentasikan kedalam bentuk matriks

## IMPLEMENTASI GRAPH



## Source Code :

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #define maxVertex 20
4  int adj[maxVertex][maxVertex];
5  int n;
6
7  void createGraph(){
8      int i, max_edges, origin, destin;
9      printf("Enter number of nodes : ");
10     scanf("%d", &n);
11     max_edges = n * (n-1);
12     for(i = 1; i <= max_edges; i++){
13         printf("masukan titik (0,0) untuk keluar : ");
14         scanf("%d %d", &origin, &destin);
15         if(origin == 0 && destin == 0){
16             break;
17         }
18         if(origin > n || destin > n || origin <= 0 || destin <= 0){
19             printf("\nInput Salah");
20         }else{
21             adj[origin][destin] = 1;
22         }
23     }
24 }
25
26 void insert_graph(){
27     int i;
28     n++;
29     printf("\nNode yang dimasukkan adalah : %d", n);
30     for(i = 1; i <= n; i++){
31         adj[i][n] = 0;
32         adj[n][i] = 0;
33     }
34 }
35
36
```

```
void displayGraph(){
    int i, j;
    for(i = 1; i <= n; i++){
        for(j = 1; j <= n; j++){
            printf("%d\t", adj[i][j]);
        }
        printf("\n");
    }
}

void deleteGraph(char u){
    int i, j;
    if(n == 0){
        printf("\nGraph kosong");
    }
    else if(u > n){
        printf("Node tidak ada di dalam graph");
    }else{
        for(i = u; i <= n; i++){
            for(j = 1; j <= n; j++){
                adj[j][i] = adj[j][i+1];
                adj[i][j] = adj[i+1][j];
            }
        }
    }
}
```

```

68     int main(){
69         int option;
70
71         // Source Control (Ctrl+Shift+G) - 12 pending changes
72         do{
73             printf("\n 1. Insert data");
74             printf("\n 2. Delete data");
75             printf("\n 3. Display Graph");
76             printf("\n 4. Delete Graph");
77             printf("\n\n Masukkan pilihan anda");
78             scanf("%d", &option);
79             switch(option){
80                 case 1:
81                 insert_graph();
82                 break;
83                 case 2:
84                 printf("Node yang mau di hapus : ");
85                 scanf("%d", &node);
86                 deleteGraph(node);
87                 break;
88                 case 3:
89                 displayGraph();
90                 break;
91                 case 4:
92                 printf("\n\n Bye");
93                 break;
94                 default :
95                 printf("\nInput salah\n");
96                 break;
97             }
98
99         }while(option != 4);
100     }

```

Output :

a. Test case pertama untuk soal no 1 menggunakan representasi matriks adjacency

```

PS D:\UPI\SEMESTER 2\STRUKTUR DATA\Pertemuan 10> cd "d:\UPI\SEMESTER 2\STRUKTUR DATA\Pertemuan 10\"; if ($?) { gcc MatriksIAgency.c -o MatriksIAgency }; if ($?) { .\MatriksIAge
ncy }
Enter number of nodes : 10
masukan titik (0,0) untuk keluar : 1 2
masukan titik (0,0) untuk keluar : 1 3
masukan titik (0,0) untuk keluar : 3 1
masukan titik (0,0) untuk keluar : 3 7
masukan titik (0,0) untuk keluar : 4 2
masukan titik (0,0) untuk keluar : 4 5
masukan titik (0,0) untuk keluar : 4 6
masukan titik (0,0) untuk keluar : 4 7
masukan titik (0,0) untuk keluar : 4 8
masukan titik (0,0) untuk keluar : 5 2
masukan titik (0,0) untuk keluar : 5 4
masukan titik (0,0) untuk keluar : 5 6
masukan titik (0,0) untuk keluar : 6 4
masukan titik (0,0) untuk keluar : 6 5
masukan titik (0,0) untuk keluar : 6 8
masukan titik (0,0) untuk keluar : 7 1
masukan titik (0,0) untuk keluar : 7 3
masukan titik (0,0) untuk keluar : 7 4
masukan titik (0,0) untuk keluar : 7 8
masukan titik (0,0) untuk keluar : 7 9
masukan titik (0,0) untuk keluar : 8 6
masukan titik (0,0) untuk keluar : 8 7
masukan titik (0,0) untuk keluar : 8 9
masukan titik (0,0) untuk keluar : 8 10
masukan titik (0,0) untuk keluar : 9 7
masukan titik (0,0) untuk keluar : 9 8
masukan titik (0,0) untuk keluar : 10 6
masukan titik (0,0) untuk keluar : 10 8
masukan titik (0,0) untuk keluar : 0 0

```

```

1. Insert data
2. Delete data
3. Display Graph
4. Delete Graph

Masukan pilihan anda3
0 1 1 0 0 0 1 0 0 0
1 0 0 1 1 0 0 0 0 0
1 0 0 0 0 0 1 0 0 0
0 1 0 0 1 1 1 1 0 0
0 1 0 1 0 1 0 0 0 0
0 0 0 1 1 0 0 1 0 0
1 0 1 1 0 0 0 1 1 0
0 0 0 0 0 1 1 0 1 1
0 0 0 0 0 0 1 1 0 0
0 0 0 0 0 1 0 1 0 0

```

b. Test case kedua untuk input berdasarkan soal nomor 2 menggunakan representasi matriks adjacency

```

PS D:\UPI\SEMESTER 2\STRUKTUR DATA> cd "d:\UPI\SEMESTER 2\STRUKTUR DATA\Pertemuan 10\"
Enter number of nodes : 9
masukan titik (0,0) untuk keluar : 1 2
masukan titik (0,0) untuk keluar : 2 3
masukan titik (0,0) untuk keluar : 2 5
masukan titik (0,0) untuk keluar : 3 6
masukan titik (0,0) untuk keluar : 4 5
masukan titik (0,0) untuk keluar : 5 6
masukan titik (0,0) untuk keluar : 6 3
masukan titik (0,0) untuk keluar : 7 6
masukan titik (0,0) untuk keluar : 7 8
masukan titik (0,0) untuk keluar : 9 8
masukan titik (0,0) untuk keluar : 0 0

1. Insert data
2. Delete data
3. Display Graph
4. Delete Graph

Masukan pilihan anda3
0 1 0 0 0 0 0 0 0
0 0 1 0 1 0 0 0 0
0 0 0 0 0 1 0 0 0
0 0 0 0 1 0 0 0 0
0 0 0 0 0 1 0 0 0
0 0 1 0 0 0 0 0 0
0 0 0 0 0 1 0 1 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0

```