

ALGORITMA DAN PEMROGRAMAN II

MERGE TABLE (PENGABUNGAN TABEL)

Rosa Ariani Sukamto

ROSA ARIANI SUKAMTO

Blog: <http://hariiniadalahhadiah.wordpress.com>

Facebook: <https://www.facebook.com/rosa.ariani.sukamto>

Email: rosa_if_itb_01@yahoo.com

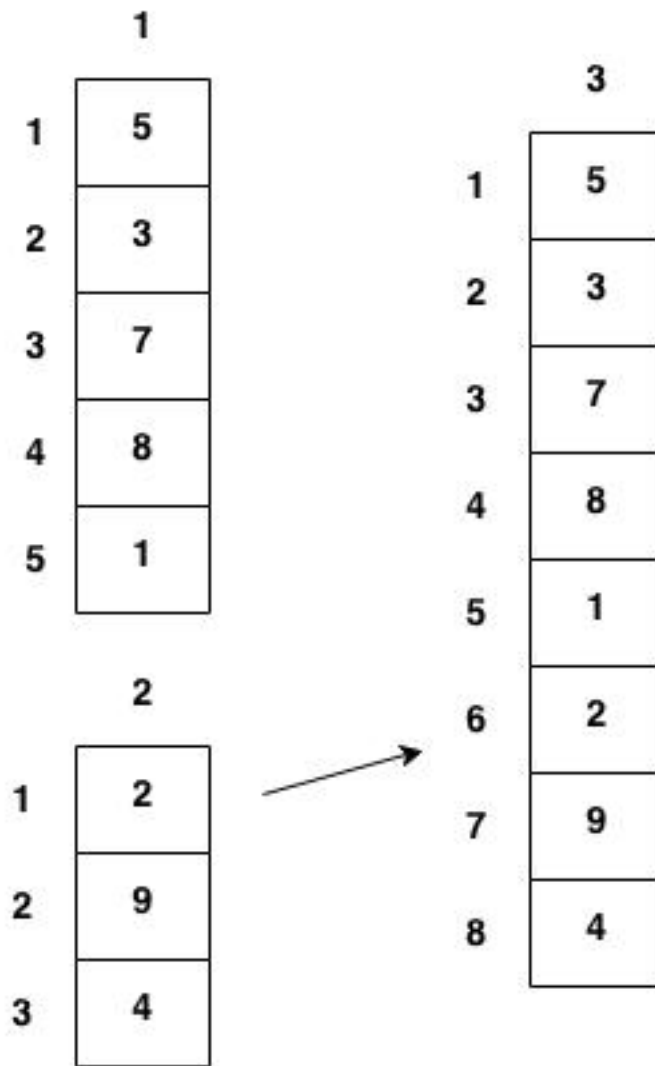


MERGE TABLE

- **Menggabungkan 2 atau lebih tabel (*array*)**
- Tidak terurut
- Terurut



MERGE TABLE - TIDAK TERURUT (1)



MERGE TABLE - TIDAK TERURUT (2)

```
int main() {

    int tabel1[4];
    int tabel2[4];

    int tabel3[8];

    int penghitung;

    for(penghitung=0; penghitung<4;
    penghitung++) {
        printf("masukkan angka:\n");
        scanf("%d",
        &tabel1[penghitung]);
        printf("\n");
    }

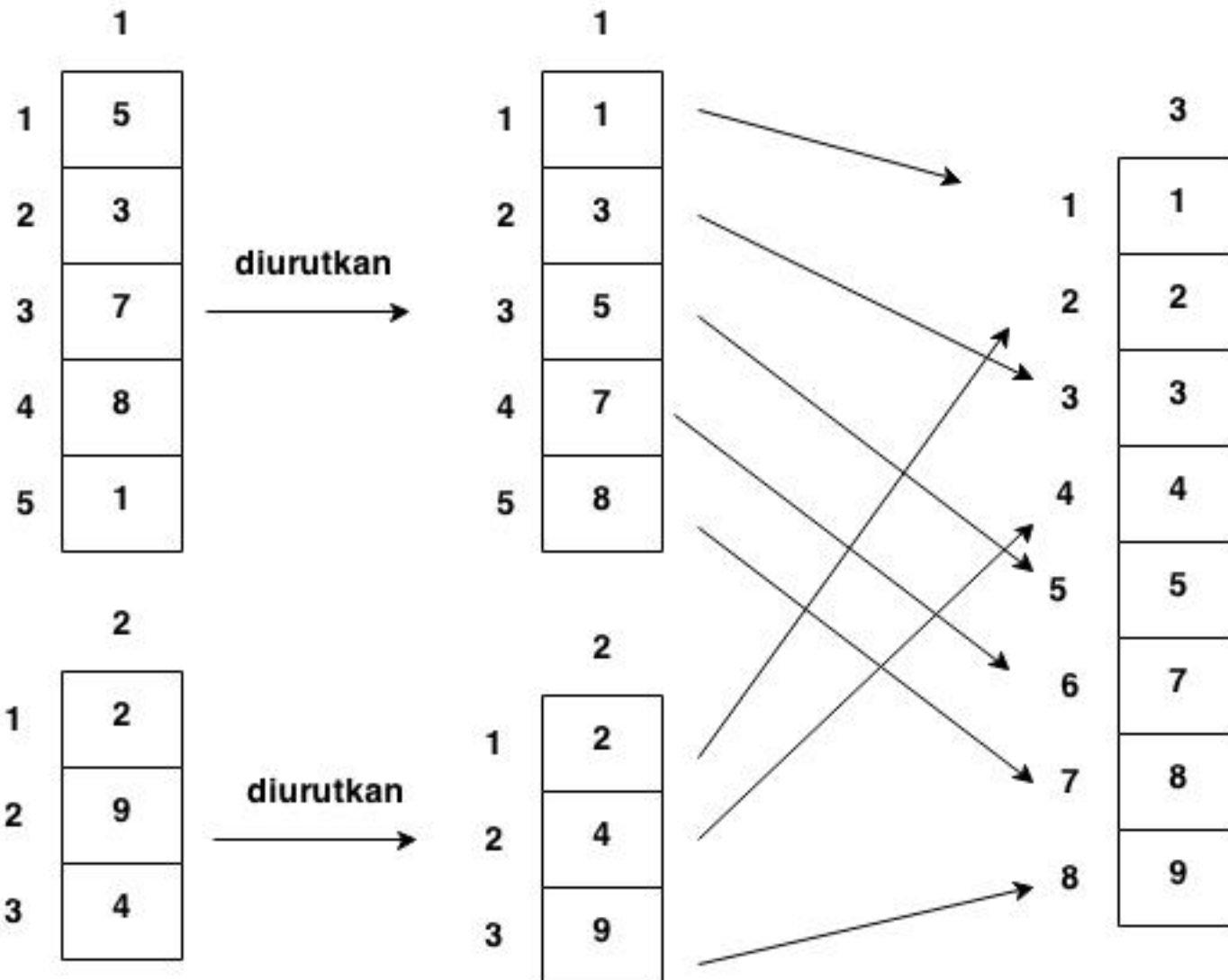
    for(penghitung=0; penghitung<4;
    penghitung++) {
        printf("masukkan angka:\n");
        scanf("%d",
        &tabel2[penghitung]);
        printf("\n");
    }
}
```

```
for(penghitung=0; penghitung<4;
penghitung++) {
    tabel3[penghitung] =
    tabel1[penghitung];
}

for(penghitung=4; penghitung<8;
penghitung++) {
    tabel3[penghitung] =
    tabel2[penghitung-4];
}

return 0;
}
```

MERGE TABLE - TERURUT (1)



MERGE TABLE - TERURUT (2)

Kenapa gak langsung digabung terus di sorting semua?

Masalah efisiensi....



MERGE TABLE - TERURUT (3-2) - ISI TABEL

```
int main(){  
    int tabel1[4];  
    int tabel2[4];  
    int tabel3[8];  
    int penghitung1;  
    int penghitung2;  
    int penghitung3;  
    int penghitung;  
  
    for(penghitung1=0; penghitung1<4; penghitung1++){  
        printf("masukkan angka:\n");  
        scanf("%d", &tabel1[penghitung1]);  
        printf("\n");  
    }  
  
    for(penghitung2=0; penghitung2<4; penghitung2++){  
        printf("masukkan angka:\n");  
        scanf("%d", &tabel2[penghitung2]);  
        printf("\n");  
    }  
}
```


MERGE TABLE - TERURUT (3-2) - PROSES BERGANTIAN

```
/*inisialisasi indeks awal setiap tabel */
penghitung1 = 0;
penghitung2 = 0;
penghitung3 = 0;
while((penghitung1 < 4) && (penghitung2 < 4)){
    if(tabel1[penghitung1] < tabel2[penghitung2]){
        tabel3[penghitung3] = tabel1[penghitung1];
        penghitung1 = penghitung1 + 1;
        penghitung3 = penghitung3 + 1;
    }else if(tabel2[penghitung2] < tabel1[penghitung1]){
        tabel3[penghitung3] = tabel2[penghitung2];
        penghitung2 = penghitung2 + 1;
        penghitung3 = penghitung3 + 1;
    }else{
        tabel3[penghitung3] = tabel1[penghitung1];
        penghitung1 = penghitung1 + 1;
        penghitung3 = penghitung3 + 1;
        tabel3[penghitung3] = tabel2[penghitung2];
        penghitung2 = penghitung2 + 1;
        penghitung3 = penghitung3 + 1;
    }
}
```

MERGE TABLE - TERURUT (3-3) - PROSES SISA

```
if(penghitung1 < 4){  
    for(penghitung=penghitung1; penghitung<4; penghitung++){  
        tabel3[penghitung3] = tabel1[penghitung];  
        penghitung3 = penghitung3 + 1;  
    }  
}
```

```
if(penghitung2 < 4){  
    for(penghitung=penghitung2; penghitung<4; penghitung++){  
        tabel3[penghitung3] = tabel2[penghitung];  
        penghitung3 = penghitung3 + 1;  
    }  
}  
return 0;
```

```
}
```

MERGE TABLE - LEBIH DARI 2 TABEL



DAFTAR PUSTAKA

S, Rosa A. dan M. Shalahuddin. 2010. Modul Pembelajaran: Algoritma dan Pemrograman. Modula: Bandung.

