

Nama : Zulfikar Rahman

NIM : 1941720192

Kelas : TI-1D

4. PERTANYAAN

1. Terdapat di method apakah proses bubble sort?
2. Terdapat di method apakah proses selection sort?
3. Apakah yang dimaksud proses swap? Tuliskan potongan program untuk melakukan proses swap tersebut!
4. Di dalam method bubbleSort(), terdapat baris program seperti di bawah ini:

```
29         if(listMhs[j].ipk > listMhs[j-1].ipk){
30             //di bawah ini proses swap atau penukaran
31             Mahasiswa tmp = listMhs[j];
32             listMhs[j] = listMhs[j-1];
33             listMhs[j-1] = tmp;
34         }
35     }
```

Untuk apakah proses tersebut?

5. Perhatikan perulangan di dalam bubbleSort() di bawah ini:

```
27         for(int i=0; i<listMhs.length-1; i++){
28             for(int j=1; j<listMhs.length-i; j++){
```

- a. Apakah perbedaan antara kegunaan perulangan i dan perulangan j?
- b. Mengapa syarat dari perulangan i adalah $i < \text{listMhs.length} - 1$?
- c. Mengapa syarat dari perulangan j adalah $j < \text{listMhs.length} - i$?
- d. Jika banyak data di dalam listMhs adalah 50, maka berapakah perulangan i akan berlangsung? Dan ada berapa **Tahap** bubble sort yang ditempuh?

6. Di dalam method selection sort, terdapat baris program seperti di bawah ini:

```
42         int idxMin = i;
43         for(int j=i+1; j<listMhs.length; j++){
44             if(listMhs[j].ipk < listMhs[idxMin].ipk){
45                 idxMin = j;
46             }
47         }
```

Untuk apakah proses tersebut?

7. Ubahlah fungsi pada InsertionSort sehingga fungsi ini dapat melaksanakan proses sorting dengan cara ascending atau decending, anda dapat melakukannya dengan menambahkan parameter pada pemanggilan fungsi insertionSort.

```
33     void insertionSort(boolean ascending){
34         int i,j;
35         for(i =1;i<listMhs.length;i++){
36             Mahasiswa temp = new Mahasiswa();
37             temp = listMhs[i];
38             j = i;
39             if(ascending){
40
41
42
43
44             }else{
45
46
47
48
49             }
50             listMhs[j] = temp;
51         }
52     }
53 }
```

Jawab :

1.

```
void bubblesort() {  
    for (int i = 0; i < listmhs.length - 1; i++) {  
        for (int j = 1; j < listmhs.length - i; j++) {  
            if (listmhs[j].ipk > listmhs[j - 1].ipk) {  
                daftarMahasiswa tmp = listmhs[j];  
                listmhs[j] = listmhs[j - 1];  
                listmhs[j - 1] = tmp;  
            }  
        }  
    }  
}
```

Terdapat pada method bubbleSort

2.

```
void selectionSort() {  
    for (int i = 0; i < listmhs.length; i++) {  
        int idxMin = i;  
        for (int j = i + 1; j < listmhs.length; j++) {  
            if (listmhs[j].ipk < listmhs[idxMin].ipk) {  
                idxMin = j;  
            }  
        }  
        daftarMahasiswa tmp = listmhs[idxMin];  
        listmhs[idxMin] = listmhs[i];  
        listmhs[i] = tmp;  
    }  
}
```

Terdapat pada method selectionSort

3. Proses Swap adalah pemindahan sebuah elemen untuk mengurutkan.

```
for (int i = 0; i < listmhs.length; i++) {  
    int idxMin = i;  
    for (int j = i + 1; j < listmhs.length; j++) {  
        if (listmhs[j].ipk < listmhs[idxMin].ipk) {  
            idxMin = j;  
        }  
    }  
    daftarMahasiswa tmp = listmhs[idxMin];  
    listmhs[idxMin] = listmhs[i];  
    listmhs[i] = tmp;  
}
```

4. proses tersebut digunakan untuk melakukan swap atau penukaran .

5.

- a. Perbedaannya adalah jika perulangan i untuk index acuan atau bisa juga disebut outer loop, sedangkan perulangan j untuk inner loop
- b. Sebab banyak elemen dari **listMhs** adalah 5, pada perulangan i , i dimulai dari 0 sehingga perulangan dimulai dari 0 sampai 4.
- c. Sebab untuk proses pengurutan dimulai dari 0, jika pengurutan dari kiri sudah berurutan maka akan di abaikan
- d. Perulangan i akan dilakukan sebanyak 49 kali dan banyaknya tahap bubbleSort yaitu 1225

6. Proses Tersebut di gunakan untuk melakukan proses pencarian indeks nilai terkecil.

7.

```
void insertionSort(boolean ascending) {
    int i, j;
    for (i = 1; i < listmhs.length; i++) {
        daftarMahasiswa temp = new daftarMahasiswa();
        temp = listmhs[i];
        j = i;
        if (ascending) {
            while ((j > 0) && (listmhs[j - 1].ipk > temp.ipk)) {
                listmhs[j] = listmhs[j - 1];
                j--;
            }
        } else {
            while ((j < 0) && (listmhs[j - 1].ipk > temp.ipk)) {
                listmhs[j] = listmhs[j - 1];
                j--;
            }
        }
        listmhs[j] = temp;
    }
}
```