Jobsheet 5

Praktikum Algoritma & Struktur Data



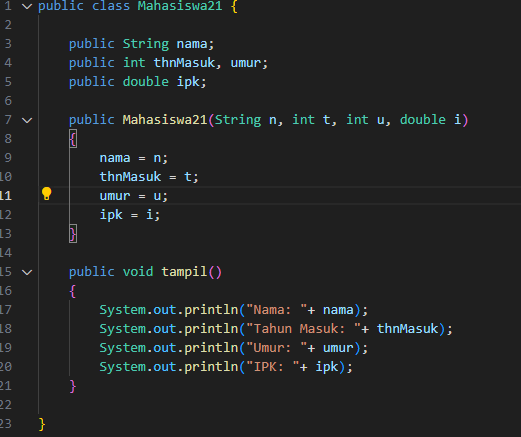
Rafi Ody Prasetyo  
(2341720180)

D-IV Teknik Informatika  
Politeknik Negeri Malang  
Semester 2  
2024

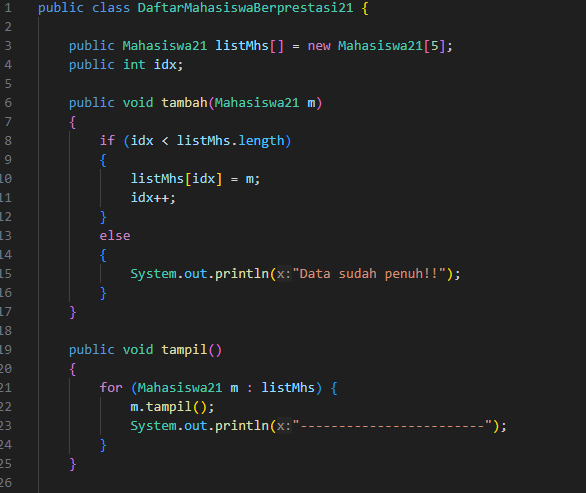
**Percobaan 1**

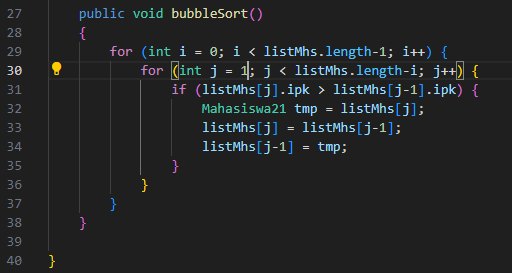
**Code:**

**Mahasiswa21.java**

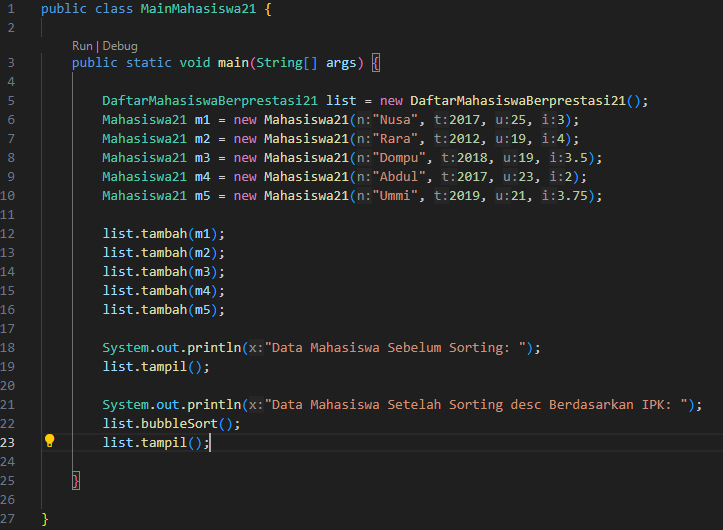
****

**DaftarMahasiswaBeprestasi21.java**

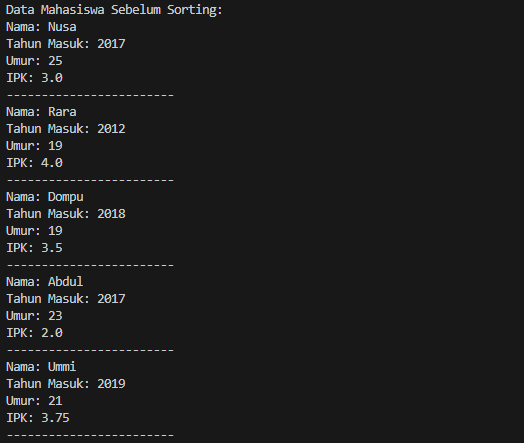
****

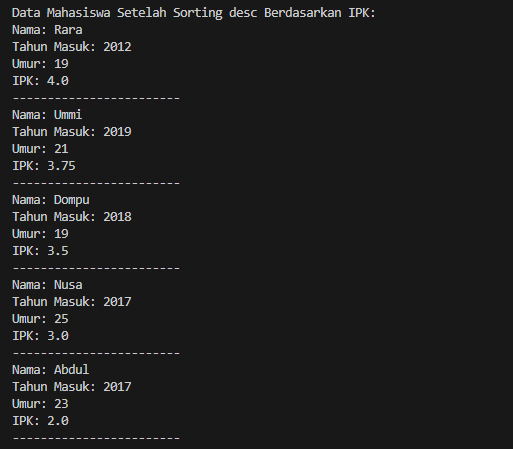
****

**MainMahasiswa21.java**

****

**Output:**

****

****

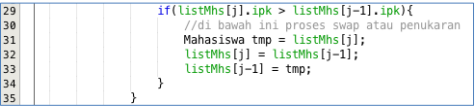
**Pertanyaan**

1. **Terdapat di method apakah proses bubble sort?**

**Jawab:**

Bubble sort terdapat pada method void bubbleSort() yang terdapat pada class DaftarMahasiswaBerprestasi.

1. **Di dalam method bubbleSort(), terdapat baris program seperti di bawah ini:**

****

**Untuk apakah proses tersebut?**

**Jawab:**

Kondisi pada statement if digunakan untuk memeriksa apakah IPK dari mahasiswa pada posisi j lebih besar dari IPK mahasiswa pada posisi j-1. Jika iya, maka artinya elemen pada posisi j perlu ditukar dengan elemen pada posisi j-1, sehingga elemen dengan IPK lebih tinggi akan bergerak ke arah kanan array. Kemudian untuk variabel tmp digunakan untuk menyimpan elemen yang akan dipindahkan (elemen pada posisi j) ke dalam variabel sementara tmp. Selanjutnya listMhs[j] = listMhs[j-1]; digunakan untuk menyalin elemen pada posisi j-1 ke posisi j, karena elemen pada posisi j akan digantikan dengan elemen yang lebih kecil. Dan yang terakhir listMhs[j-1] = tmp; digunakan untuk menempatkan kembali elemen yang disimpan dalam variabel sementara (tmp) ke posisi j-1, sehingga elemen pada posisi j-1 sekarang berisi elemen yang lebih besar.

1. **Perhatikan perulangan di dalam bubbleSort() di bawah ini:**

****

1. **Apakah perbedaan antara kegunaan perulangan i dan perulangan j?**

**Jawab:**

Outer loop digunakan untuk melakukan iterasi sejumlah yang diperlukan untuk mengurutkan array. Sedang inner loop digunakan untuk melakukan perbandingan dan penukaran elemen array. Loop akan berjalan dari 1 hingga listMhs.length - 1. Hal ini dilakukan untuk memastikan bahwa iterasi loop dalam tidak akan melebihi batas yang telah diurangi dari panjang array dan iterasi loop luar. Dengan demikian, pada setiap iterasi dari loop luar, satu elemen besar (atau kecil, tergantung pada urutan yang diinginkan) akan diposisikan dengan benar di akhir array.

1. **Mengapa syarat dari perulangan i adalah i<listMhs.length-1 ?**

**Jawab:**

Pada setiap iterasi dari perulangan i, algoritma Bubble Sort memastikan bahwa elemen terbesar (atau terkecil, tergantung pada urutan yang diinginkan) telah berpindah ke posisi yang benar di ujung array. Jika kita terus melanjutkan proses pengurutan setelah elemen terakhir berada di posisi yang benar, itu hanya akan membuang waktu karena tidak akan ada penukaran yang terjadi.

1. **Mengapa syarat dari perulangan i adalah j<listMhs.length-1 ?**

**Jawab:**

Agar iterasi tidak membandingkan elemen yang sudah diurutkan.

1. **Jika banyak data di dalam listMhs adalah 50, maka berapakali perulangan i akanberlangsung? Dan ada berapa Tahap bubble sort yang ditempuh?**

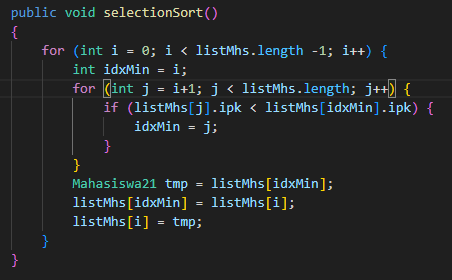
**Jawab:**

Perulangan i akan melakukan iterasi sebanyak 49 kali, dan melakukan bubble sort sebanyak maksimal 1225 tahap.

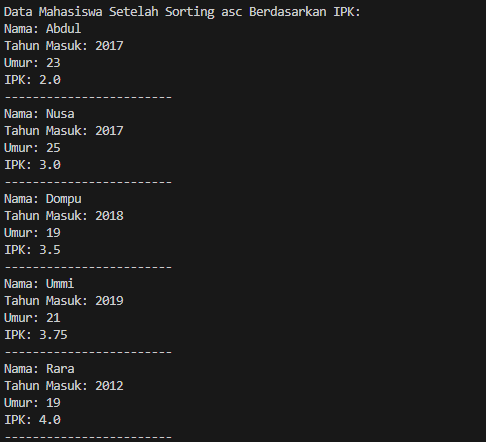
**Percobaan 2**

**Code:**

**DaftarMahasiswaBerprestasi21.java**

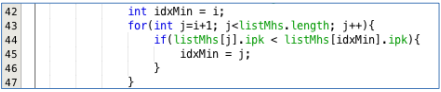
****

**Output:**



**Pertanyaan**

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

****

**Untuk apakah proses tersebut, jelaskan!**

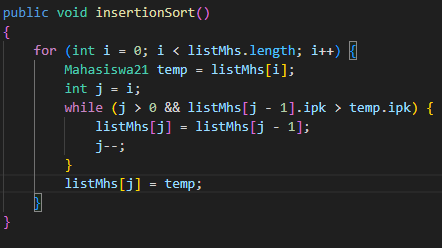
**Jawab:**

Proses tersebut berfungsi untuk mengurutkan nilai dalam array listMhs dari yang terkecil ke terbesar berdasarkan nilai ipk. Variabel idxMin diguanakan untuk nilai awal dengan nilai i. Kemudian untuk variabel j digunakan untuk membandingkan nilai dengan nilai variabel idxMin. Jika nilai j lebih kecil dari nilai variabel idxMin maka nilai idxMin akan diganti dengan j.

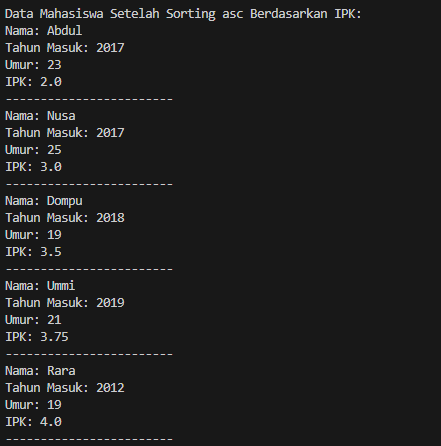
**Percobaan 4**

**Code:**

**DaftarMahasiswaBerprestasi21.java**

****

**Output:**

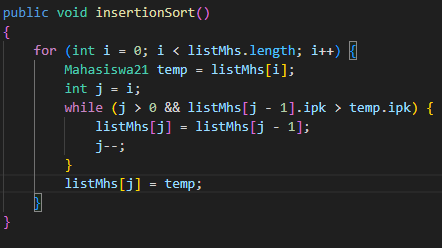
****

**Pertanyaan**

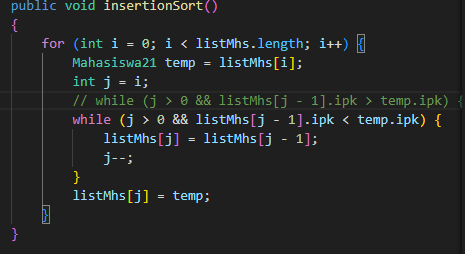
1. **Ubahlah fungsi pada InsertionSort sehingga fungsi ini dapat melaksanakan proses sorting dengan cara descending**

**Jawab:**

Before:

****

After:



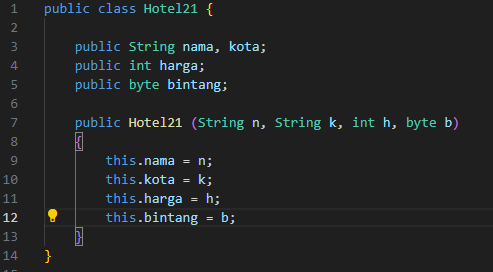
Output:



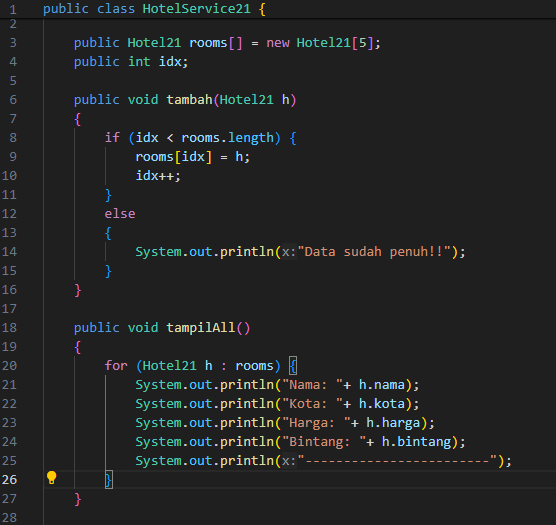
**Praktikum**

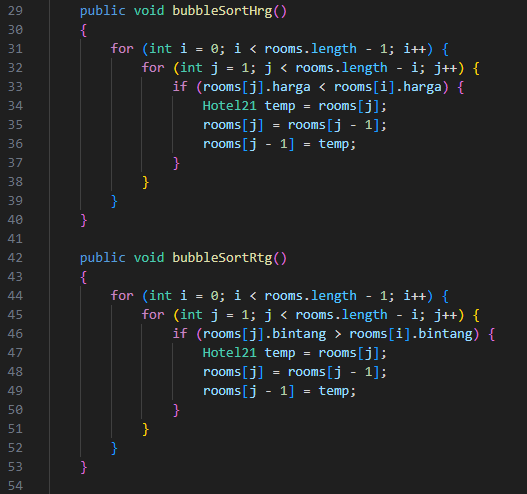
**Code:**

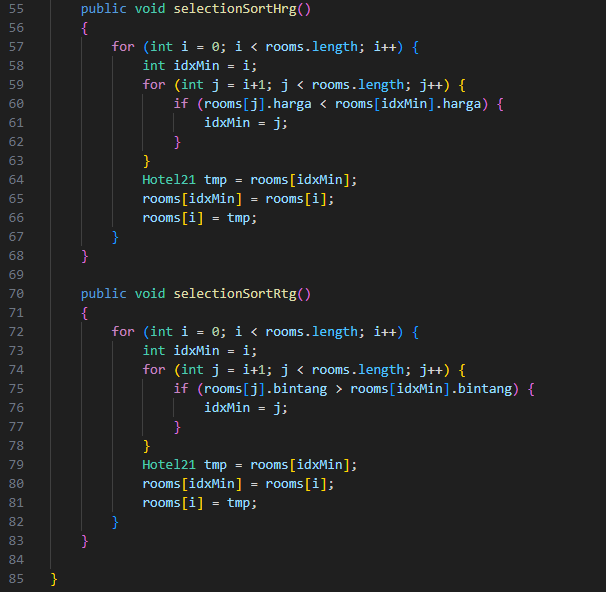
**Hotel21.java**

****

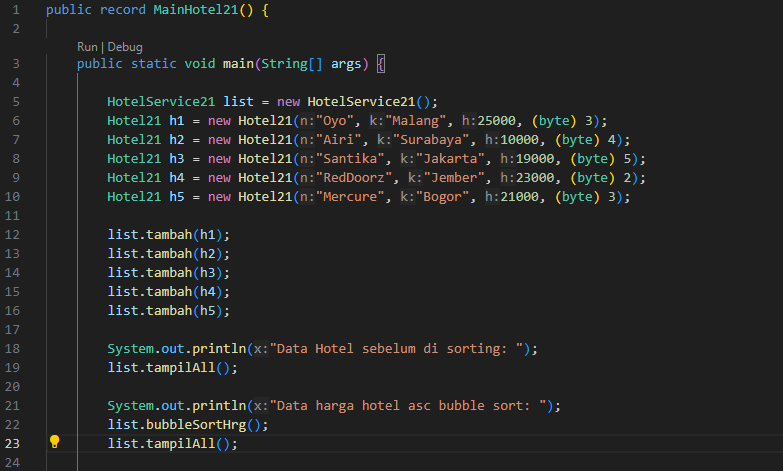
**HotelService21.java**

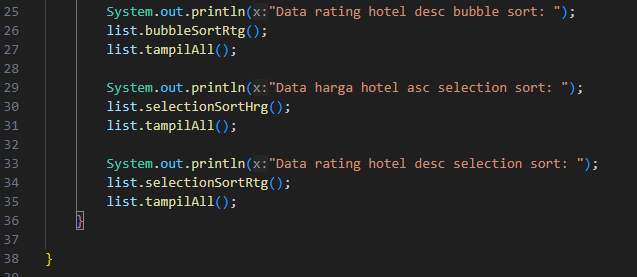
****

****

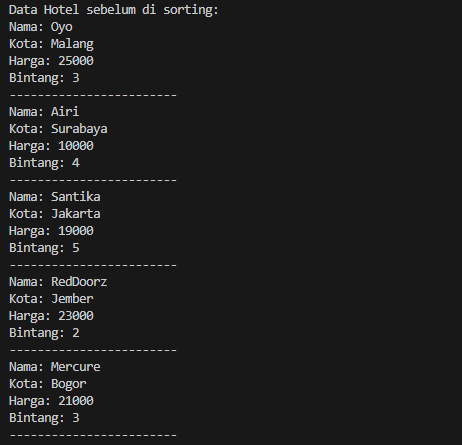
****

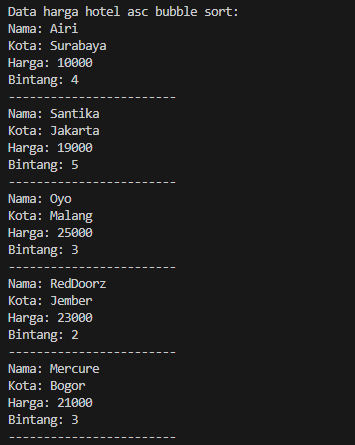
**MainHotel21.java**

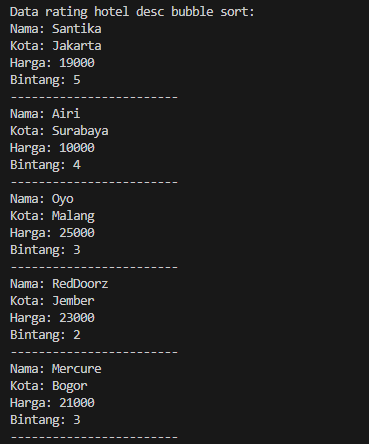
****

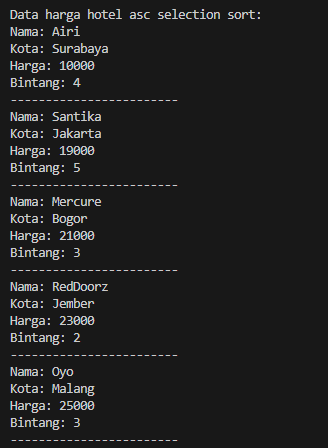
****

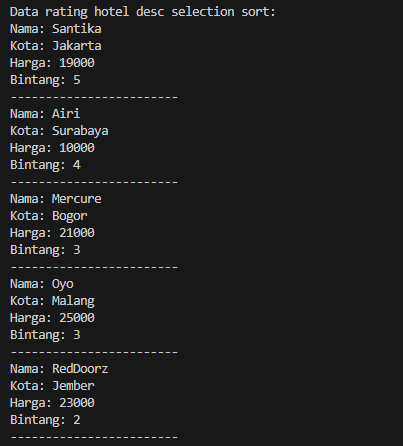
**Output:**

****

****

****

****

****