**LAPORAN**

**PRAKTIKUM ALGORITMA DAN STRUKTUR DATA**

**( MODUL 6 )**

**“ PENGURUTAN LANJUTAN”**

****

**Disusun oleh :**

**NAMA : CINDI DILA APRILIANA**

**NIM : L200200106**

**KELAS : E**

**INFORMATIKA**

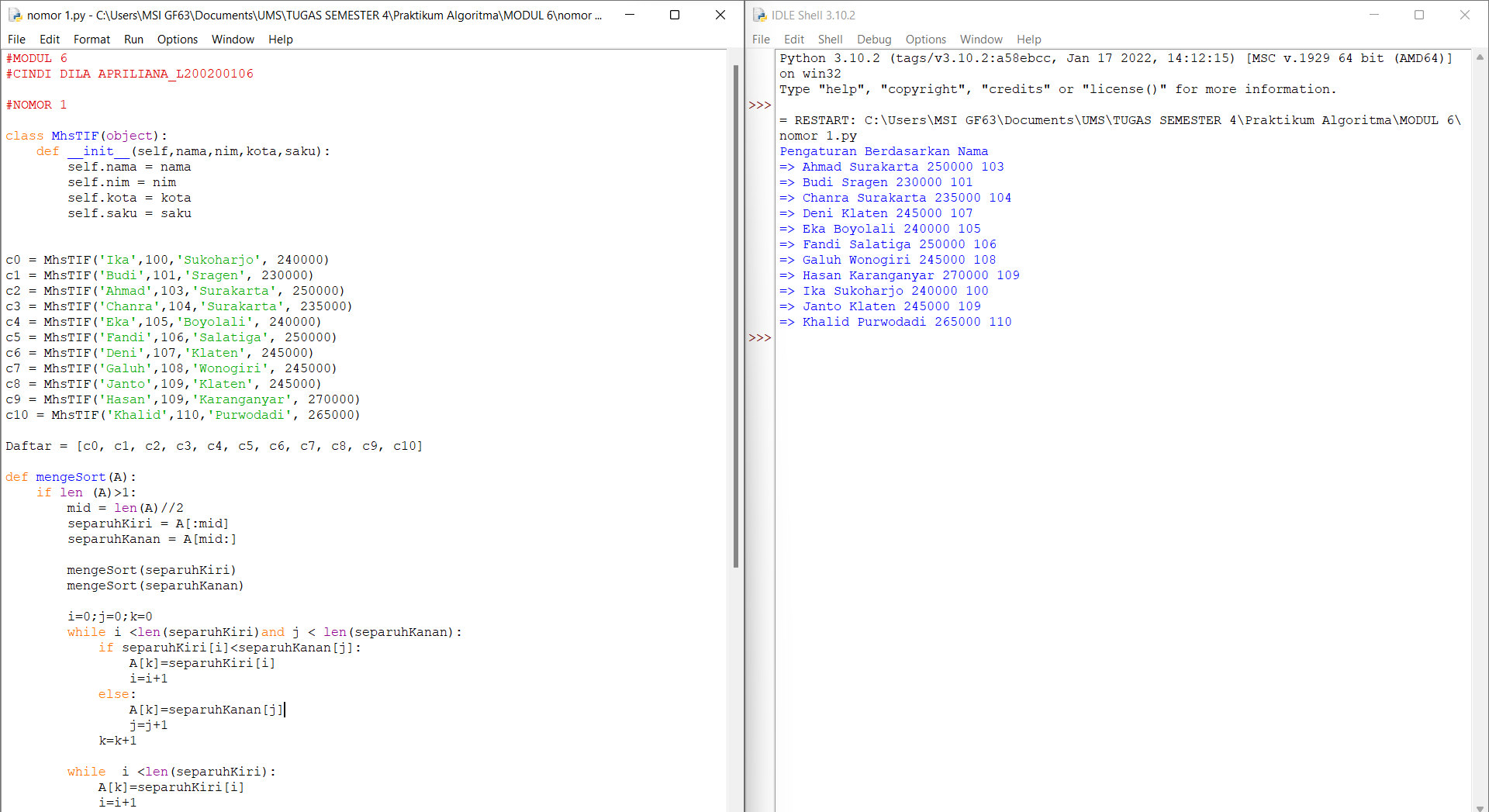
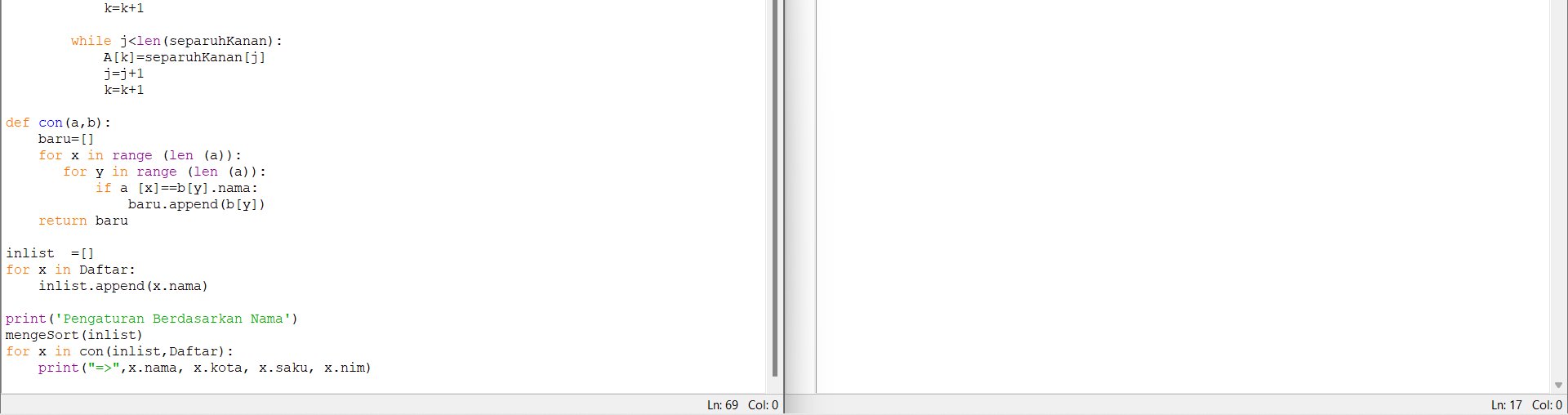
**FAKULTAS KOMUNIKASI DAN INFORMATIKA**

**UNIVERSITAS MUHAMMADIYAH SURAKARTA**

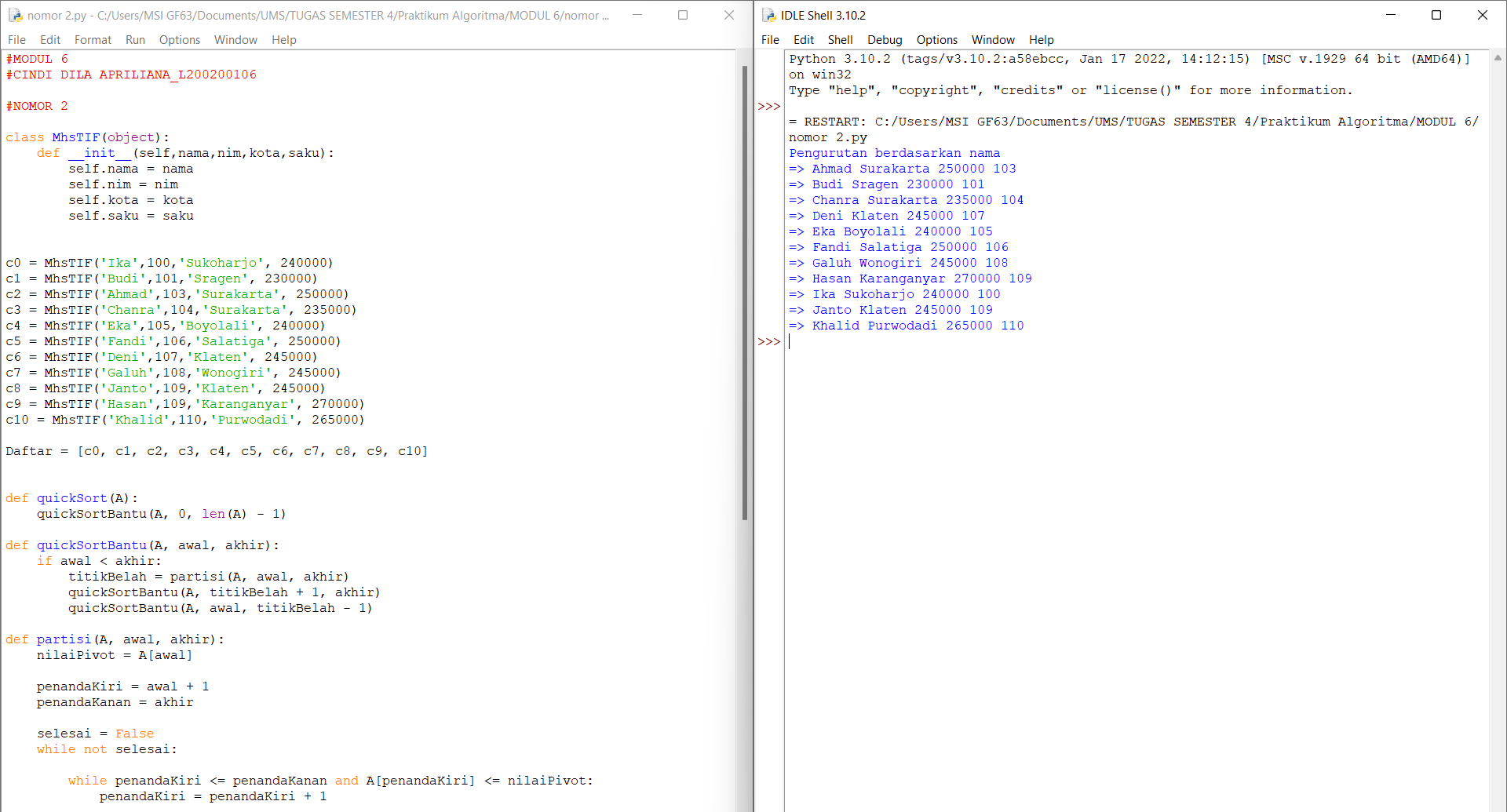
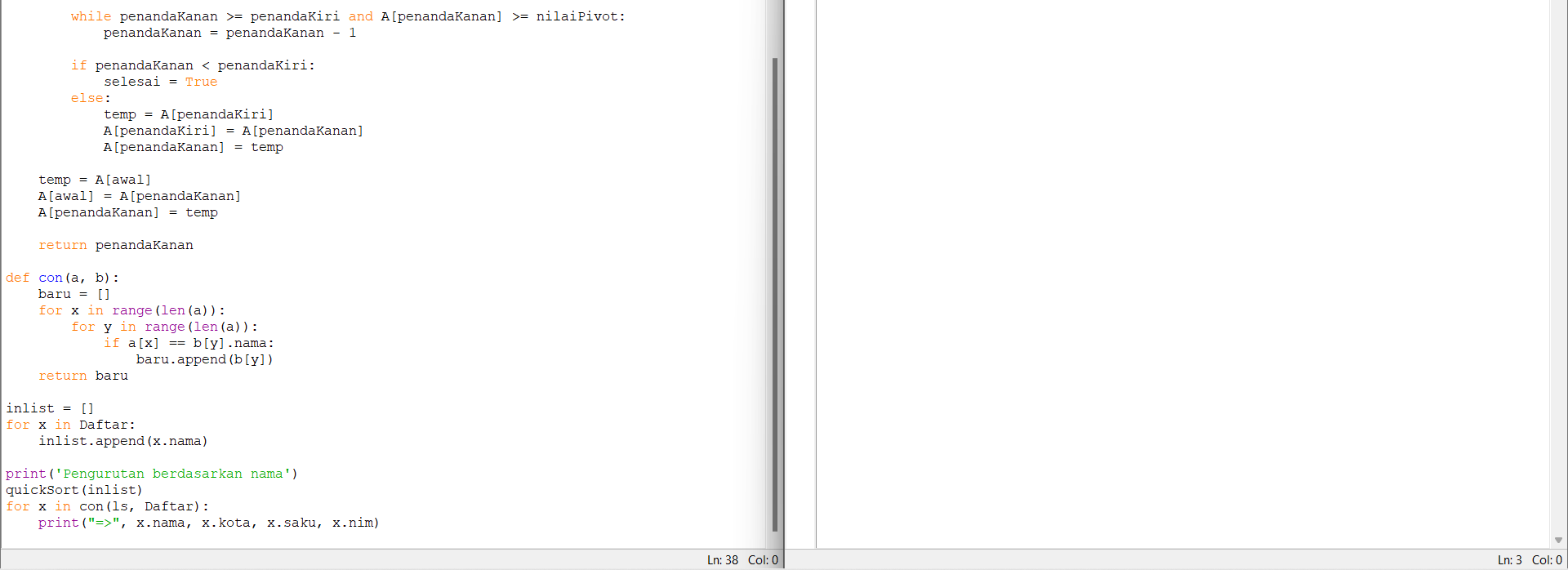
**TAHUN 2021/2022**

1. Ubahlah kode mergeSort dan quickSort di atas agar bisa mengurutkan list yang berisi object-object mhsTIF yang sudah kamu buat di Modul 2. Uji programmu secukupnya.

Soal 1

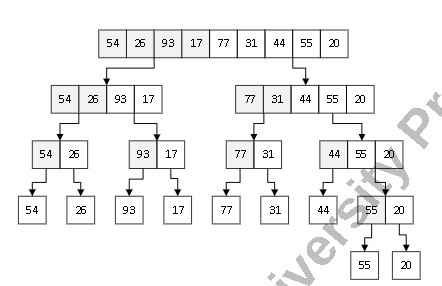


* Soal 1.2



1. Memakai bolpen merah atau biru, tandai dan beri nomer urut eksekusi proses pada Gambar 6.1 dan 6.2, dengan mengacu pada output di halaman 59.

* Merge Sort



1

2

3

4

5

6

7

8

9

10

1

1

1

2

1

3

1

4

1

5

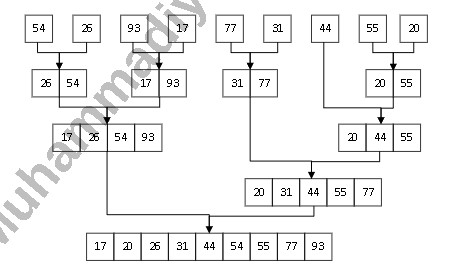
1

6

1

7

* Quick Sort



1

2

4

5

8

9

1

1

1

2

1

3

3

6

1

0

1

4

1

5

7

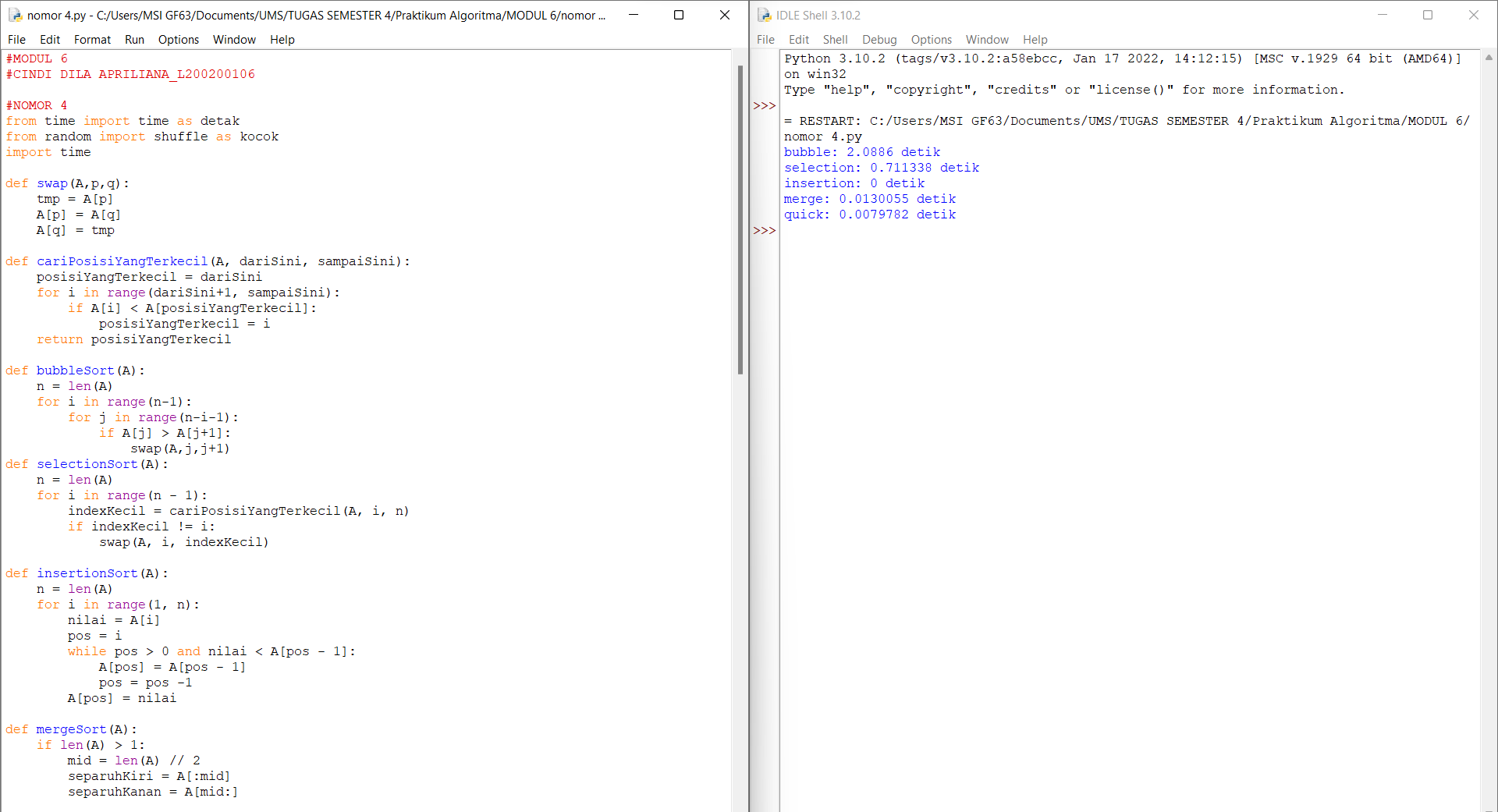
1

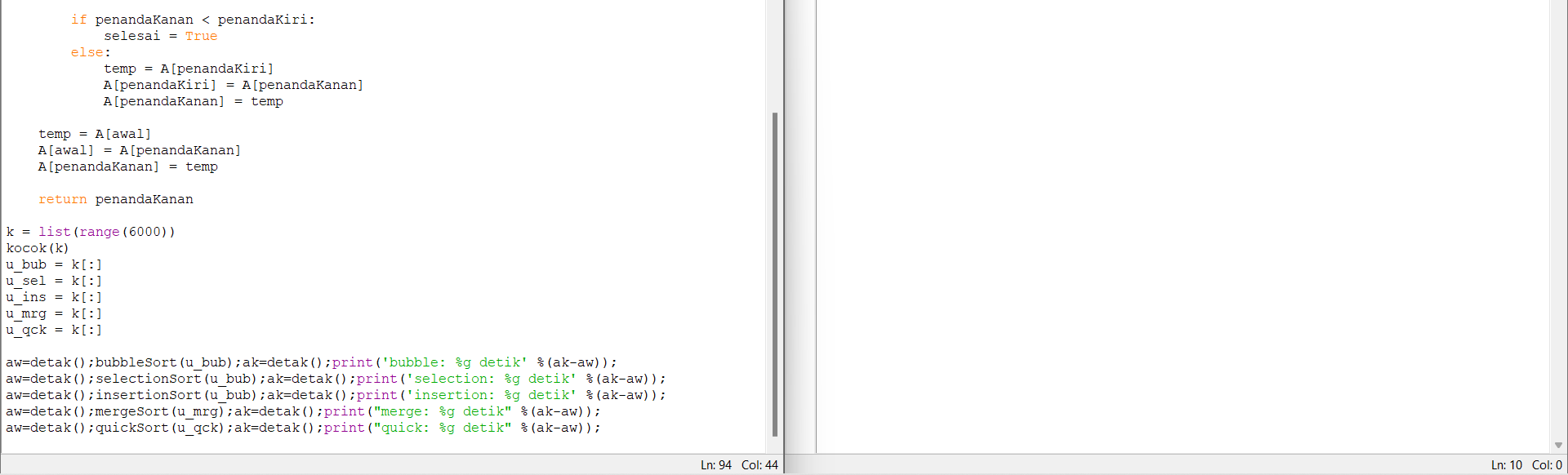
6

1

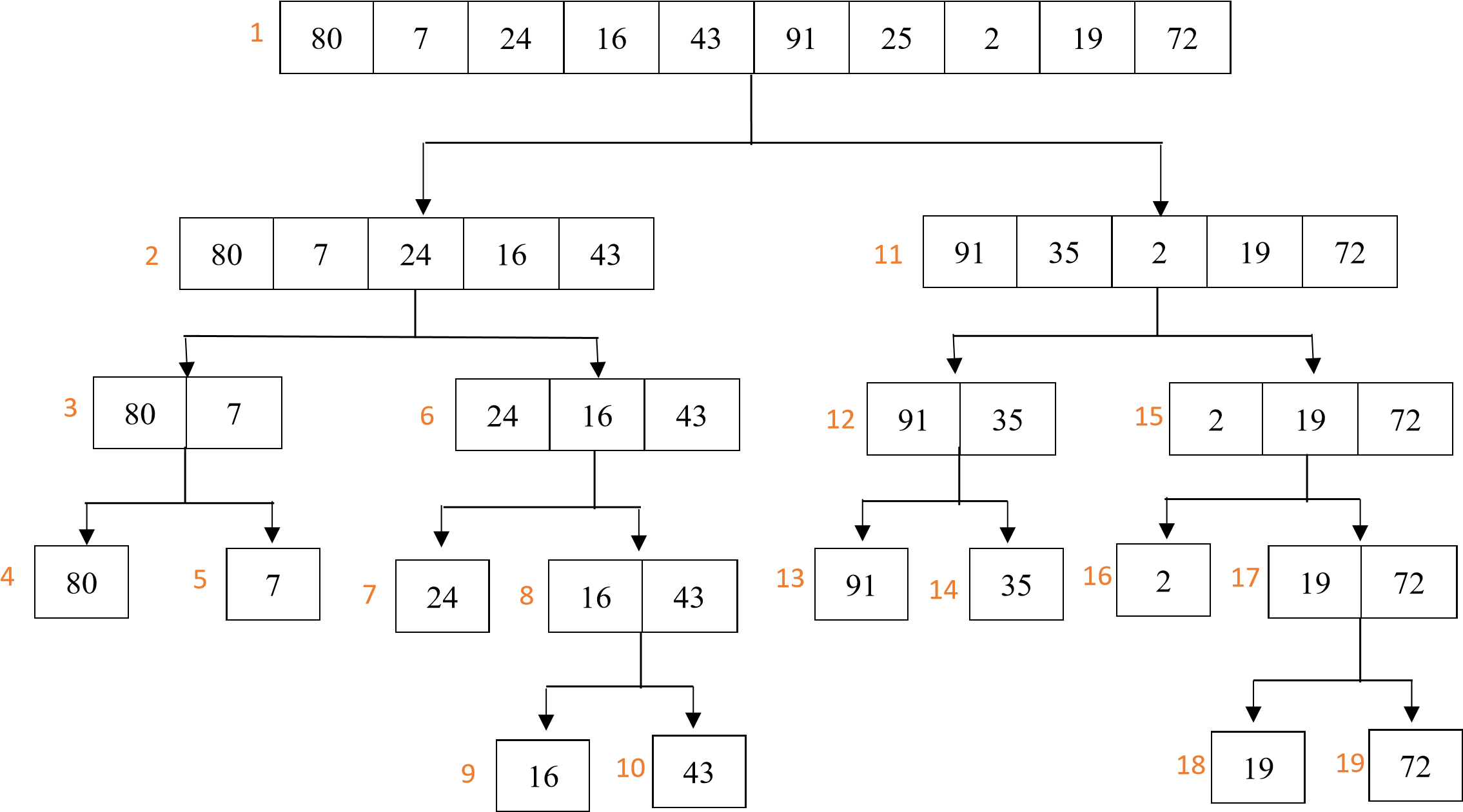
7

1. Uji kecepatan. Ujilah mergeSort dan quickSort di atas (bersama metode sort yang kamu pelajari sebelumnya) dengan kode di bawah ini.

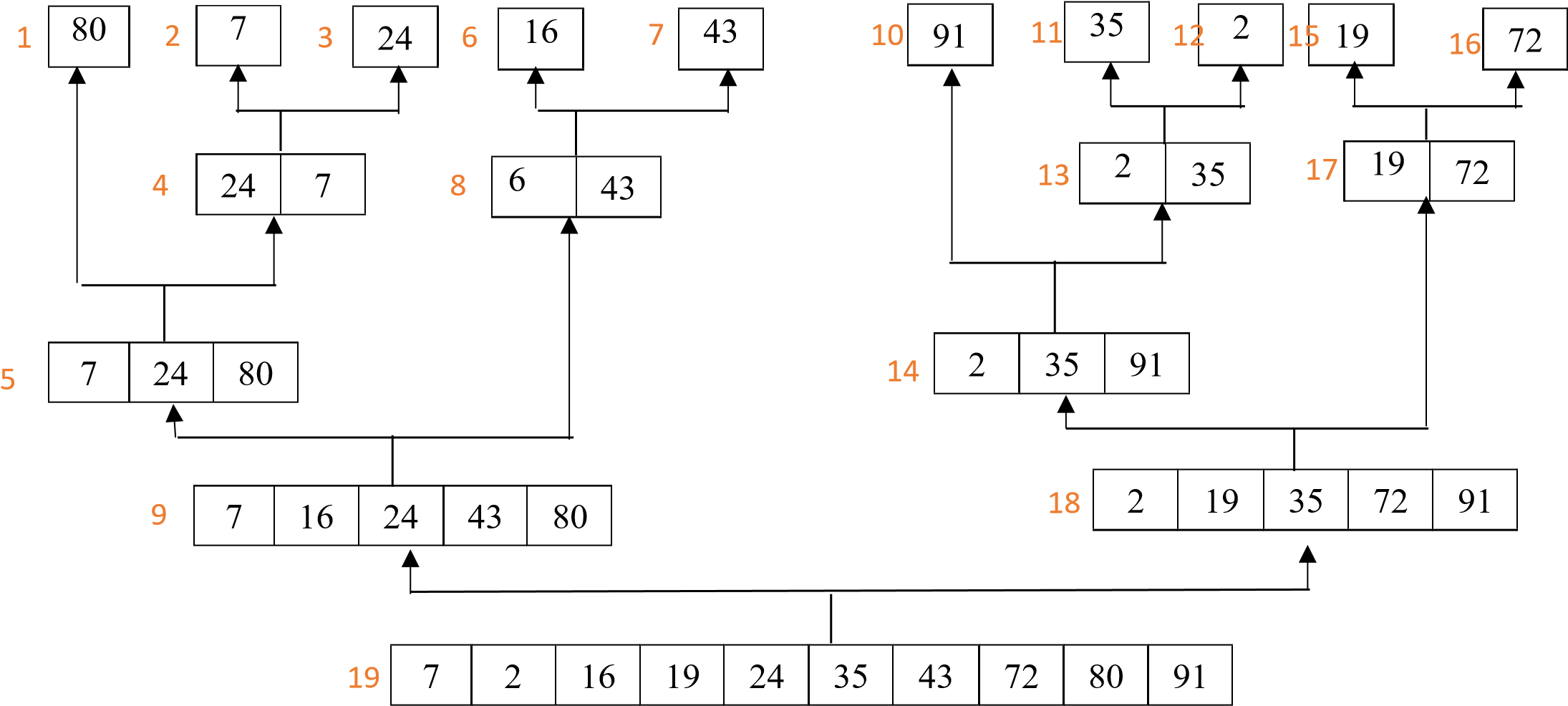




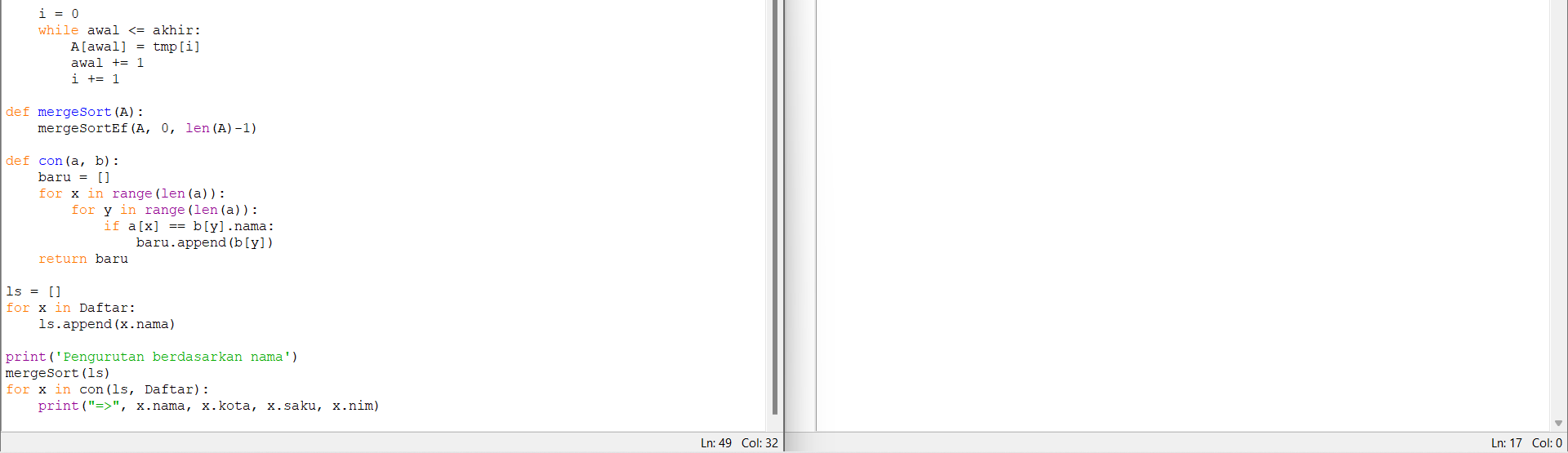
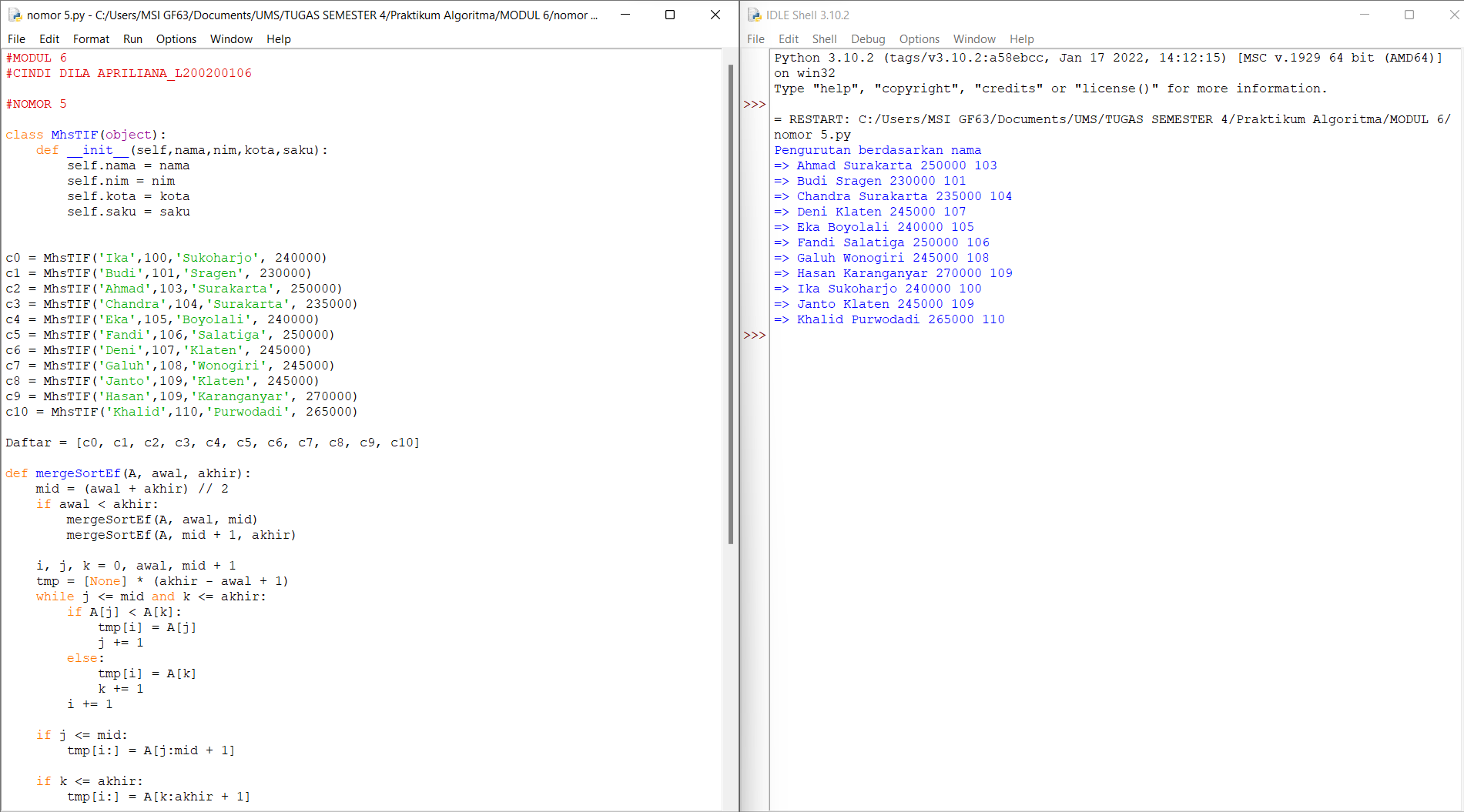
1. Diberikan list L = [80, 7, 24, 16, 43, 91, 35, 2, 19, 72], gambarlah trace pengurutan10 untuk algoritma
2. merge sort



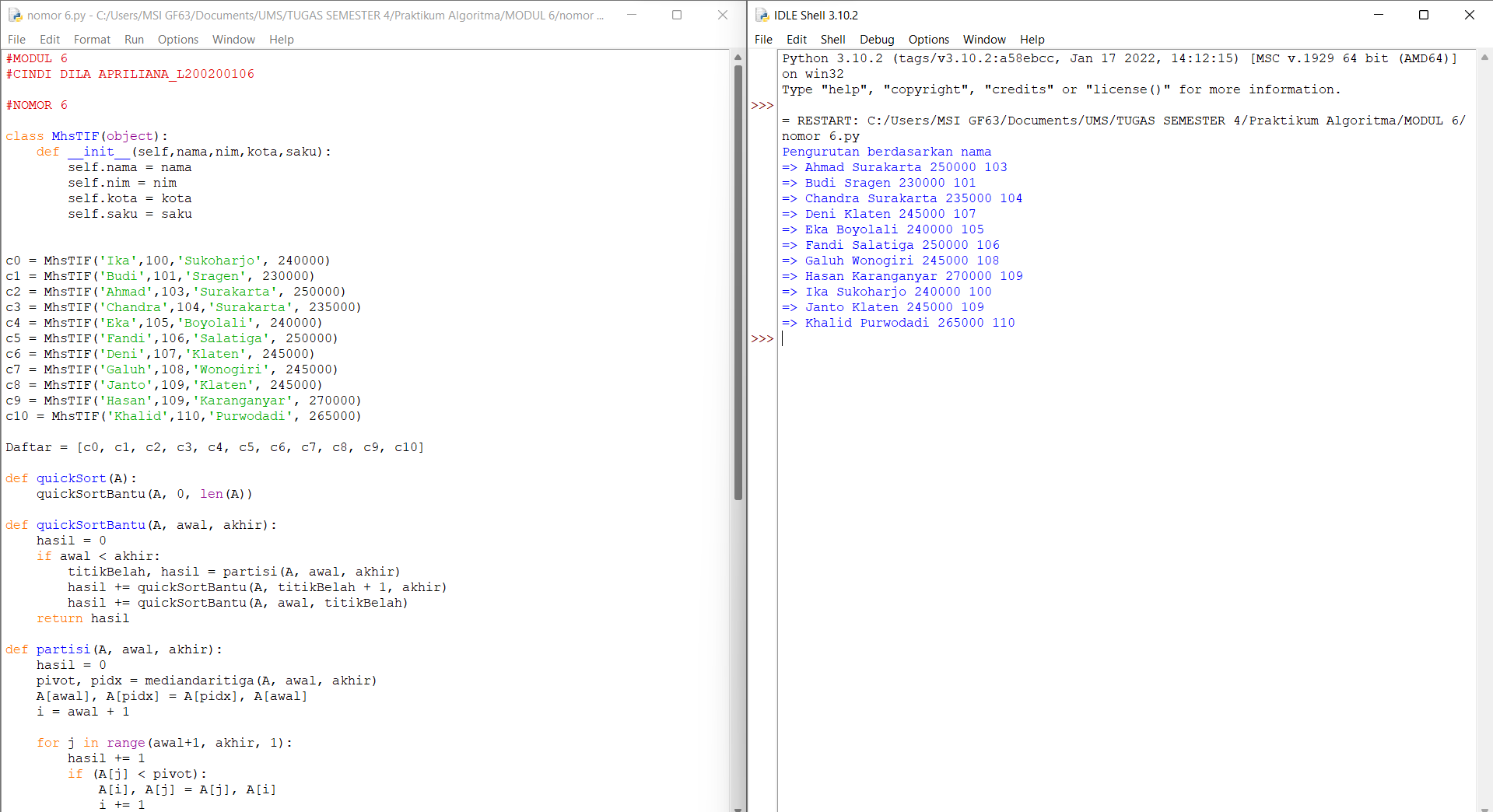
1. quick sort



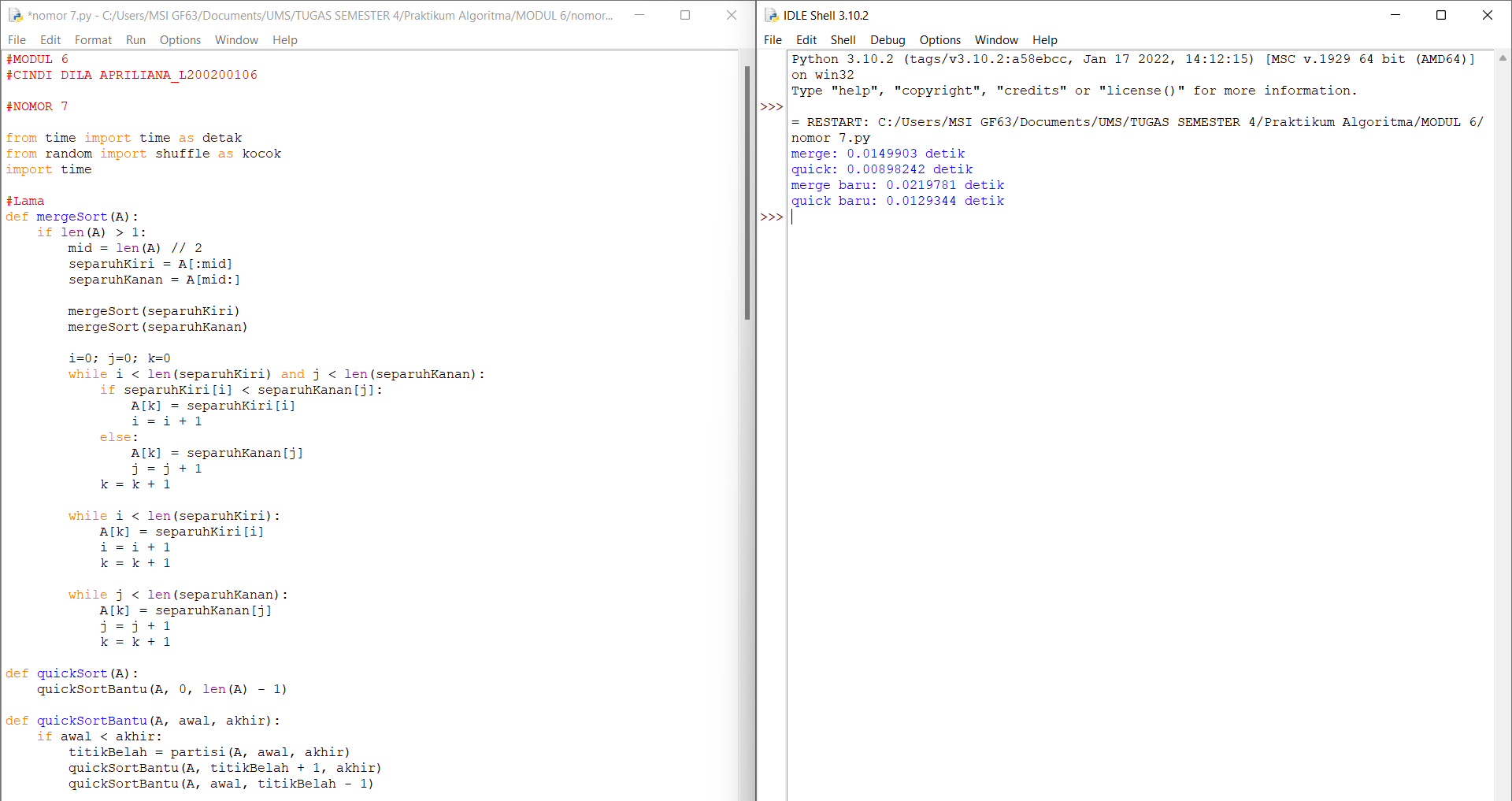
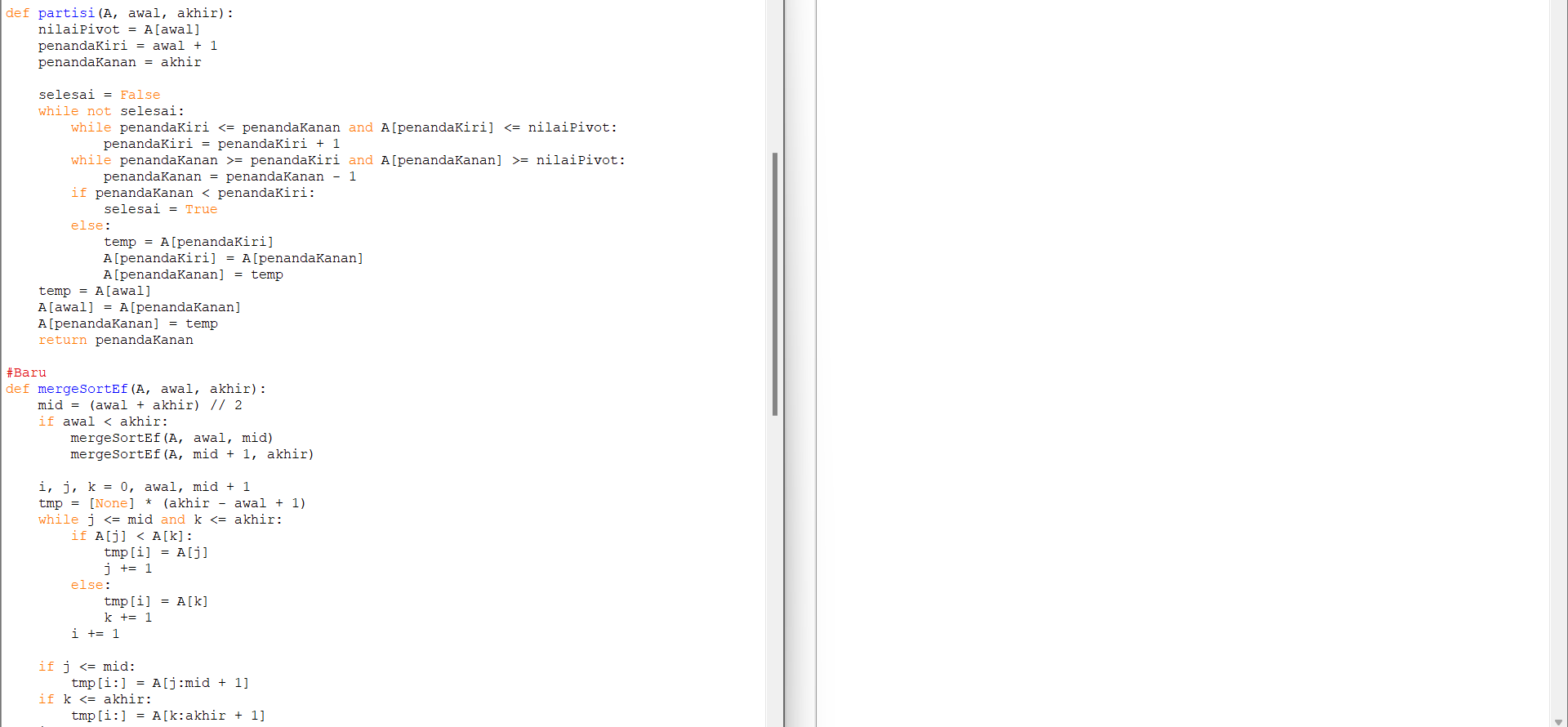
1. Tingkatkan efisiensi program mergeSort dengan tidak memakai operator slice (seperti A[:mid] dan A[mid:]), dan lalu mem-pass index awal dan index akhir bersama listnya saat kita memanggil mergeSort secara rekursif. Kamu akan perlu memisah fungsi mergeSort itu menjadi beberapa fungsi, mirip halnya dengan apa yang dilakukan algoritma quick sorth

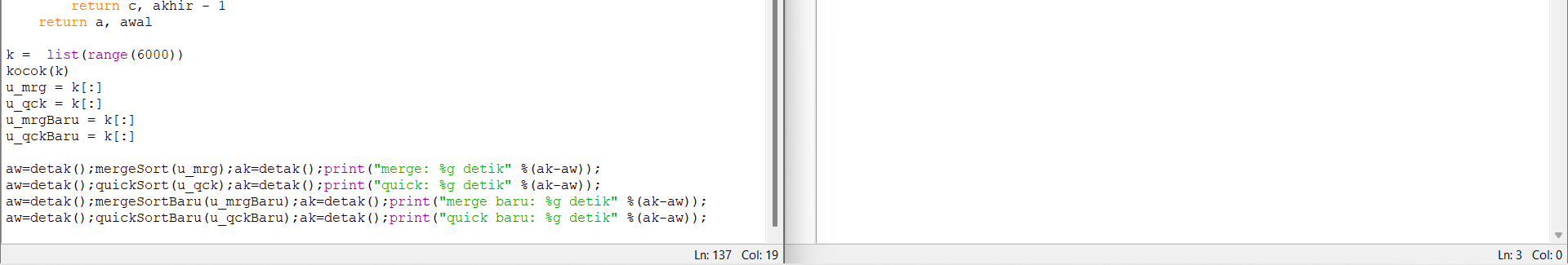
1. Apakah kita bisa meningkatkan efisiensi program quickSort dengan memakai metode median-dari-tiga untuk memilih pivotnya? Ubahlah kodenya dan ujilah.



1. Uji-kecepatan keduanya dan perbandingkan juga dengan kode awalnya.







1. Buatlah versi linked-list untuk program mergeSort di atas.

