**PRAKTIKUM ALGORITMA STRUKTUR DATA**

**TEKNIK INFORMATIKA**

**Prak-06**



Oleh :

Faathir Akbar Nugroho

4522210033

Kelas A

**Pseudocode (Nomor 04)**

**Kamus/Deklarasi Variabel Function Fatirmerge(int Fatirlow, int Fatirmid, int Fatirhigh)**

Fatirlow, Fatirmid, Fatirhigh, Fatirh, Fatiri, Fatirj, Fatirb[], Fatirk = int

**Algoritma/Deskripsi Function Fatirmerge(int Fatirlow, int Fatirmid, int Fatirhigh)**

Fatirh = Fatirlow

Fatiri = Fatirlow

Fatirj = Fatirmid + 1

while ((Fatirh <= Fatirmid) && (Fatirj <= Fatirhigh))

if (Fatira[Fatirh] <= Fatira[Fatirj])

Fatirb[Fatiri] = Fatira[Fatirh]

Fatirh++

else

Fatirb[Fatiri] = Fatira[Fatirj]

Fatirj++

endif

Fatiri++

endwhile

if (Fatirh > Fatirmid)

for (Fatirk = Fatirj; Fatirk <= Fatirhigh; Fatirk++)

Fatirb[Fatiri] = Fatira[Fatirk]

Fatiri++

endfor

else

for (Fatirk = Fatirh; Fatirk <= Fatirmid; Fatirk++)

Fatirb[Fatiri] = Fatira[Fatirk]

Fatiri++

endfor

endif

for (Fatirk = Fatirlow; Fatirk <= Fatirhigh; Fatirk++)

Fatira[Fatirk] = Fatirb[Fatirk]

endfor

**Kamus/Deklarasi Variabel Function Fatirmerge\_sort(int Fatirlow, int Fatirhigh)**

Fatirmid = int

**Algoritma/Deskripsi Function Fatirmerge\_sort(int Fatirlow, int Fatirhigh)**

if (Fatirlow < Fatirhigh)

Fatirmid = (Fatirlow + Fatirhigh) / 2

Fatirmerge\_sort(Fatirlow, Fatirmid)

Fatirmerge\_sort(Fatirmid + 1, Fatirhigh)

Fatirmerge(Fatirlow, Fatirmid, Fatirhigh)

endif

**Kamus/Deklarasi Variabel**

Fatirnum, Fatiri = int

**Algoritma/Deskripsi**

Fatirnum = 5

for (Fatiri = 0; Fatiri < Fatirnum; Fatiri++)

print(Fatira[Fatiri])

endfor

Fatirmerge\_sort(0, Fatirnum - 1)

for (Fatiri = 0; Fatiri < Fatirnum; Fatiri++)

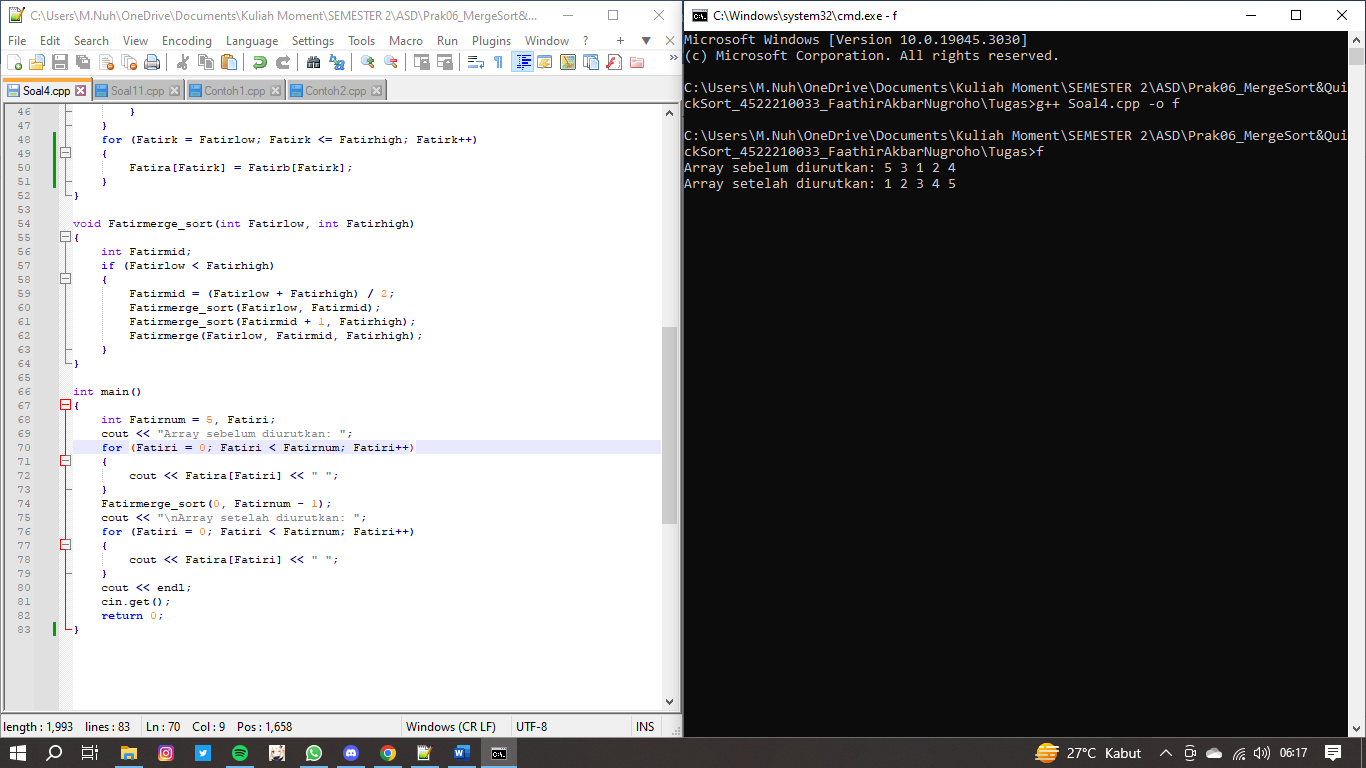
print(Fatira[Fatiri])

endfor

**Algoritma/Bahasa Natural (Nomor 04)**

1. Membuat function Fatirmerge(int Fatirlow, int Fatirmid, int Fatirhigh)
2. Fatirh = Fatirlow
3. Fatiri = Fatirlow
4. Fatirj = Fatirmid + 1
5. Selama ((Fatirh <= Fatirmid) && (Fatirj <= Fatirhigh)), maka kerjakan baris 6 s.d 11, kalau tidak kerjakan baris 12
6. Jika (Fatira[Fatirh] <= Fatira[Fatirj]), maka kerjakan baris 7 s.d 8, kalau tidak kerjakan baris 9 s.d 10
7. Fatirb[Fatiri] = Fatira[Fatirh]
8. Fatirh++
9. Fatirb[Fatiri] = Fatira[Fatirj]
10. Fatirj++
11. Fatiri++
12. Jika (Fatirh > Fatirmid), maka kerjakan baris 13 s.d 17, kalau tidak kerjakan baris 18
13. Fatirk = Fatirj
14. Selama (Fatirk <= Fatirhigh), maka kerjakan baris 15 s.d 17, kalau tidak kerjakan baris 18
15. Fatirb[Fatiri] = Fatira[Fatirk]
16. Fatiri++
17. Fatirk++
18. Fatirk = Fatirh
19. Selama (Fatirk <= Fatirmid), maka kerjakan baris 20 s.d 22, kalau tidak kerjakan baris 23
20. Fatirb[Fatiri] = Fatira[Fatirk]
21. Fatiri++
22. Fatirk++
23. Fatirk = Fatirlow
24. Selama (Fatirk <= Fatirhigh), maka kerjakan baris 25 s.d 26, kalau tidak kerjakan baris 27
25. Fatira[Fatirk] = Fatirb[Fatirk]
26. Fatirk++
27. Membuat function Fatirmerge\_sort(int Fatirlow, int Fatirhigh)
28. Jika (Fatirlow < Fatirhigh)
29. Fatirmid = (Fatirlow + Fatirhigh) / 2
30. Fatirmerge\_sort(Fatirlow, Fatirmid)
31. Fatirmerge\_sort(Fatirmid + 1, Fatirhigh)
32. Fatirmerge(Fatirlow, Fatirmid, Fatirhigh)
33. Fatirnum = 5
34. Fatiri = 0
35. Selama (Fatiri < Fatirnum), maka kerjakan baris 36 s.d 37, kalau tidak kerjakan baris 38
36. Menampilkan isi/nilai variabel (Fatira[Fatiri])
37. Fatiri++
38. Memanggil function Fatirmerge\_sort(0, Fatirnum - 1)
39. Fatiri = 0
40. Selama (Fatiri < Fatirnum), maka kerjakan baris 41 s.d 42, kalau tidak kerjakan baris 43
41. Menampilkan isi/nilai variabel (Fatira[Fatiri])
42. Fatiri++
43. Selesai

**Program (Nomor 04)**



**Pseudocode (Nomor 11)**

**Kamus/Deklarasi Variabel Function Fatirsort()**

arr[], left, right, Fatirpivot, Fatirl\_hold, Fatir\_hold = int

**Algoritma/Deskripsi Function Fatirsort()**

Fatirl\_hold = Fatirleft

Fatir\_hold = Fatirright

Fatirpivot = FatirNumbers[Fatirleft]

while (Fatirleft < Fatirright)

while ((FatirNumbers[Fatirright] >= Fatirpivot) && (Fatirleft < Fatirright))

Fatirright--

endwhile

if (Fatirleft != Fatirright)

FatirNumbers[Fatirleft] = FatirNumbers[Fatirright];

Fatirleft++

endif

while ((FatirNumbers[Fatirleft] <= Fatirpivot) && (Fatirleft < Fatirright))

Fatirleft++

endwhile

if (Fatirleft != Fatirright)

FatirNumbers[Fatirright] = FatirNumbers[Fatirleft]

Fatirright--

endif

endwhile

FatirNumbers[Fatirleft] = Fatirpivot

Fatirpivot = Fatirleft

Fatirleft = Fatirl\_hold

Fatirright = Fatir\_hold

if (Fatirleft < Fatirpivot)

FatirSort(FatirNumbers, Fatirleft, Fatirpivot - 1).Fatirsort()

endif

if (Fatirright > Fatirpivot)

FatirSort(FatirNumbers, Fatirpivot + 1, Fatirright).Fatirsort()

endif

**Kamus/Deklarasi Variabel**

FatirNumList[], FatirNumbers, Fatirleft, Fatirright = int

**Algoritma/Deskripsi**

class FatirSort

private: int FatirNumbers, int Fatirleft, int Fatirright

public: FatirSort(int arr[], int left, int right)

FatirNumbers = arr

Fatirleft = left

Fatirright = right

FatirNumList[9] = { 65,2,44,26,19,22,5,3,12}

for (int Fatird = 0; Fatird < 9; Fatird++)

print(FatirNumList[Fatird])

endfor

FatirSort(FatirNumList, 0, 9).Fatirsort()

for (int Fatiri = 0; Fatiri < 9; Fatiri++)

print(FatirNumList[Fatiri])

endfor

**Algoritma/Bahasa Natural (Nomor 11)**

1. Membuat function Fatirsort()
2. Fatirl\_hold = Fatirleft
3. Fatir\_hold = Fatirright
4. Fatirpivot = FatirNumbers[Fatirleft]
5. Selama (Fatirleft < Fatirright), maka kerjakan baris 6 s.d 15, kalau tidak kerjakan baris 16
6. Selama ((FatirNumbers[Fatirright] >= Fatirpivot) && (Fatirleft < Fatirright)), maka kerjakan baris 7, kalau tidak kerjakan baris 8
7. Fatirright--
8. Jika (Fatirleft != Fatirright), maka kerjakan baris 9 s.d 10, kalau tidak kerjakan baris 11
9. FatirNumbers[Fatirleft] = FatirNumbers[Fatirright]
10. Fatirleft++
11. Selama ((FatirNumbers[Fatirleft] <= Fatirpivot) && (Fatirleft < Fatirright)), maka kerjakan baris 12, kalau tidak kerjakan baris 13
12. Fatirleft++
13. Jika (Fatirleft != Fatirright), maka kerjakan baris 14 s.d 15, kalau tidak kerjakan baris 16
14. FatirNumbers[Fatirright] = FatirNumbers[Fatirleft]
15. Fatirright--
16. FatirNumbers[Fatirleft] = Fatirpivot
17. Fatirpivot = Fatirleft
18. Fatirleft = Fatirl\_hold
19. Fatirright = Fatir\_hold
20. Jika (Fatirleft < Fatirpivot), maka kerjakan baris 21, kalau tidak kerjakan baris 22
21. FatirSort(FatirNumbers, Fatirleft, Fatirpivot - 1).Fatirsort()
22. Jika (Fatirright > Fatirpivot), maka kerjakan baris 23, kalau tidak kerjakan baris 24
23. FatirSort(FatirNumbers, Fatirpivot + 1, Fatirright).Fatirsort()
24. Mendeklarasikan class (class FatirSort (FatirNumbers, Fatirleft, Fatirright)) dengan penentu akses private
25. Mendeklarasikan class (class FatirSort (FatirSort(int arr[], int left, int right))) dengan penentu akses public
26. FatirNumList[9] = {65,2,44,26,19,22,5,3,12}
27. Fatird = 0
28. Selama (Fatird < 9), maka kerjakan baris 29 s.d 30, kalau tidak kerjakan baris 31
29. Menampilkan isi/nilai variabel (FatirNumList[Fatird])
30. Fatird++
31. Memanggil function FatirSort(FatirNumList, 0, 9).Fatirsort()
32. Fatiri = 0
33. Selama (Fatiri < 9), maka kerjakan baris 34 s.d 35, kalau tidak kerjakan baris 36
34. Menampilkan isi/nilai variabel (FatirNumList[Fatiri])
35. Fatiri++
36. Selesai

**Program (Nomor 11)**

