**PRAKTIKUM ALGORITMA STRUKTUR DATA**

**TEKNIK INFORMATIKA**

**(Contoh)**



Oleh :

Faathir Akbar Nugroho

4522210033

Kelas A

**Pseudocode (Contoh 1)**

**Kamus/Deklarasi Variabel FatirSearch01(int Fatirtika[], int FatirJmlElemenArray, int Elemen)**

-

**Algoritma/Deskripsi FatirSearch01(int Fatirtika[], int FatirJmlElemenArray, int FatirElemen)**

int Fatirflag = -1

for (int Fatircount = 0; Fatircount < FatirJmlElemenArray; Fatircount++)

if (FatirElemen == Fatirtika[Fatircount])

Fatirflag = Fatircount

endif

endfor

return Fatirflag

**Kamus/Deklarasi Variabel**

-

**Algoritma/Deskripsi**

const int FatirJmlElemenArray = 11

int Fatirtika[FatirJmlElemenArray] = { 22, 61, 15, 66, 18, 25, 34, 87, 55, 45, 10 }

for (int Fatircount = 0; Fatircount < FatirJmlElemenArray; Fatircount++)

print((Fatircount+1),( Fatirtika[Fatircount]))

endfor

int FatirSearchElemen = 0

int Fatirflag = 0

input(FatirSearchElemen)

Fatirflag = FatirSearch01(Fatirtika, FatirJmlElemenArray, FatirSearchElemen)

if (Fatirflag != -1)

print(Fatirflag+1)

else

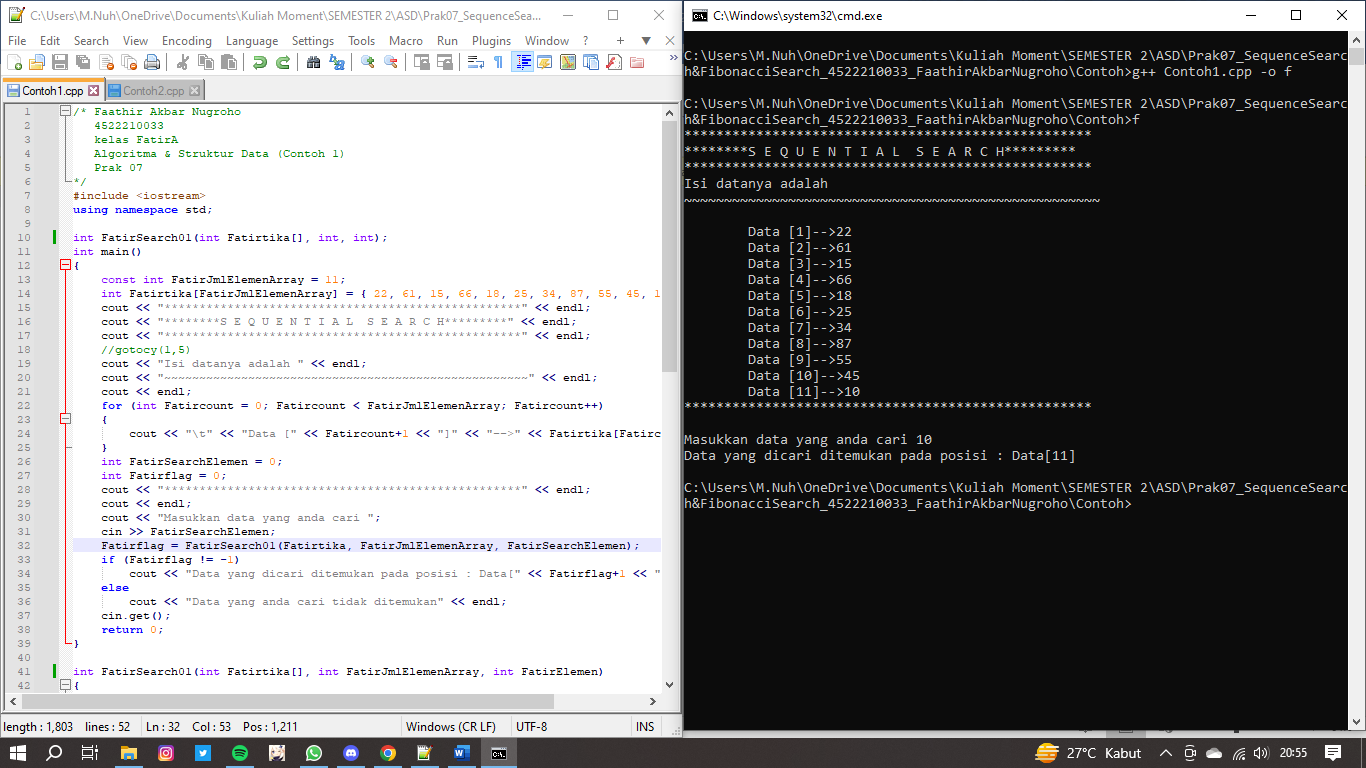
print("Data yang anda cari tidak ditemukan")

endif

**Algoritma/Bahasa Natural (Contoh 1)**

1. Membuat function FatirSearch01(int Fatirtika[], int FatirJmlElemenArray, int FatirElemen)
2. Fatirflag = -1
3. Fatircount = 0
4. Selama (Fatircount < FatirJmlElemenArray), maka kerjakan baris 5 s.d 7, kalau tidak kerjakan baris 8
5. Jika (FatirElemen == Fatirtika[Fatircount]), maka kerjakan baris 6, kalaut tidak kerjakan baris 7
6. Fatirflag = Fatircount
7. Fatircount++
8. Kembali ke Fatirflag
9. FatirJmlElemenArray = 11
10. Fatirtika[FatirJmlElemenArray] = { 22, 61, 15, 66, 18, 25, 34, 87, 55, 45, 10 }
11. Fatircount = 0
12. Selama (Fatircount < FatirJmlElemenArray), maka kerjakan baris 13 s.d 14, kalau tidak kerjakan baris 15
13. Menampilkan isi/nilai variabel ((Fatircount+1),( Fatirtika[Fatircount]))
14. Fatircount++
15. FatirSearchElemen = 0
16. Fatirflag = 0
17. Memasukkan isi/nilai variabel FatirSearchElemen
18. Fatirflag = FatirSearch01(Fatirtika, FatirJmlElemenArray, FatirSearchElemen)
19. Jika (Fatirflag != -1), maka kerjakan baris 20, kalau tidak kerjakan baris 21
20. Menampilkan isi/nilai variabel (Fatirflag+1)
21. Menampilkan ("Data yang anda cari tidak ditemukan")
22. Selesai

**Program (Contoh 1)**



**Pseudocode (Contoh 2)**

**Kamus/Deklarasi Variabel Function Fatirfibonacci(int \*FatirF)**

-

**Algoritma/Deskripsi**

FatirF[0] = 1

FatirF[1] = 1

for (int Fatiri = 2; Fatiri < FatirMAXSIZE; Fatiri++)

FatirF[Fatiri] = FatirF[Fatiri - 2] + FatirF[Fatiri - 1]

endfor

**Kamus/Deklarasi Variabel Fatirfibonacci\_search(int \*Fatira, int Fatirkey, int Fatirn)**

FatirF[] = int

**Algoritma/Deskripsi Fatirfibonacci\_search(int \*Fatira, int Fatirkey, int Fatirn)**

int FatirRendah = 0, FatirTinggi = Fatirn - 1

int Fatirmid = 0

int Fatirk = 0

Fatirfibonacci(FatirF)

while (Fatirn > FatirF[Fatirk] - 1)

++Fatirk

endwhile

for (int Fatiri = Fatirn; Fatiri < FatirF[Fatirk] - 1; ++Fatiri)

Fatira[Fatiri] = Fatira[FatirTinggi]

endfor

while (FatirRendah <= FatirTinggi)

Fatirmid = FatirRendah + FatirF[Fatirk - 1] - 1

if (Fatira[Fatirmid] > Fatirkey)

FatirTinggi = Fatirmid - 1

Fatirk = Fatirk - 1

else if (Fatira[Fatirmid] < Fatirkey)

FatirRendah = Fatirmid + 1

Fatirk = Fatirk - 2

else

if (Fatirmid <= FatirTinggi)

return Fatirmid

else

return -1

endif

endif

endwhile

return -1

Kamus/Deklarasi Variabel

Fatirk = int

Algoritma/Deskripsi

int Fatira[FatirMAXSIZE] = {1, 4, 6, 8, 9, 11, 23, 35, 47, 51, 55, 63, 64, 78, 88, 95, 99}

input(Fatirk)

int Fatirpos = Fatirfibonacci\_search(Fatira, Fatirk, 17)

if (Fatirpos != -1)

print((Fatirpos + 1),( Fatirk))

else

print(“Data tidak ditemukan”)

endif

**Algoritma/Bahasa Natural (Contoh 2)**

1. Membuat function Fatirfibonacci(int \*FatirF)
2. FatirF[0] = 1
3. FatirF[1] = 1
4. Fatiri = 2
5. Selama (Fatiri < FatirMAXSIZE), maka kerjakan baris 6 s.d 7, kalau tidak kerjakan baris 8
6. FatirF[Fatiri] = FatirF[Fatiri - 2] + FatirF[Fatiri - 1]
7. Fatiri++
8. Membuat function Fatirfibonacci\_search(int \*Fatira, int Fatirkey, int Fatirn)
9. FatirRendah = 0
10. FatirTinggi = Fatirn - 1
11. Fatirmid = 0
12. Fatirk = 0
13. Memanggil function Fatirfibonacci(FatirF)
14. Selama (Fatirn > FatirF[Fatirk] - 1), maka kerjakan baris 15, kalau tidak kerjakan baris 16
15. Fatiri = Fatirn
16. Selama (Fatiri < FatirF[Fatirk] - 1), maka kerjakan baris 17 s.d 18, kalau tidak kerjakan baris 19
17. Fatira[Fatiri] = Fatira[FatirTinggi]
18. ++Fatiri
19. Selama (FatirRendah <= FatirTinggi), maka kerjakan baris 20 s.d 29, kalau tidak kerjakan baris 30
20. Fatirmid = FatirRendah + FatirF[Fatirk - 1] - 1
21. Jika (Fatira[Fatirmid] > Fatirkey), maka kerjakan baris 22 s.d 23, kalau tidak kerjakan baris 24
22. FatirTinggi = Fatirmid - 1
23. Fatirk = Fatirk - 1
24. Jika (Fatira[Fatirmid] < Fatirkey), maka kerjakan baris 25 s.d 26, kalau tidak kerjakan baris 27
25. FatirRendah = Fatirmid + 1
26. Fatirk = Fatirk - 2
27. Jika (Fatirmid <= FatirTinggi), maka kerjakan baris 28, kalau tidak kerjakan baris 29
28. Kembali ke Fatirmid
29. Kembali -1
30. Kembali -1
31. Fatira[FatirMAXSIZE] = {1, 4, 6, 8, 9, 11, 23, 35, 47, 51, 55, 63, 64, 78, 88, 95, 99}
32. Memasukkan isi/nilai variabel Fatirk
33. Fatirpos = Fatirfibonacci\_search(Fatira, Fatirk, 17)
34. Jika (Fatirpos != -1), maka kerjakan baris 35, kalau tidak kerjakan baris 36
35. Menampilkan isi/nilai variabel ((Fatirpos + 1),( Fatirk))
36. Menampilkan (“Data tidak ditemukan”)
37. Selesai

**Program (Contoh 2)**

