**PRAKTIKUM ALGORITMA STRUKTUR DATA**

**TEKNIK INFORMATIKA**

**(Contoh)**



Oleh :

Faathir Akbar Nugroho

4522210033

Kelas A

**Pseudocode (Contoh 1)**

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

**-**

**Algoritma/Deskripsi Function FatirInisialisasi()**

FatirRoot = NULL

FatirP = NULL

**Kamus/Deklarasi Variabel Function FatirBuatSimpul(char FatirX)**

**-**

**Algoritma/Deskripsi Function FatirBuatSimpul(char FatirX)**

FatirP=(FatirSimpul\*) malloc(sizeof(FatirSimpul))

if(FatirP != NULL)

FatirP->FatirINFO = FatirX

FatirP->FatirLEFT = NULL

FatirP->FatirRIGHT = NULL

else

print(“Pembuatan Simpul Gagal \n”)

endif

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

**-**

**Algoritma/Deskripsi Function FatirBuatSimpulAkar()**

if(FatirRoot == NULL)

if(FatirP !=NULL)

FatirRoot = FatirP

FatirP->FatirLEFT = NULL

FatirP->FatirRIGHT = NULL

else

print(“Simpul Belum dibuat \n”)

endif

else

print(“Pohon Sudah Ada !!!! \n”)

endif

**Kamus/Deklarasi Variabel Function FatirInsertUrutNomer(char FatirInput[6])**

Fatiri, Fatirj, FatirFlag = int

Fatirx = char

**Algoritma/Deskripsi Function FatirInsertUrutNomer(char FatirInput[6])**

FatirFlag = 0

Fatiri=1

Fatirj=1

FatirQ[Fatiri] = FatirRoot

while(FatirFlag == 0 && Fatirj < 6)

FatirX = FatirInput[Fatirj-1]

if(FatirX != '0')

FatirBuatSimpul(FatirX)

FatirCurrent = FatirQ[Fatiri]

FatirCurrent -> FatirLEFT = FatirP

Fatirj++

FatirQ[Fatirj] = FatirP

else

FatirFlag = 1

Fatirj++

FatirQ[Fatirj] = NULL

endif

if(FatirFlag == 0)

FatirX = FatirInput[Fatirj-1]

if(FatirX !='0')

FatirBuatSimpul(FatirX)

FatirCurrent->FatirRIGHT =FatirP

Fatirj++

FatirQ[Fatirj] = FatirP

else

FatirFlag = 1

Fatirj++

FatirQ[Fatirj] = NULL

endif

endwhile

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

Fatiri, Fatirj, Fatirn, FatirCounter = int

**Algoritma/Deskripsi Function FatirBacaUrutNomer()**

Fatiri=1;Fatirj=1;Fatirn=1;FatirCounter=0

int Fatirlevel=0

while(FatirQ[Fatiri] != NULL)

FatirCurrent = FatirQ[Fatiri]

if(Fatiri == 1)

print(Fatirlevel)

endif

print(FatirCurrent->FatirINFO)

FatirCounter++

if(FatirCounter == Fatirn)

Fatirlevel++

print(Fatirlevel)

endif

if(FatirCounter == Fatirn)

FatirCounter = 0

Fatirn = Fatirn\*2

endif

if(FatirCurrent->FatirLEFT !=NULL)

Fatirj++

FatirQ[Fatiri] = FatirCurrent->FatirLEFT

endif

if(FatirCurrent->FatirRIGHT !=NULL)

Fatirj++

FatirQ[Fatiri]=FatirCurrent->FatirRIGHT

endif

Fatiri++

endwhile

**Kamus/Deklarasi Variabel**

FatirINFO, FatirX = char

**Algoritma/Deskripsi**

struct FatirNode

char FatirINFO

struct FatirNode\* FatirLEFT

struct FatirNode\* FatirRIGHT

typedef struct FatirNode FatirSimpul

FatirSimpul \*FatirRoot, \*FatirP, \*FatirQ[30], \*FatirR, \*FatirCurrent

char Fatirroot = 'C'

char FatirDaun[6] = {'A', 'B', 'D', 'E', 'F', 'G'}

FatirInisialisasi()

FatirBuatSimpul(Fatirroot)

FatirBuatSimpulAkar()

FatirInsertUrutNomer(FatirDaun)

FatirBacaUrutNomer()

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

1. Membuat function FatirInisialisasi()
2. FatirRoot = NULL
3. FatirP = NULL
4. Membuat function FatirBuatSimpul(char FatirX)
5. FatirP=(FatirSimpul\*) malloc(sizeof(FatirSimpul))
6. Jika (FatirP != NULL), maka kerjakan baris 7 s.d 9, kalau tidak kerjakan baris 10
7. FatirP->FatirINFO = FatirX
8. FatirP->FatirLEFT = NULL
9. FatirP->FatirRIGHT = NULL
10. Menampilkan (“Pembuatan Simpul Gagal \n”)
11. Membuat function FatirBuatSimpulAkar()
12. Jika (FatirRoot == NULL), maka kerjakan baris 13 s.d 17, kalau tidak kerjakan baris 18
13. Jika (FatirP !=NULL), maka kerjakan baris 14 s.d 16, kalau tidak kerjakan baris 17
14. FatirRoot = FatirP
15. FatirP->FatirLEFT = NULL
16. FatirP->FatirRIGHT = NULL
17. Menampilkan (“Simpul Belum dibuat \n”)
18. Menampilkan (“Pohon Sudah Ada !!!! \n”)
19. Membuat function FatirInsertUrutNomer(char FatirInput[6])
20. FatirFlag = 0
21. Fatiri=1
22. Fatirj=1
23. FatirQ[Fatiri] = FatirRoot
24. Selama (FatirFlag == 0 && Fatirj < 6), maka kerjakan baris 25 s.d 44, kalau tidak kerjakan baris 45
25. FatirX = FatirInput[Fatirj-1]
26. Jika (FatirX != '0'), maka kerjakan baris 27 s.d 31, kalau tidak kerjakan baris 32 s.d 34
27. FatirBuatSimpul(FatirX)
28. FatirCurrent = FatirQ[Fatiri]
29. FatirCurrent -> FatirLEFT = FatirP
30. Fatirj++
31. FatirQ[Fatirj] = FatirP
32. FatirFlag = 1
33. Fatirj++
34. FatirQ[Fatirj] = NULL
35. Jika (FatirFlag == 0), maka kerjakan baris 36 s.d 44, kalau tidak kerjakan baris 45
36. FatirX = FatirInput[Fatirj-1]
37. Jika (FatirX !='0'), maka kerjakan baris 38 s.d 41, kalau tidak kerjakan baris 42 s.d 44
38. FatirBuatSimpul(FatirX)
39. FatirCurrent->FatirRIGHT =FatirP
40. Fatirj++
41. FatirQ[Fatirj] = FatirP
42. FatirFlag = 1
43. Fatirj++
44. FatirQ[Fatirj] = NULL

45. Membuat function FatirBacaUrutNomer()

46. Fatiri=1;Fatirj=1;Fatirn=1;FatirCounter=0

47. int Fatirlevel=0

48. Selama (FatirQ[Fatiri] != NULL), maka kerjakan baris 49 s,d 66, kalau tidak kerjakan baris 67

49. FatirCurrent = FatirQ[Fatiri]

50. Jika (Fatiri == 1), maka kerjakan baris 51, kalau tidak kerjakan baris 52

51. Menampilkan isi/nilai variabel Fatirlevel

52. Menampilkan isi/nilai variabel (FatirCurrent->FatirINFO)

53. FatirCounter++

54. Jika (FatirCounter == Fatirn), maka kerjakan baris 55 s.d 56, kalau tidak kerjakan baris 57

55. Fatirlevel++

56. Menampilkan isis/nilai variabel Fatirlevel

57. Jika (FatirCounter == Fatirn), maka kerjakan baris 58 s.d 59, kalau tidak kerjakan baris 60

58. FatirCounter = 0

59. Fatirn = Fatirn\*2

60. Jika (FatirCurrent->FatirLEFT !=NULL), maka kerjakan baris 61 s.d 62, kalau tidak kerjakan baris 63

61. Fatirj++

62. FatirQ[Fatiri] = FatirCurrent->FatirLEFT

63. Jika (FatirCurrent->FatirRIGHT !=NULL), maka kerjakan baris 64 s.d 65, kalau tidak kerjakan baris 66

64. Fatirj++

65. FatirQ[Fatiri]=FatirCurrent->FatirRIGHT

66. Fatiri++

67. Mendeklarasikan struct (struct FatirNode(char FatirINFO, struct FatirNode\* FatirLEFT, struct FatirNode\* FatirRIGHT))

68. Mendefinisikan struct (FatirNode FatirSimpul)

69. Mendefinisikan struct (FatirSimpul \*FatirRoot, \*FatirP, \*FatirQ[30], \*FatirR, \*FatirCurrent)

70. char Fatirroot = 'C'

71. char FatirDaun[6] = {'A', 'B', 'D', 'E', 'F', 'G'}

72. Memanggil function FatirInisialisasi()

73. Memanggil function FatirBuatSimpul(Fatirroot)

74. Memanggil function FatirBuatSimpulAkar()

75. Memanggil function FatirInsertUrutNomer(FatirDaun)

76. Memanggil function FatirBacaUrutNomer()

77. Selesai

**Program (Contoh 1)**

