

Pseudocode ServiceV2.cpp

ServiceApp

A. VALIDASI INPUT

FUNGSI input_IntegerWithRetry(num, pertanyaan)

 ULANGI

 Tampilkan pertanyaan

 Jika input bukan angka maka

 Bersihkan buffer

 Tampilkan pesan error

 Jika valid maka

 Kembalikan true

 SAMPAI valid

END FUNGSI

FUNGSI input_StringWithRetry(str, pertanyaan, allowEmpty)

 ULANGI

 Tampilkan pertanyaan

 Baca string

 Jika tidak boleh kosong dan string kosong maka

 Tampilkan error

 Jika valid maka

 Ubah ke huruf kecil

 Kembalikan true

 SAMPAI valid

END FUNGSI

FUNGSI input_JenisDeviceWithRetry(jenis, pertanyaan)

ULANGI

Input jenis

Jika jenis = "laptop" ATAU "hp" maka

Kembalikan true

Jika tidak maka

Tampilkan error

SAMPAI valid

END FUNGSI

B. OPERASI BST SERVICE

PROSEDUR create_BST(T)

T.root \leftarrow NIL

END PROSEDUR

PROSEDUR insert_BST(T, P)

Jika tree kosong maka

T.root \leftarrow P

SELESAI

current \leftarrow T.root

parent \leftarrow NIL

SELAMA current \neq NIL

parent \leftarrow current

Jika ID < current.ID maka

current \leftarrow current.left

Jika ID > current.ID maka

current \leftarrow current.right

Jika sama maka

Batalkan insert (ID duplikat)

AKHIRSELAMA

Sisipkan P sebagai anak kiri/kanan parent

END PROSEDUR

FUNGSI find_Service(root, id)

Jika root = NIL ATAU ID ditemukan maka

Kembalikan root

Jika $id < root.ID$ maka

Rekursif ke kiri

Jika tidak maka

Rekursif ke kanan

END FUNGSI

PROSEDUR delete_Service(T, id)

Cari node target

Jika node tidak ditemukan → SELESAI

Jika node leaf → hapus langsung

Jika node punya 1 anak → naikkan child

Jika node punya 2 anak →

Cari successor

Salin data successor

Hapus successor

END PROSEDUR

C. MULTI LINKED LIST & QUEUE TEKNISI

PROSEDUR insert_Teknisi(L, P)

Jika list kosong maka

L.first \leftarrow P

Jika tidak maka

Tambahkan di akhir list

END PROSEDUR

PROSEDUR insert_Antrian(T, A)

Jika antrian kosong maka

T.firstAntrian \leftarrow A

Jika tidak maka

Tambahkan A di akhir queue

END PROSEDUR

PROSEDUR delete_FirstAntrian(T)

Jika antrian kosong maka

Tampilkan error

Jika tidak maka

Hapus node pertama (FIFO)

END PROSEDUR

FUNGSI serviceAlready_Assigned(L, idService)

Telusuri semua teknisi

Telusuri semua antrian

Jika idService ditemukan \rightarrow true

Kembalikan false

END FUNGSI

D. LAPORAN & STATISTIK

FUNGSI count_TotalService(root)

Jika root = NIL \rightarrow 0

Kembalikan 1 + kiri + kanan

END FUNGSI

FUNGSI find_MaxKomponen(root)

Bandingkan root, kiri, kanan

Kembalikan node dengan komponen terbanyak

END FUNGSI

FUNGSI find_TeknisiPalingSedikit(L)

Cari teknisi dengan jumlah antrian minimum

END FUNGSI