

```

DECLARE STRUCT Data:
    nim : string
    nama : string
    status : array[5] of char
    hadir : integer
    izin : integer
    sakit : integer
    alpa : integer
END STRUCT

DECLARE mahasiswa[10] AS ARRAY OF Data
DECLARE jumlahMhs = 10

// =====
// Inisialisasi Data Mahasiswa
// =====
FOR i FROM 0 TO 9:
    INPUT mahasiswa[i].nim
    INPUT mahasiswa[i].nama
    SET semua status[i][0..4] = '\0'
    SET hadir = izin = sakit = alpa = 0
END FOR

// =====
// Fungsi Cari Mahasiswa
// =====
FUNCTION cariMahasiswa(nim):
    FOR i FROM 0 TO jumlahMhs - 1:
        IF mahasiswa[i].nim == nim:
            RETURN i
    END FOR
    RETURN -1
END FUNCTION

// =====
// Input Presensi Mahasiswa
// =====
PROCEDURE inputPresensi:

    INPUT nim
    SET idx = cariMahasiswa(nim)

    IF idx == -1:
        PRINT "NIM tidak ditemukan"
        RETURN
    END IF

    PRINT "Nama: " + mahasiswa[idx].nama

    SET hari = -1
    FOR i FROM 0 TO 4:
        IF mahasiswa[idx].status[i] == '\0':
            hari = i
            BREAK
    END FOR

    IF hari == -1:
        PRINT "Presensi penuh!"
        RETURN

```

```

END IF

PRINT "Mengisi presensi pertemuan ke-" + (hari + 1)
PRINT "Masukkan status (H/I/S/A): "
INPUT statusPresensi

UBAH statusPresensi ke huruf besar
mahasiswa[idx].status[hari] = statusPresensi

PRINT "Presensi berhasil disimpan"

END PROCEDURE

// =====
// Hitung Rekap Kehadiran
// =====
PROCEDURE hitungRekap(arrayMahasiswa, n):

    FOR i FROM 0 TO n-1:
        SET hadir = izin = sakit = alpa = 0

        FOR j FROM 0 TO 4:
            SWITCH arrayMahasiswa[i].status[j]:
                CASE 'H': hadir++
                CASE 'I': izin++
                CASE 'S': sakit++
                CASE 'A': alpa++
            END SWITCH
        END FOR

        arrayMahasiswa[i].hadir = hadir
        arrayMahasiswa[i].izin = izin
        arrayMahasiswa[i].sakit = sakit
        arrayMahasiswa[i].alpa = alpa

    END FOR

END PROCEDURE

// =====
// Sort Alpa Tertinggi
// =====
PROCEDURE bubbleSortAlpa(arrayMahasiswa, n):

    FOR i FROM 0 TO n - 2:
        FOR j FROM 0 TO n - i - 2:
            IF arrayMahasiswa[j].alpa < arrayMahasiswa[j+1].alpa:
                TUKAR arrayMahasiswa[j] dengan arrayMahasiswa[j+1]
            END IF
        END FOR
    END FOR

END PROCEDURE

// =====
// Tampilkan Rekap + Alpa Tertinggi
// =====
PROCEDURE tampilkanRekap(arrayMahasiswa, n):

```

```

CALL hitungRekap(arrayMahasiswa, n)

PRINT "-----"
PRINT "NIM   | Nama | H | I | S | A"
PRINT "-----"

FOR i FROM 0 TO n-1:
    PRINT mahasiswa[i].nim, nama, hadir, izin, sakit, alpa
END FOR

CALL bubbleSortAlpa(arrayMahasiswa, n)

PRINT "Mahasiswa dengan alpa terbanyak:"
PRINT arrayMahasiswa[0].nama
PRINT arrayMahasiswa[0].nim
PRINT arrayMahasiswa[0].alpa

END PROCEDURE

// =====
// Cari Mahasiswa
// =====
PROCEDURE cariData:

    INPUT nim
    SET idx = cariMahasiswa(nim)

    IF idx == -1:
        PRINT "Mahasiswa tidak ditemukan"
        RETURN
    END IF

    CALL hitungRekap(mahasiswa, jumlahMhs)

    PRINT "Nama   : " + mahasiswa[idx].nama
    PRINT "Hadir  : " + mahasiswa[idx].hadir
    PRINT "Izin   : " + mahasiswa[idx].izin
    PRINT "Sakit  : " + mahasiswa[idx].sakit
    PRINT "Alpa   : " + mahasiswa[idx].alpa

END PROCEDURE

// =====
// PROGRAM UTAMA
// =====
MAIN:

    REPEAT:
        PRINT "===== MENU PRESENSI MAHASISWA ====="
        PRINT "1. Input Presensi"
        PRINT "2. Tampilkan Rekap"
        PRINT "3. Cari Mahasiswa"
        PRINT "4. Keluar"
        INPUT pilih

        SWITCH pilih:
            CASE 1: CALL inputPresensi()
            CASE 2: CALL tampilkanRekap(mahasiswa, jumlahMhs)

```

```
        CASE 3: CALL cariData()
        CASE 4: PRINT "Program selesai."
        DEFAULT: PRINT "Pilihan tidak valid"
    END SWITCH

    UNTIL pilih == 4

END MAIN
```