Ahmad Sultani Dayanullah

D121211080

Tugas 7 ASD

BAB 14

No 1

Perbedaan pada deklarasi 2 buah matriks temperatur tekanan

A

DEKLARASI

const NBarisMaks = ...

const NKolomMaks = ...

type TP : record < T, P : real >

B : array [1..NBarisMaks, 1..NKolomMaks] of TP

B

DEKLARASI

const NBarisMaks = ...

const NKolomMaks = ...

type TP : record < T, P : array [1..NBarisMaks, 1..NKolomMaks] of real >

B : TP

Dapat dilihat bahwa:

A merupakan array dari record yang berisi field T dan P yang bertipe real

B merupakan record yang berisi field T dan P yang merupakan array dari bilangan real

Karenanya, cara mengakses elemen-elemen A dan B berbeda

Bentuk ekivalen dari A dan B untuk mengakses temperatur baris ke-i dan kolom ke-j adalah:

A. B[i, j].T {array 2 dimensi yang bertipe record dengan field real}

B. B.T[i, j] {record yang berisi field array T dan P yang bertipe real}

No 6

program nilai\_mahasiswa

{Menghitung nilai rata-rata dari setiap mahasiswa dengan jumlah mata kuliah sebanyak n}

DEKLARASI

M, n, jumSKS, JumNilai, nilaiKonversi, i, j : integer

NR : array [1..n] of real

data : record < indeksNilai : character,

SKS : integer >

NilaiMhs : array [1..M, 1..n] of data

ALGORITMA

A:

jumSKS = 0

jumNilai = 0

read(M, n)

for i <- 1 to M do

for j <- 1 to n do

read(NilaiMhs[i, j].indeksNilai)

nilai = NilaiMhs[i, j].indeksNilai

nilaiKonversi <- 0

case nilai of

'A' : nilaiKonversi <- 4

'B' : nilaiKonversi <- 3

'C' : nilaiKonversi <- 2

'D' : nilaiKonversi <- 1

'E' : nilaiKonversi <- 0

Otherwise :

write("Nilai tidak valid")

goto A:

read(NilaiMhs[i, j].SKS)

JumNilai <- JumNilai + (nilaiKonversi \* NilaiMhs[i, j].SKS)

jumSKS <- jumSKS + NilaiMhs[i, j].SKS

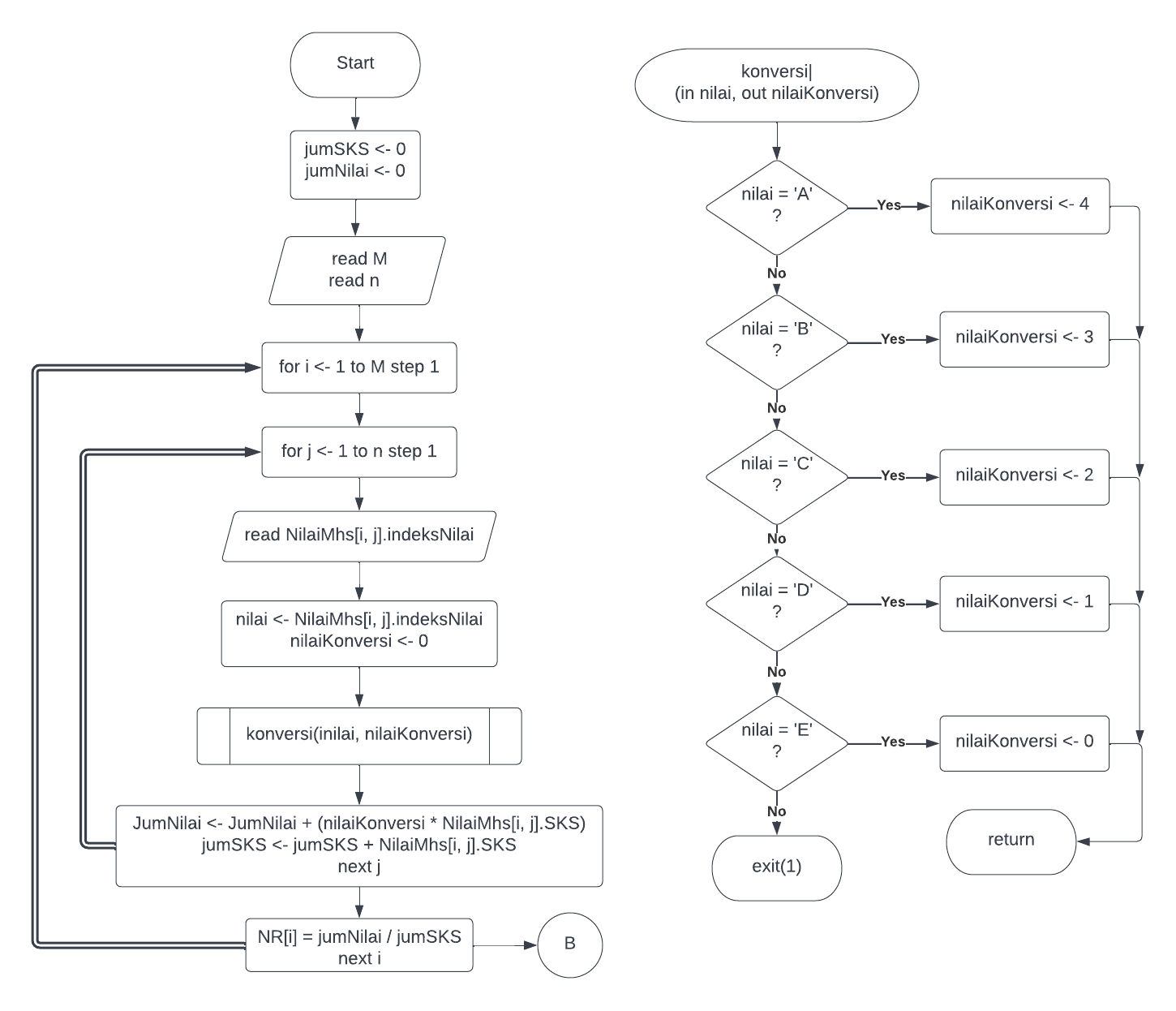
endfor

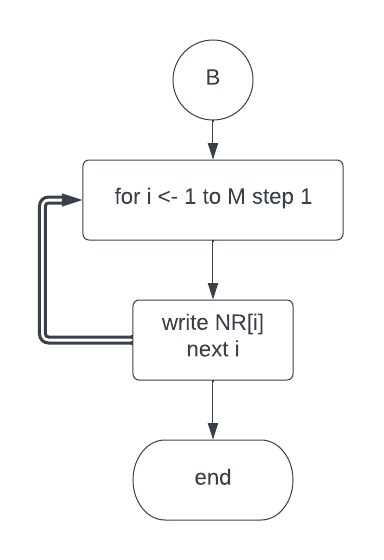
NR[i] <- JumNilai / jumSKS

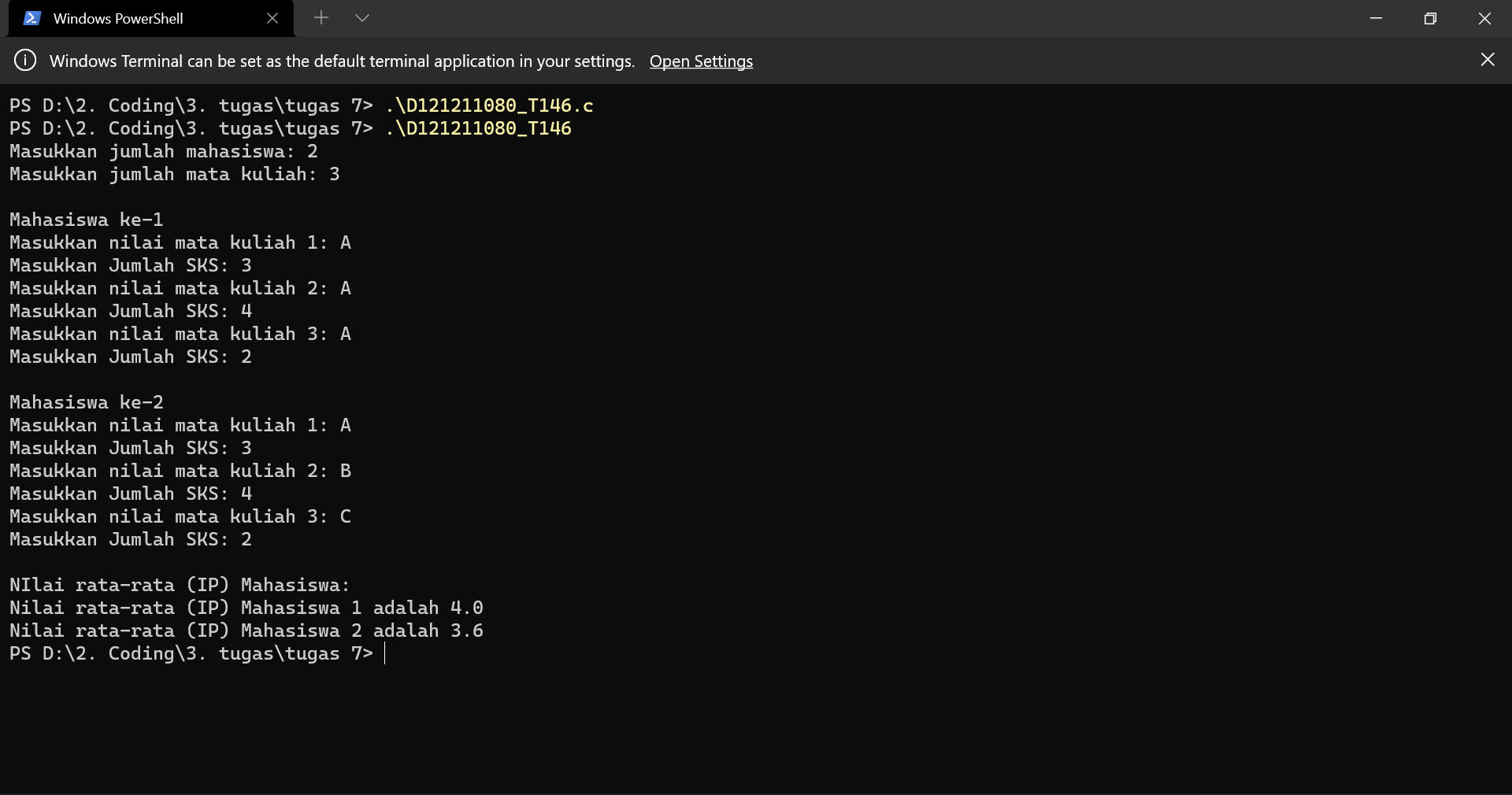
endfor

for i <- 1 to M do

write(NR[i])

endfor





BAB 15

No 1

program sequential\_reversed

{mencari nilai x dalam larik dengan pencarian beruntun dari elemen terakhir}

DEKLARASI

x, i, j, N : integer

arr : array [1..N] of integer

ALGORITMA

read(N)

for i <- 1 to N do

read(arr[i])

endfor

j <- N

while j >= 1 do

if arr[j] = x then

write(x, "ditemukan pada indeks ke-", i)

return 0

endif

j <- j - 1

endfor

write(x, "tidak ditemukan dalam array")

No 5

function pencarian\_interpolasi(input: arr : LarikInt, N, target : integer)

{salah satu variasi dari pencarian bagidua}

DEKLARASI

left, right, mid : integer

ALGORITMA

left <- 1

right <- N

while (left != right) or (arr[left] != arr[right]) do

mid <- left + (right-left)/(arr[right]-arr[left]) \* (target-arr[left])

if arr[mid] = target then

return mid

else

if arr[mid] < target then

left <- mid + 1

else

right <- mid - 1

endif

endif

endwhile

return -1

program pencarian

{mencari suatu bilangan dalam larik bilangan}

DEKLARASI

N, target, index, i : integer

type LarikInt : array [1..N] of integer

arr : LarikInt

function pencarian\_interpolasi(input: arr : LarikInt, N, target : integer)

{salah satu variasi dari pencarian bagidua}

ALGORITMA

read(N)

for i<-1 to N do

read(arr[i])

endfor

read(target)

index = pencarian\_interpolasi(arr, N, target)

if index = -1 then

write("Bilangan tidak ditemukan")

else

write("Bilangan ditemukan pada indeks ke ", index)

endif

