

Nama : Mumtaz Hana Najda Hafidh
Nim : 24060120130107
Kelas : A

Nomor 1

①

Function IsPolindrome (L : List) \rightarrow boolean

{Mengirimkan nilai TRUE jika semua elemen list membentuk palindrome}

{List kosong, dan list 1 elemen TRUE}

{L : 2 TRUE, L : 2 \rightarrow 3 \rightarrow 2 TRUE, L : 2 \rightarrow 3 \rightarrow 3 \rightarrow 2 TRUE, L : 2 \rightarrow 3 \rightarrow 6 \rightarrow 3 \rightarrow 2 TRUE}

{L : 2 \rightarrow 3 \rightarrow 1 \rightarrow 3 \rightarrow 1 \rightarrow 2 FALSE}

{Kamus Lokal}

P, Q : address

Lt : List {list sementara}

cek : integer {mengecek jumlah elemen yang sama untuk mengetahui list palindrome atau tidak}

{Algoritma}

CreateList (Lt)

CopyList (L, Lt)

InversList (Lt)

P \leftarrow First(L)

Q \leftarrow First (Lt)

cek \leftarrow 0

while (P \neq Nil and Q \neq Nil) do

if (info(P) = info(Q)) then

cek \leftarrow cek + 1

P \leftarrow next (P)

Q \leftarrow next (Q)

if (cek = Nbelmt (L)) then

\rightarrow true

else

\rightarrow false

Nomor 2

(2)

Program SumLargeNumber

↓ Diketahui L_1 dan L_2 merupakan 2 buah bilangan integer large number

↓ Buatlah algoritma program utama untuk menjumlahkan L_1 dan L_2 yang

↓ hasilnya disimpan pada list baru L

↓ Example $L_1: 2 \rightarrow 3 \rightarrow 1 \rightarrow 3 \rightarrow 1 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 9 \rightarrow 9$, $L_2: 7 \rightarrow 3 \rightarrow 1 \rightarrow 3 \rightarrow 9 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 9 \rightarrow 9$

↓ maka $L_1 + L_2 = L: 9 \rightarrow 6 \rightarrow 2 \rightarrow 7 \rightarrow 0 \rightarrow 5 \rightarrow 3 \rightarrow 7 \rightarrow 9 \rightarrow 8$

↓ Kamus

L, L_1, L_2 : List

A, B : address

$carry$: integer {menyimpan hasil lebih dari penjumlahan, seperti $8+2=10$,
carry adalah 1}

sum : integer {hasil penjumlahan}

↓ Algoritma

CreateList (L_1)

CreateList (L_2)

CreateList (L)

InversList (L_1) {dibalik karena pengerjaan dari belakang}

InversList (L_2)

$A \leftarrow \text{first}(L_1)$

$B \leftarrow \text{first}(L_2)$

$carry \leftarrow 0$

while ($A \neq \text{Nil}$ and $B \neq \text{Nil}$) do

$sum \leftarrow (\text{info}(A) + \text{info}(B) + carry) \bmod 10$

$carry \leftarrow (\text{info}(A) + \text{info}(B) + carry) \div 10$

InsVLast (L, sum)

$A \leftarrow \text{next}(A)$

$B \leftarrow \text{next}(B)$

If ($carry \neq 0$) then

InsVLast ($L, carry$)

InversList (L) {mengembalikan list ke bentuk semula}

```
printf("\n---SUMLARGENUMBER---\n");
/* Menjumlahkan L1 dan L2 yang hasilnya disimpan pada L*/
/* Isi L1 */
CreateList(&L1);
InsVLast(&L1, 2);
InsVLast(&L1, 3);
InsVLast(&L1, 1);
InsVLast(&L1, 3);
InsVLast(&L1, 1);
InsVLast(&L1, 2);
InsVLast(&L1, 6);
```

```

InsVLast(&L1, 8);
InsVLast(&L1, 9);
InsVLast(&L1, 9);
printf("Isi L1 = ");
PrintInfo(L1);

/* Isi L2 */
CreateList(&L2);
InsVLast(&L2, 7);
InsVLast(&L2, 3);
InsVLast(&L2, 1);
InsVLast(&L2, 3);
InsVLast(&L2, 9);
InsVLast(&L2, 9);
InsVLast(&L2, 2);
InsVLast(&L2, 6);
InsVLast(&L2, 8);
InsVLast(&L2, 9);
InsVLast(&L2, 9);
printf("\nIsi L2 = ");
PrintInfo(L2);

/* Program SumLargeNumber */
CreateList(&L);
InversList(&L1);
InversList(&L2);
A = First(L1);
B = First(L2);
carry = 0;

while(A!=Nil && B!=Nil){
    sum = (info(A)+info(B)+carry)%10;
    carry = (info(A)+info(B)+carry)/10;
    InsVLast(&L, sum);
    A = next(A);
    B = next(B);
}
if(carry!=0){
    InsVLast(&L, carry);
}
InversList(&L);
printf("\nL = ");
PrintInfo(L);

```

---SUMLARGENUMBER---

Isi L1 = 2 | 3 | 1 | 3 | 1 | 2 | 6 | 8 | 9 | 9 |

Isi L2 = 7 | 3 | 1 | 3 | 9 | 2 | 6 | 8 | 9 | 9 |

L = 9 | 6 | 2 | 7 | 0 | 5 | 3 | 7 | 9 | 8 |

Process returned 0 (0x0) execution time : 0.412 s

Press any key to continue.

Nomor 3

③

Program MulLargeNumber

{Diketahui L₁ dan L₂ merupakan 2 buah bilangan integer large number}

{Buatlah algoritma program utama untuk perkalian L₁ dan L₂ yang}

{hasilnya disimpan pada list baru L}

{Example L₁ : 2 → 3 → 1 → 3 → 1 → 2 → 6 → 8 → 9 → 9, L₂ : 7 → 3 → 1 → 3 → 9 → 2 → 6 → 8 → 9 → 9}

{maka L₁ * L₂ = L : 1 → 6 → 9 → 1 → 8 → 0 → 4 → 1 → 0 → 4 → 7 → 3 → 9 → 6 → 0 → 0 → 0 → 0}

{Kamus}

L, L₁, L₂ : list

A, B : address

num1, num2, mul : integer

{Algoritma}

CreateList (L₁)

CreateList (L₂)

CreateList (L)

A ← First (L₁)

B ← First (L₂)

num1 ← 0

num2 ← 0

While (A ≠ Nil and B ≠ Nil) do

num1 ← (num1 * 10) + info(A)

A ← next(A)

num2 ← (num2 * 10) + info(B)

B ← next(B)

mul ← num1 * num2

While (mul ≠ 0) do

InsVLast (L, (mul mod 10))

mul ← mul div 10

InversList (L)

```
printf("\n\n---MULLARGENUMBER---\n");
/* Perkalian L1 dan L2 yang hasilnya disimpan pada L */
/* Isi L1 */
CreateList(&L1);
InsVLast(&L1, 2);
InsVLast(&L1, 3);
InsVLast(&L1, 1);
InsVLast(&L1, 3);
InsVLast(&L1, 1);
InsVLast(&L1, 2);
InsVLast(&L1, 6);
InsVLast(&L1, 8);
InsVLast(&L1, 9);
InsVLast(&L1, 9);
printf("Isi L1 = ");
```

```

PrintInfo(L1);

/* Isi L2 */
CreateList(&L2);
InsVLast(&L2, 7);
InsVLast(&L2, 3);
InsVLast(&L2, 1);
InsVLast(&L2, 3);
InsVLast(&L2, 9);
InsVLast(&L2, 2);
InsVLast(&L2, 6);
InsVLast(&L2, 8);
InsVLast(&L2, 9);
InsVLast(&L2, 9);
printf("\nIsi L2 = ");
PrintInfo(L2);

CreateList(&L);
A = First(L1);
B = First(L2);
num1=0;
num2=0;
while(A!=Nil && B!=Nil){
    num1 = (num1*10)+info(A);
    A = next(A);
    num2 = (num2*10)+info(B);
    B = next(B);
}
printf("\nIsi num1 = %llu\n", num1);
printf("Isi num2 = %llu\n", num2);

mul = num1*num2;
printf("Hasil perkalian = %llu\n", mul);

while(mul!=0){
    InsVLast(&L, (mul%10));
    mul = mul/10;
}

InversList(&L);
printf("L = ");
PrintInfo(L);

```

---MULLARGENUMBER---

Isi L1 = 2 | 3 | 1 | 3 | 1 | 2 | 6 | 8 | 9 | 9 |

Isi L2 = 7 | 3 | 1 | 3 | 9 | 2 | 6 | 8 | 9 | 9 |

Isi num1 = 2313126899

Isi num2 = 7313926899

Hasil perkalian = 16918041047396556201

L = 1 | 6 | 9 | 1 | 8 | 0 | 4 | 1 | 0 | 4 | 7 | 3 | 9 | 6 | 5 | 5 | 6 | 2 | 0 | 1 |

Process returned 0 (0x0) execution time : 0.090 s

Press any key to continue.