

Praktikum 8

Double Linked List (2)

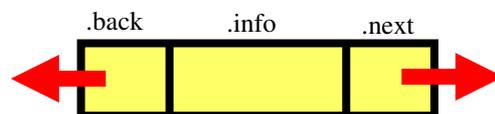
A. TUJUAN PEMBELAJARAN

Setelah mempelajari materi dalam bab ini, mahasiswa diharapkan mampu:

1. Memahami konsep operasi menyisipkan sebelum simpul tertentu
2. Memahami konsep operasi menyisipkan setelah simpul tertentu
3. Mengimplementasikan semua operasi single linked list dalam pemrograman
4. Mengidentifikasi permasalahan-permasalahan pemrograman yang harus diselesaikan dengan menggunakan double linked list dan menyelesaikannya.

B. DASAR TEORI

Double linked list dibentuk dengan menyusun sejumlah elemen sehingga pointer *next* menunjuk ke elemen yang mengikutinya dan pointer *back* menunjuk ke elemen yang mendahuluinya. Dalam Gambar 8.1 ini diilustrasikan sebuah simpul dalam *double linked list*. Info adalah data yang digunakan dalam simpul, back adalah pointer yang menunjuk pada simpul sebelumnya, dan next adalah pointer yang menunjuk pada simpul sesudahnya



Gambar 8.1 Ilustrasi sebuah simpul dalam *Double linked list*

B.1 Operasi Pada Linked List

Terdapat beberapa Operasi yang penting pada double linked list, yaitu:

1. Menyisipkan sebagai simpul ujung(awal) dari linked list.
2. Membaca atau menampilkan
3. Mencari sebuah simpul tertentu
4. Menghapus simpul tertentu (simpul depan)
5. Menghapus simpul tertentu (simpul di tengah)
6. Menghapus simpul tertentu (simpul terakhir)
7. Menyisipkan sebelum simpul tertentu
8. Menyisipkan setelah simpul tertentu

B.1.1 Menyisipkan Sebagai Sebelum Simpul Tertentu

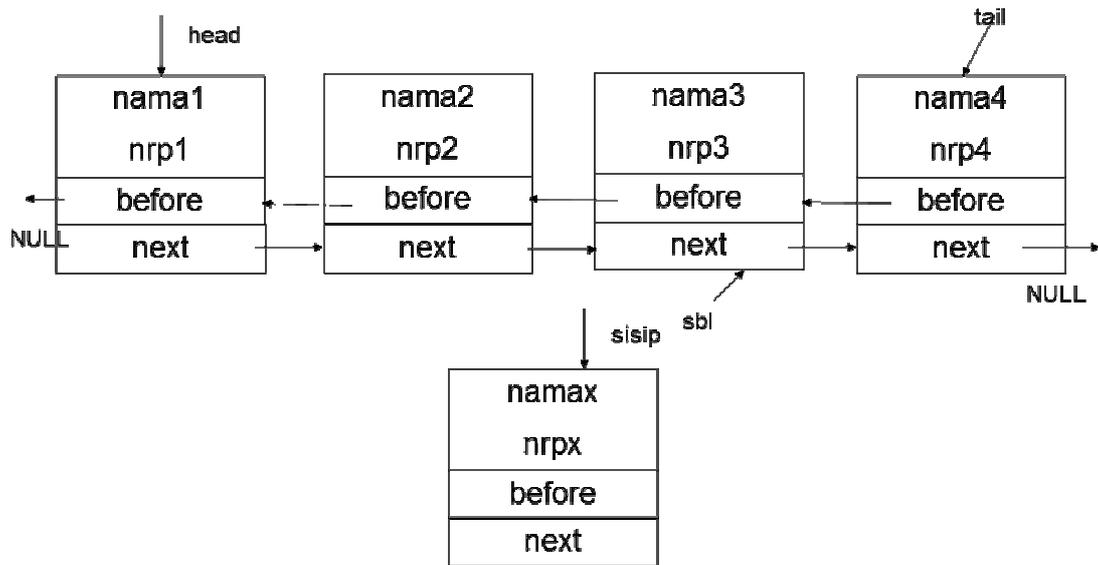
Langkah-langkah untuk menyisipkan simpul sisip sebelum simpul tertentu pada double linked list yang sudah terbentuk di atas adalah sebagai berikut:

1. Alokasikan memori untuk simpul sisip yang akan disisipkan
2. Inisialisasi sebuah variabel bertipe `struct simpul* (sbl)` dengan `head`
3. Lakukan proses pengulangan pencarian sampai `sbl->nama` sama dengan nama yang dicari
4. Arahkan `sisip->next` ke `sbl`
5. Arahkan `sisip->before` ke `sbl->before`
6. Arahkan `sbl->before->next` ke `sisip`
7. Arahkan `sbl->before` ke `sisip`

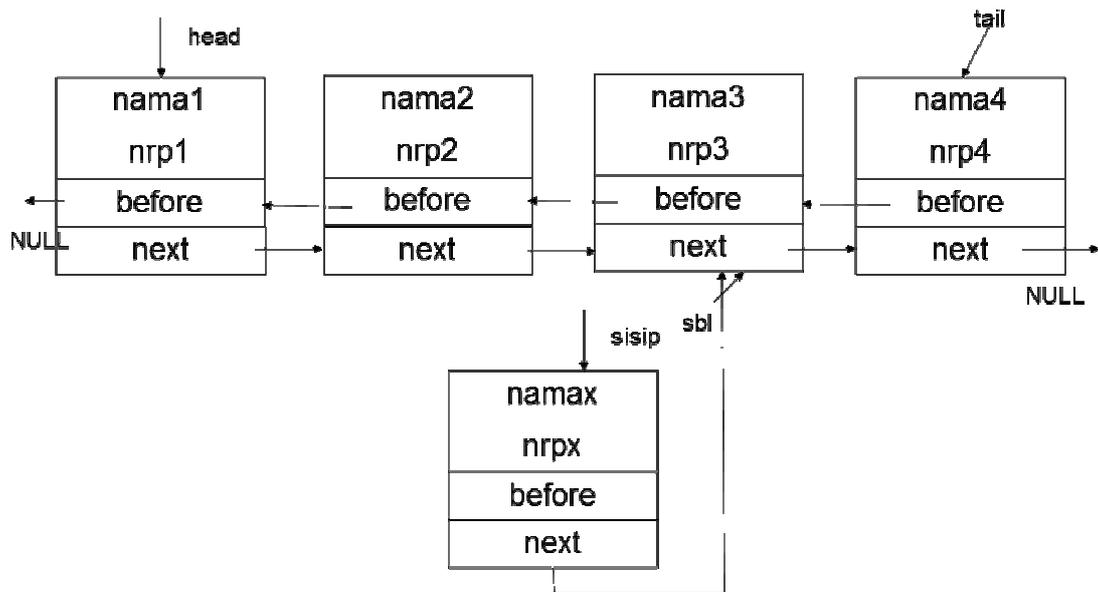
Berikut ini adalah perintah untuk menyisipkan data baru sebagai simpul terakhir pada single linked list

```
1. sisip = alokasi_simpul ();
2. sbl=head;
3. while (sbl->nama!=nama3)
4.     sbl=sbl->next;
5. sisip->next=sbl;
6. sisip->before=sbl->before;
7. sbl->before->next=sisip;
8. sbl->before=sisip;
```

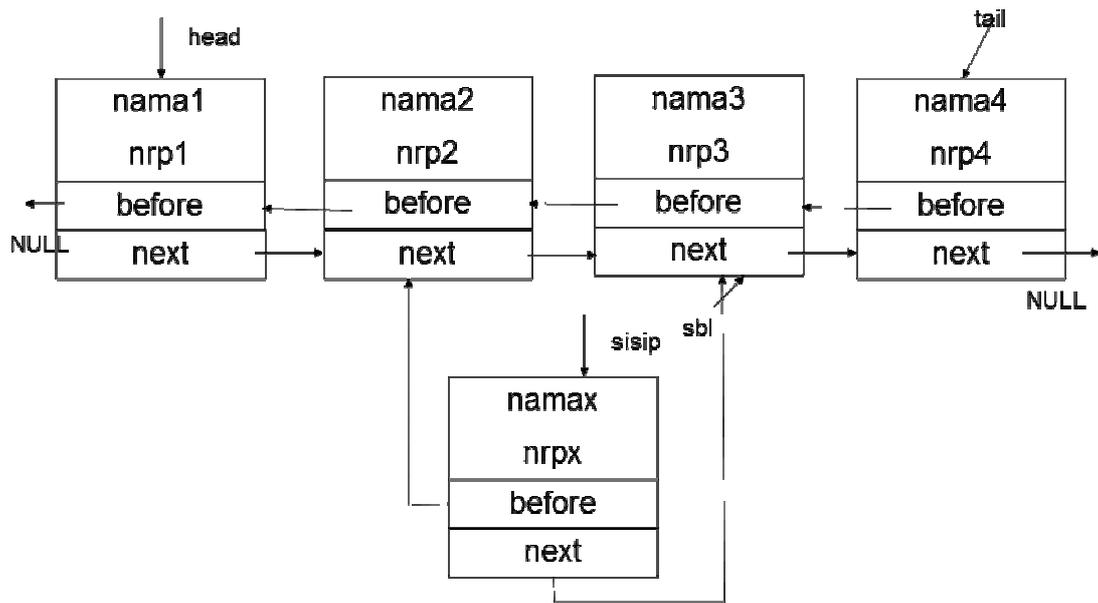
Setelah perintah baris ke-4



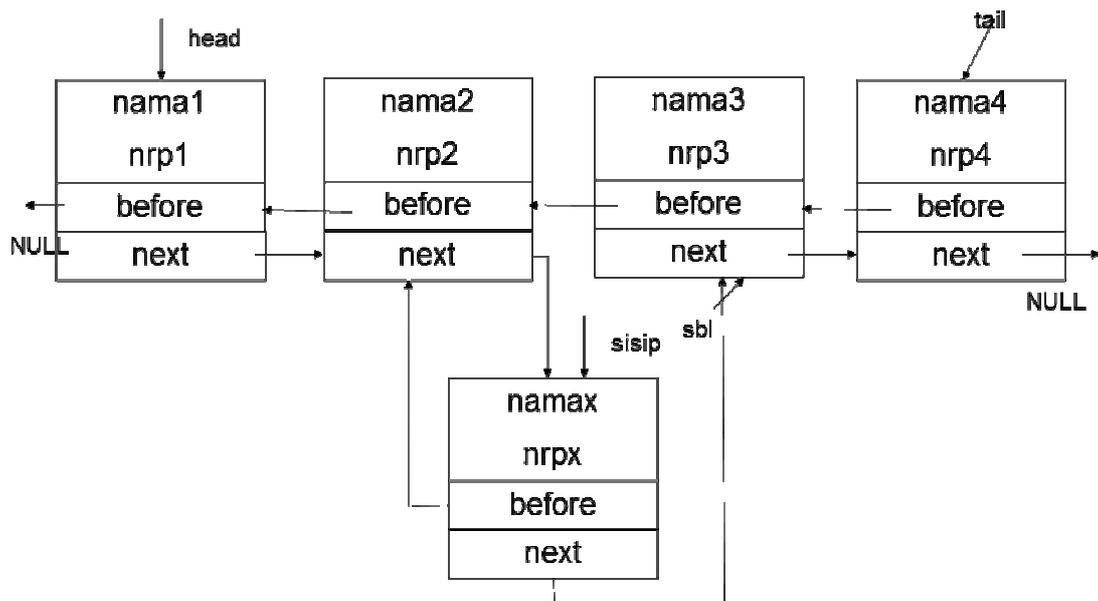
Setelah perintah `sisip->next=sbl`



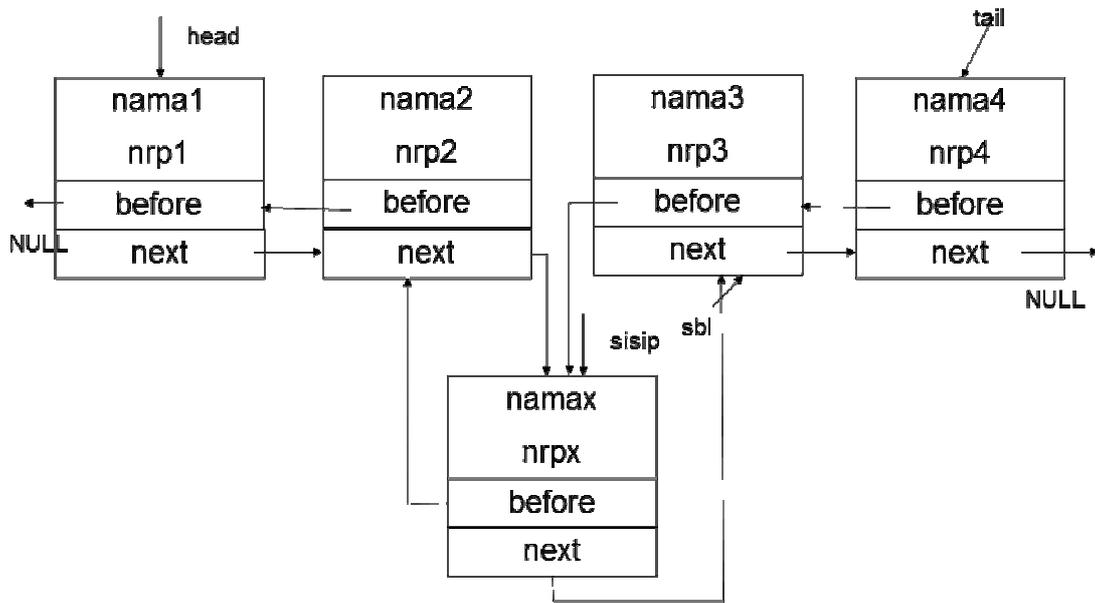
Setelah perintah `sisip->before=sbl->before`



Setelah perintah `sbl->before->next=sisip`



Setelah perintah `sbl->before=sisip`



B.1.2 Menyisipkan Setelah Simpul Tertentu

Langkah-langkah untuk menyisipkan simpul baru sebagai simpul terakhir pada linked list yang sudah terbentuk di atas adalah sebagai berikut:

1. Alokasikan memori untuk simpul sisip yang akan disisipkan
2. Inisialisasi sebuah variabel bertipe `struct simpul* (sbl)` dengan `head`
3. Lakukan proses pengulangan pencarian sampai `sbl->nama` sama dengan nama yang dicari
4. Arahkan `sisip->next` ke `sbl`
5. Arahkan `sisip->before` ke `sbl->before`
6. Arahkan `sbl->before->next` ke `sisip`
7. Arahkan `sbl->before` ke `sisip`

Berikut ini adalah perintah untuk menyisipkan data baru sebagai simpul terakhir pada single linked list

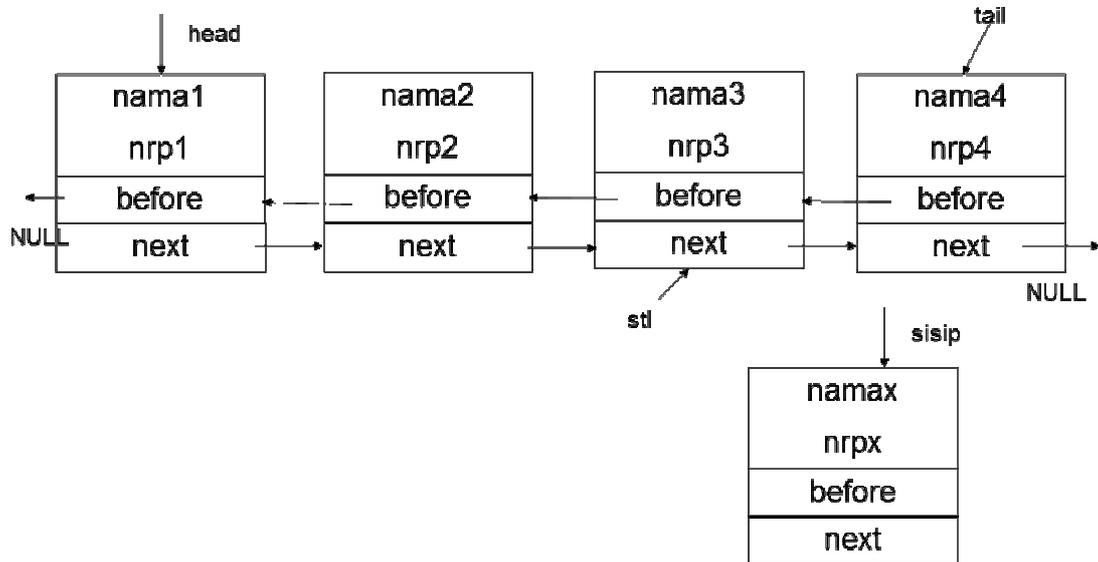
```

1.  sisip = alokasi_simpul ();
2.  stl=head;
3.  while(stl->nama!=nama3)
4.      stl=stl->next;
5.  sisip->before=stl;
6.  sisip->next=stl->next;
7.  stl->next->before=sisip;

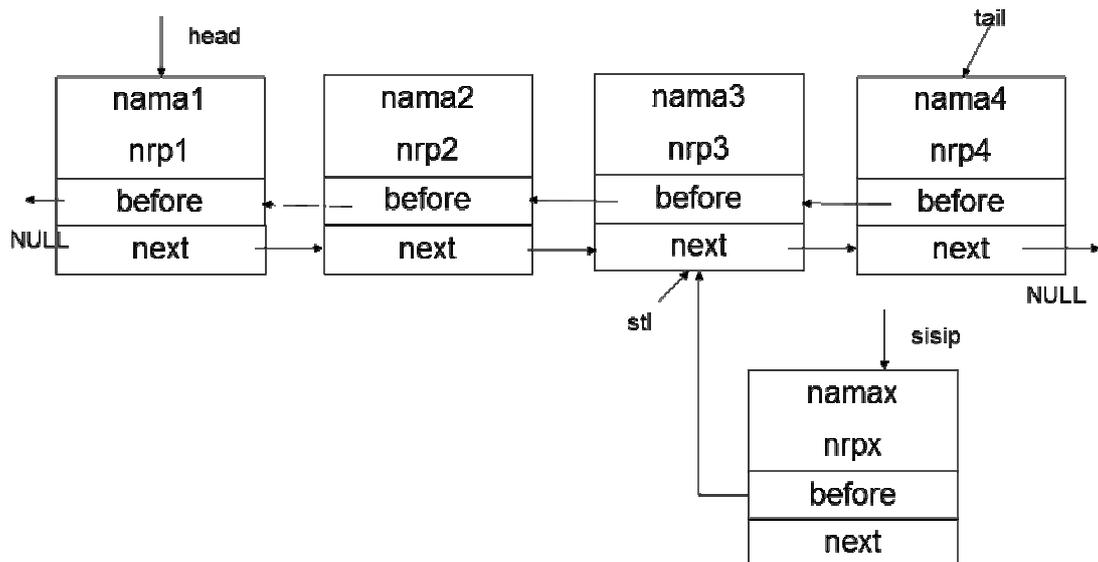
```

8. `stl->next=sisip;`

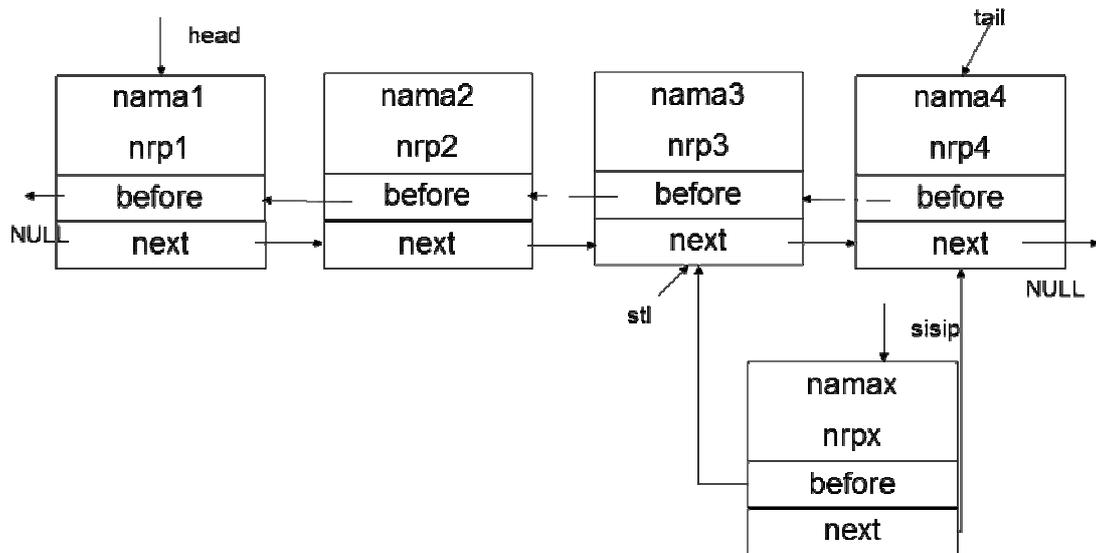
Setelah perintah baris ke-4



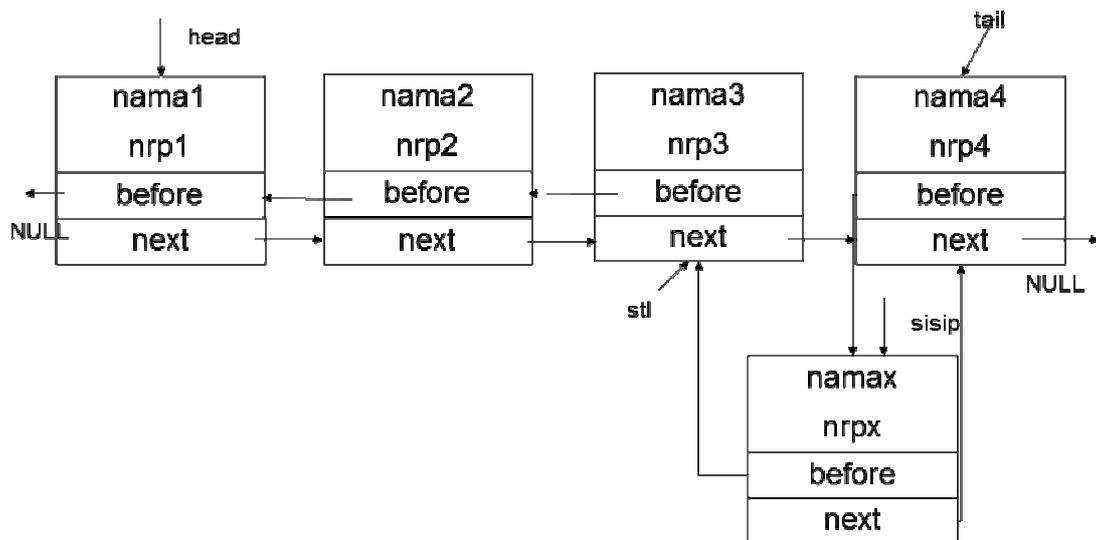
Setelah perintah `sisip->before=stl`



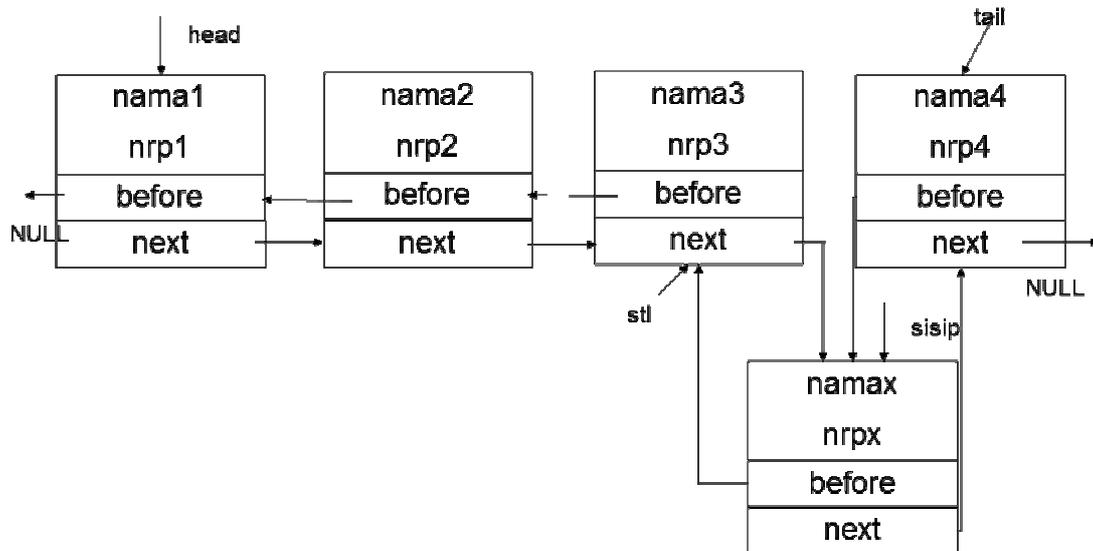
Setelah perintah `sisip->next=stl->next`



Setelah perintah `stl->next->before=sisip`



Setelah perintah `stl->next=sisip`



C. TUGAS PENDAHULUAN

Untuk semua operasi dasar single linked list persoalan di bawah ini, desainlah algoritma dan flowchartnya :

1. Menyisipkan sebelum simpul tertentu
2. Menyisipkan setelah simpul tertentu

D. PERCOBAAN

1. Implementasikan operasi dasar Double Linked List : Menyisipkan sebelum simpul tertentu. Tambahkan kondisi jika data yang disisipkan sebelumnya adalah data pertama.
2. Implementasikan operasi dasar Double Linked List : Menyisipkan setelah simpul tertentu. Tambahkan kondisi jika data yang disisipkan setelahnya adalah data terakhir.
3. Gabungkan semua operasi di atas dalam sebuah Menu Pilihan.

E. LATIHAN

Merepresentasikan sebuah bilangan polinomial dengan double linked list. Masalah aritmatika polinom adalah membuat sekumpulan subrutin manipulasi terhadap polinom simbolis (symbolic Polynomial).

Misalnya: $P1 = 6x^8 + 8x^7 + 5x^5 + x^3 + 15$

$$P2 = 3x^9 + 4x^7 + 3x^4 + 2x^3 + 2x^2 + 10$$

Representasikan bilangan polinom dengan menggunakan linked list dan buatlah prosedur-prosedur untuk :

- Menyisipkan simpul di awal jika pangkat yang dimasukkan lebih dari pangkat tertinggi dari bilangan polinomial.
- Menyisipkan simpul di tengah jika pangkat dari bilangan yang kita sisipkan berada di tengah.
- Menyisipkan simpul di akhir jika pangkat dari bilangan yang disisipkan adalah 0.
- Menghapus simpul, baik di awal, di tengah, ataupun di akhir.

F. LAPORAN RESMI

1. Kerjakan hasil percobaan(D) dan latihan(E) di atas dan tambahkan analisa.
2. Tuliskan kesimpulan dari percobaan dan latihan yang telah anda lakukan.