

**PERANCANGAN PERANGKAT LUNAK ANALISIS
TERHADAP REAKSI-REAKSI KIMIA
PADA SENYAWA HIDROKARBON
GOLONGAN ALIFATIK JENUH**

Oleh :

**MUHAMMAD IKHSAN, ST., M.KOM
NIP . 19830415 201101 1 008**



PERPUSTAKAAN
UIN - SU
MEDAN

TGL. TERIMA: 23-01-2017
NO. INDUK: 08/LP/EST/01/17
ASAL: Sumbangan
Dosen UIN-SU

KONSULTAN

**MHD. FURQAN, S.Si, M.Com,Sc
NIP. 19800806 200604 1 003**

p
347
IKH
p
c.1

**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI
SUMATERA UTARA
MEDAN
2017**

Laporan Penelitian

**PERANCANGAN PERANGKAT LUNAK ANALISIS
TERHADAP REAKSI-REAKSI KIMIA
PADA SENYAWA HIDROKARBON
GOLONGAN ALIFATIK JENUH**

Oleh :

**MUHAMMAD IKHSAN, ST., M.KOM
NIP . 19830415 201101 1 008**



**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI
SUMATERA UTARA
MEDAN
2017**

ABSTRAK

Kehidupan manusia sebagai salah satu organisme bisa dikatakan tidak bisa terlepas dari pemakaian senyawa-senyawa organik pada kehidupannya sehari-hari. Baik kita sadari ataupun tidak, bahkan di dalam tubuh kita sendiri terjadi ratusan reaksi dari senyawa organik.

Karena dari segi kuantitas jumlah senyawa organik yang begitu banyak, juga karena sifat serta klasifikasinya yang khas dan peranannya yang sangat penting dalam kehidupan, maka senyawa karbon yang disebut senyawa organik itu terus diusahakan untuk dibuat dan ditiru dilaboratorium. Namun banyak aspek teknis yang dinilai cukup memberatkan, baik dari segi biaya, waktu serta tenaga.

Tujuan akhir dari pembahasan ini adalah untuk merancang suatu sistem perangkat lunak yang dapat menentukan tatanama senyawa hidrokarbon serta menganalisa dan men-simulasikan kemungkinan hasil reaksi pembakaran sembarang senyawa hidrokarbon sehingga dapat ditentukan rumus empirik dan rumus molekul senyawa tersebut berdasarkan sifat serta karakteristik yang dimilikinya sebelum dilakukan pengujian lebih lanjut di laboratorium.

BAB I

PENDAHULUAN

I. 1 Latar Belakang

Salah satu cabang ilmu kimia adalah Kimia Organik yang secara khusus mempelajari senyawa karbon yang juga disebut sebagai senyawa organik, istilah senyawa organik digunakan karena pada mulanya manusia berpendapat bahwa senyawa-senyawa tersebut hanya dapat dihasilkan makhluk hidup dan tidak dapat ditiru (dibuat) di laboratorium hal ini sesuai dengan paham *vitalisme* yang menyatakan bahwa makhluk hidup mempunyai daya hidup (*vis-vitalis*) yang ketika itu dianut. Namun selaras dengan perkembangan ilmu pengetahuan, manusia telah mampu membuat beberapa senyawa organik seperti karet, vitamin C, alkohol, asam cuka dan lain-lain.

Banyak juga senyawa yang berdasarkan sifat-sifatnya dimasukkan ke dalam senyawa organik walaupun tidak pernah dihasilkan oleh makhluk hidup, telah berhasil dibuat sebagai contoh plastik dan nilon. Walaupun senyawa-senyawa organik itu telah diyakini tidak harus berasal dari organisme, namun sumber utama senyawa-senyawa tersebut adalah makhluk hidup (tumbuhan dan hewan) atau sisa-sisa makhluk hidup seperti batu bara, minyak bumi dan gas alam.

Kehidupan manusia sebagai salah satu organisme bisa dikatakan tidak bisa terlepas dari pemakaian senyawa-senyawa organik pada kehidupannya sehari-hari. Baik kita sadari ataupun tidak, bahkan di dalam tubuh kita sendiri terjadi ratusan reaksi dari senyawa organik. Para Ahli Kimia berusaha keras untuk terus menciptakan atau meniru suatu senyawa organik baru dengan melakukan reaksi dari senyawa-senyawa organik

yang ada (dikenal) baik melalui reaksi *substitusi*, *adisi*, *cracking* dan lain-lain sehingga diharapkan dapat menghasilkan senyawa-senyawa organik baru yang bermanfaat bagi kehidupan manusia.

Perkembangan dunia informasi dan teknologi pada saat ini berkembang semakin cepat dalam berbagai bidang. Salah satu perkembangan yang penting adalah di bidang komputer. Komputer memegang peranan dalam segala hal, terutama yang menginginkan segala pekerjaan diselesaikan dengan cara cepat, tepat dan hemat biaya. Maka dari itu penulis tertarik ingin membahas sistem analisis kemungkinan hasil reaksi pembakaran dari senyawa hidrokarbon, koefisiennya dan aturan penamaannya menurut IUPAC. Sistem ini kelak akan dapat digunakan sebagai pembanding hasil analisa di laboratorium secara manual dengan sistem secara komputerisasi sehingga beberapa aspek teknis seperti waktu dan biaya dapat diminimalkan.

I. 2 Identifikasi Masalah

Karena dari segi kuantitas jumlah senyawa organik yang begitu banyak, juga karena sifat serta klasifikasinya yang khas dan peranannya yang sangat penting dalam kehidupan, maka senyawa karbon yang disebut senyawa organik itu terus diusahakan untuk dibuat dan ditiru dilaboratorium. Namun banyak aspek teknis yang dinilai cukup memberatkan, salah satunya dari segi biaya. Maka penulis merasa perlu adanya suatu perangkat lunak yang dapat menganalisa dan men-simulasikan kemungkinan hasil reaksi dari senyawa hidrokarbon, koefisiennya dan aturan penamaannya menurut IUPAC (*Internation Union of Pure and Applied Chemistry*) sebelum dilakukan pengujian lebih lanjut di laboratorium.

I. 3 Tujuan Pembahasan

Tujuan akhir dari pembahasan ini adalah untuk merancang suatu sistem perangkat lunak yang dapat menganalisa dan men-simulasikan kemungkinan hasil reaksi dari senyawa hidrokarbon berdasarkan sifat serta karakteristik yang dimiliki tiap senyawa tersebut sebelum dilakukan pengujian lebih lanjut di laboratorium, hal ini sangat efisien baik dari segi biaya, waktu serta tenaga.

I. 4 Batasan Masalah

Agar pembahasan tidak menyimpang dari tujuan diatas, maka perlu dibuat suatu batasan masalah yaitu :

1. Senyawa Hidrokarbon yang dianalisa masih mencakup senyawa hidrokarbon golongan alifatik jenuh.
2. Aturan penamaan tiap senyawa hidrokarbon memakai aturan yang telah ditetapkan oleh IUPAC (*International Union of Pure and Applied Chemistry*)
3. Sistem Perangkat Lunak yang akan dirancang berjalan diatas sistem operasi Windows dan dibangun dengan Development Tools Visual Basic 6.0.

I. 5 Manfaat Penulisan

Adapun manfaat dari Penulisan ini adalah :

1. Bagi Penulis, merupakan aspek analisa dan implementasi dari pelbagai disiplin ilmu yang telah diperoleh ke bentuk perancangan dan pembuatan suatu sistem perangkat lunak yang kompleks.

I. 3 Tujuan Pembahasan

Tujuan akhir dari pembahasan ini adalah untuk merancang suatu sistem perangkat lunak yang dapat menganalisa dan men-simulasikan kemungkinan hasil reaksi dari senyawa hidrokarbon berdasarkan sifat serta karakteristik yang dimiliki tiap senyawa tersebut sebelum dilakukan pengujian lebih lanjut di laboratorium, hal ini sangat efisien baik dari segi biaya, waktu serta tenaga.

I. 4 Batasan Masalah

Agar pembahasan tidak menyimpang dari tujuan diatas, maka perlu dibuat suatu batasan masalah yaitu :

1. Senyawa Hidrokarbon yang dianalisa masih mencakup senyawa hidrokarbon golongan alifatik jenuh.
2. Aturan penamaan tiap senyawa hidrokarbon memakai aturan yang telah ditetapkan oleh IUPAC (*International Union of Pure and Applied Chemistry*)
3. Sistem Perangkat Lunak yang akan dirancang berjalan diatas sistem operasi Windows dan dibangun dengan Development Tools Visual Basic 6.0.

I. 5 Manfaat Penulisan

Adapun manfaat dari Penulisan ini adalah :

1. Bagi Penulis, merupakan aspek analisa dan implementasi dari pelbagai disiplin ilmu yang telah diperoleh ke bentuk perancangan dan pembuatan suatu sistem perangkat lunak yang kompleks.

2. Bagi Dunia Akademis, merupakan konsep dasar untuk perancangan perangkat lunak yang memiliki daya guna yang tinggi dan bersifat komersial jika dikembangkan lebih lanjut.
3. Pengembangan program lebih lanjut memiliki potensi yang sangat baik untuk memajukan ilmu pengetahuan khususnya pada bidang Kimia Organik.

I. 6 Metoda Penelitian

Dalam penyusunan Laporan penelitian ini penulis menggunakan beberapa metoda penelitian yaitu :

1. Metoda Pengumpulan Data yaitu mengumpulkan berbagai sumber data baik dari literatur, artikel, jurnal, situs dan lain-lain yang mendukung dan berhubungan dalam penyusunan Laporan Penelitian ini.
2. Metoda Observasi yaitu pengamatan langsung pada objek penelitian seperti karakteristik yang bersifat fisikis dan spesifik dari senyawa hidrokarbon golongan alifatik

I. 7 Sistematika Penulisan

Secara garis besar sistematika penulisan dalam pembuatan Laporan penelitian ini adalah sebagai berikut :

BAB I : PENDAHULUAN

Pembahasan dalam bagian ini berkisar pada latar belakang masalah, rumusan perancangan yang dibangun dalam sistem, tujuan penelitian dan sistematika penulisan.

BAB II : TINJAUAN TEORITIS

Pada bab ini berisi uraian tentang teori-teori yang melandasi penulisan dan perancangan perangkat lunak, seperti sifat serta karakteristik dari senyawa hidrokarbon golongan alifatik jenuh dan tak jenuh. Juga aturan penamaan yang telah ditetapkan oleh IUPAC.

BAB III : PERANCANGAN SISTEM

Berisi tahapan-tahapan perancangan yang meliputi perancangan layout perangkat lunak, desain *interface*, perancangan prosedur dan fungsi serta modul-modul program yang akan membangun keseluruhan perangkat lunak tersebut.

BAB IV : IMPLEMENTASI PROGRAM

Bab ini membahas mengenai algoritma-algoritma yang digunakan dalam merancang perangkat lunak serta meliputi pengujian dari sistem yang dirancang serta keintegritasannya antara tiap bagian program.

BAB V : KESIMPULAN DAN SARAN

Membahas kesimpulan yang merupakan pendapat terakhir berdasarkan uraian-uraian pada bab sebelumnya dari hasil pengujian sistem serta saran yang berisi tentang gagasan-gagasan yang dapat dikembangkan sebagai kelanjutan dari sistem perangkat lunak ini kedepannya.

BAB II

TINJAUAN TEORITIS

II. 1 Pengertian Senyawa Organik

Kimia organik merupakan salah satu cabang ilmu kimia yang khusus mempelajari *senyawa karbon* yang disebut juga *senyawa organik*. Istilah senyawa organik itu digunakan karena pada mulanya senyawa-senyawa tersebut hanya dapat dihasilkan makhluk hidup dan tidak dapat ditiru dalam laboratorium. Dulu bahkan pernah timbul suatu paham, yaitu *vitalisme*, yang mengatakan bahwa makhluk hidup mempunyai daya hidup (*vis vitalis*) yang memungkinkannya membentuk senyawa organik tersebut. Paham ini merupakan sisa-sisa pemikiran yang berbau mistik, yang lazim dianut sebelum zaman ilmu pengetahuan. Pada tahun 1828, **Frederich Wohler**, seorang ahli kimia bangsa Jerman, berhasil membuat urea (suatu senyawa organik yang terdapat dalam urine mamalia) dari Ammonium Sianat, dengan pemanasan :



Sejak penemuan Wohler ini, *vitalisme* tidak dapat diterima lagi. Kini banyak sekali dari senyawa organik itu yang telah berhasil ditiru, seperti karet, vitamin C, alkohol, asam cuka, dan lain-lain. Walaupun masih banyak yang belum berhasil ditiru, tetapi para ahli yakin bahwa hal ini tinggal menunggu waktu, kelak semuanya akan dapat dibuat. Banyak juga senyawa, yang berdasarkan sifat-sifatnya dimasukkan ke dalam senyawa organik walaupun tidak pernah dihasilkan makhluk hidup, telah berhasil dibuat. Sebagai contoh ialah plastik dan nilon.

Walaupun senyawa-senyawa organik itu telah diyakini tidak harus berasal dari organisme, namun sumber utama senyawa-senyawa tersebut adalah makhluk hidup (tumbuhan dan hewan) atau sisa-sisa makhluk hidup seperti batu bara, minyak bumi, dan gas alam. Karena jumlahnya yang sangat banyak, juga karena sifat serta klasifikasinya yang khas, serta peranannya yang sangat penting dalam kehidupan, maka senyawa karbon yang disebut senyawa organik itu akan dibahas secara khusus dalam *kimia organik*. Semua senyawa yang tidak termasuk senyawa organik disebut senyawa *anorganik*. Perlu disebutkan bahwa tidak semua senyawa karbon tergolong senyawa organik, CO, CO₂ dan karbonat-karbonat misalnya tergolong senyawa anorganik. Sekali lagi penggolongan tersebut berdasarkan sifat-sifat senyawa tersebut.

II. 2 Perbandingan Senyawa Organik Dan Anorganik

Senyawa organik dan anorganik memiliki banyak sekali perbedaan, baik dari segi fisik, sifat dan beberapa aspek lainnya. Beberapa hal yang dapat dibandingkan antara senyawa organik dan Anorganik diantaranya :

II. 2. 1 Dari Segi Jumlah

Jumlah senyawa organik lebih banyak daripada senyawa an-organik. Hal ini terutama disebabkan oleh dua faktor. Pertama, karena atom karbon mampu membentuk ikatan berantai. Kedua, karena kemungkinan pembentukan ikatan karbon-karbon sama dengan kemungkinan pembentukan ikatan karbon dengan beberapa unsur lain, terutama dengan H, O, N dan halogen (F, Cl, Br, dan I).

II. 2. 2 Stabilitas Terhadap Suhu

Senyawa organik jauh kurang stabil dibandingkan dengan senyawa anorganik. Umumnya molekul senyawa organik sudah mengalami penguraian pada suhu diatas 700°C . Hal ini terutama disebabkan karena ikatan dalam senyawa organik adalah ikatan *kovalen* yang energi ikatannya relatif kecil. Stabilitas terhadap zat lain, misalnya terhadap akuaregia, senyawa organik banyak yang lebih stabil daripada senyawa anorganik. Plastik, parafin adalah contoh kelompok zat organik yang tahan terhadap akuaregia.

II. 2. 3 Titik Cair dan Titik Didih

Ikatan *kovalen* yang paling banyak terdapat dalam senyawa organik adalah ikatan C-H yang *nonpolar*. Oleh karena itu, gaya tarik-menarik antar molekul dalam senyawa organik umumnya hanyalah gaya-gaya *Van Der Waals* yang lemah, sehingga titik cair dan titik didihnya relatif rendah. Beberapa golongan senyawa seperti alkohol, *asam karboksilat*, dan lain-lainnya yang mempunyai ikatan hidrogen, memperlihatkan titik cair dan titik didihnya yang menyolok (lihat Tabel 2.1)

TABEL 2. 1 Titik cair (t.c) dan titik didih (t.d) dari beberapa senyawa organik.

Zat*	C_2H_6	$\text{C}_2\text{H}_5\text{Cl}$	$\text{C}_2\text{H}_5\text{OH}$	CH_3COOH
Titik Cair (t.c) ($^{\circ}\text{C}$)	-172	-136,4	-117,3	17
Titik Didih (t.d)($^{\circ}\text{C}$)	-89	-12,27	78,5	118

C_2H_6 bersifat nonpolar, $\text{C}_2\text{H}_5\text{Cl}$ bersifat polar, pada $\text{C}_2\text{H}_5\text{OH}$ dan CH_3COOH

terdapat ikatan hidrogen.

II. 2. 4 Kelarutan

Zat yang polar lebih mudah larut dalam pelarut yang polar daripada dalam pelarut nonpolar, sedangkan zat yang nonpolar lebih mudah larut dalam pelarut nonpolar daripada dalam pelarut yang polar. Karena molekul senyawa organik pada umumnya bersifat nonpolar maka senyawa organik lebih mudah larut dalam pelarut-pelarut nonpolar seperti *benzena*, *tetraklorometana*, dan lain-lain. Beberapa golongan senyawa yang polar seperti karbohidrat sederhana, alkohol rendah, asam karboksilat, dan lain-lain, mudah larut dalam air (pelarut polar).

II. 2. 5 Reaktivitas

Laju reaksi organik umumnya lebih lambat daripada laju reaksi anorganik.

Contoh reaksi :

1. Asam Oksalat + $\text{KMnO}_4 \longrightarrow$ (organik)
2. Garam Mohr + $\text{KMnO}_4 \longrightarrow$ (anorganik)

Reaksi (1) berlangsung lambat pada suhu kamar. Untuk mempercepat reaksi, perlu pemanasan. Reaksi (2) berlangsung cepat. Reaktivitas juga ada hubungannya dengan polaritas molekul. Molekul yang nonpolar, makin besar energi pengaktifan makin kecil laju reaksi. Khusus pembakaran, banyak zat organik yang mudah terbakar seperti alkohol, aseton, hidrokarbon, dan lain-lain.

II. 3 Penggolongan Senyawa Karbon (Senyawa Organik)

Penggolongan senyawa karbon didasarkan pada jenis unsur, jenis ikatan karbon-karbon, bentuk rantai atau jenis gugus fungsinya.

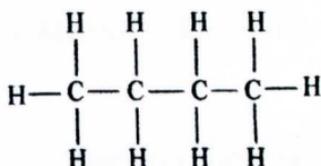
Berdasarkan *jenis unsurnya*, kita mengenal golongan hidrokarbon, yaitu golongan senyawa yang hanya terdiri atas hidrogen dan karbon, karbohidrat mengandung karbon, hidrogen dan oksigen.

Berdasarkan *jenis ikatannya* kita membedakan senyawa jenuh dan senyawa tidak jenuh. Pada senyawa jenuh semua ikatan karbon-karbon adalah ikatan tunggal ($-C-C-$) sedangkan senyawa tidak jenuh ialah senyawa yang mengandung ikatan karbon-karbon rangkap ($-C=C-$) atau ikatan karbon-karbon rangkap tiga ($-C\equiv C-$) (Perhatikan GAMBAR 2.1)

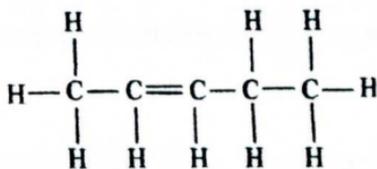
Berdasarkan bentuk rantainya, kita membedakan senyawa *alifatik* dari senyawa *siklik*. Senyawa *alifatik* ialah senyawa dengan rantai terbuka, sedangkan senyawa *siklik* ialah senyawa dengan rantai tertutup (melingkar). Apabila rantai lingkaran (cincin) terdiri atas sejenis atom (karbon) maka disebut *homosiklik* (karbosiklik), sedangkan apabila terdapat satu atom lain (atom hetero) disebut *heterosiklik*. Atom hetero yang paling sering terdapat pada senyawa heterosiklik ialah N, O dan S. Senyawa siklik, seperti benzena, yang dalam molekulnya terdapat *ikatan konjugat* (ikatan tunggal dan ikatan rangkap silih berganti) memperlihatkan sifat yang khas. Golongan senyawa ini disebut senyawa *aromatik*. Semua senyawa *siklik* yang tidak tergolong senyawa aromatik disebut senyawa *alisiklik*, karena sifat-sifatnya mirip dengan senyawa *alifatik*.

Hidrokarbon dapat dianggap sebagai induk dari semua senyawa organik. Atom H dari hidrokarbon itu dapat digantikan oleh atom atau gugus atom lain sehingga terbentuk

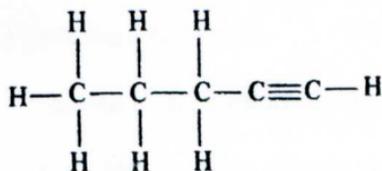
senyawa-senyawa lain. Gugus pengganti itu menjadi bagian yang reaktif dalam molekul dan memberi sifat khas pada senyawa yang bersangkutan, oleh karena itu disebut *gugus fungsi*. Senyawa-senyawa dengan gugus fungsi yang sama menunjukkan kemiripan sifat. Jenis ikatan karbon-karbon juga dianggap sebagai gugus fungsi, karena jenis ikatan dalam suatu senyawa sangat mempengaruhi sifat-sifat atau reaksi-reaksi senyawa tersebut. Penggolongan senyawa organik berdasarkan gugus fungsi dapat dilihat pada tabel dibawah.



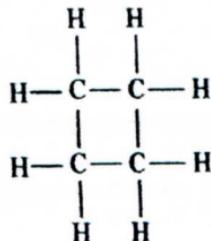
N-Butana
(Hidrokarbon alifatik Jenuh)



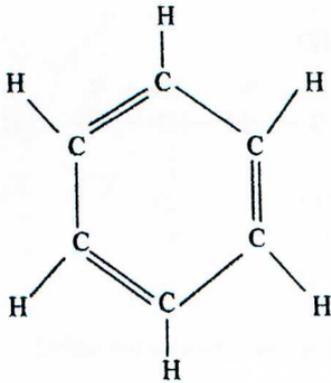
2-Pentena
(Hidrokarbon alifatik tak jenuh)



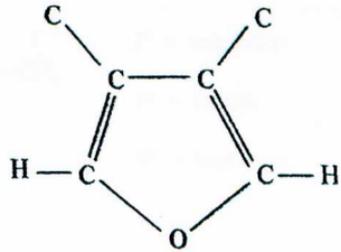
1-Pentena
(Hidrokarbon alifatik takjenuh)



SikloButana
(suatu senyawa alisiklik)



Benzena
(suatu senyawa aromatik)



Furan
(suatu senyawa heterosiklik)

GAMBAR 2.1 Berbagai Jenis Ikatan Karbon-Karbon Dalam Senyawa Organik

Suku-suku dari suatu golongan senyawa dengan gugus fungsi yang sama membentuk suatu deret yang disebut *deret sepancaraan* atau *deret homolog*. Dalam suatu deret homolog suku-suku yang berturutan berbeda sebesar CH_2 . Misalnya deret sepancaraan alkana yang suku-sukunya ialah CH_4 , C_2H_6 , C_3H_8 dan seterusnya.

II. 4 Hidrokarbon Alkana

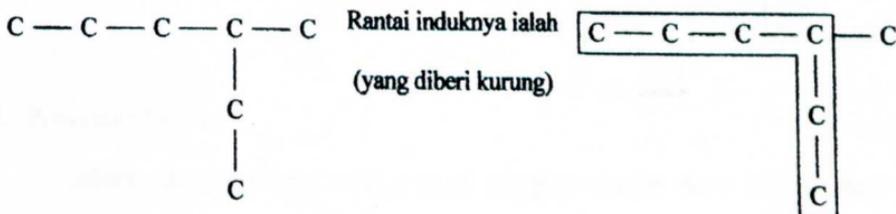
Alkana ialah hidrokarbon *alifatik* jenuh, dengan rumus empirik $\text{C}_n\text{H}_{2n+2}$. Ada 4 jenis atom karbon dalam *alkana*, yaitu atom karbon primer, sekunder, tersier, dan kuaterner. Atom karbon primer ialah atom karbon yang terikat langsung hanya pada 1 atom karbon yang lain, atom karbon sekunder ialah atom karbon yang terikat langsung pada 2 atom karbon yang lain, dan seterusnya.

II. 4. 1. 1 Tata Nama Alkana Bercabang

1. Pemilihan Rantai Induk

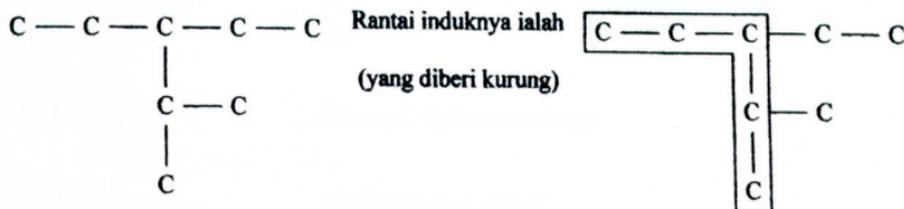
Rantai induk ialah rantai atom karbon terpanjang.

Contoh :

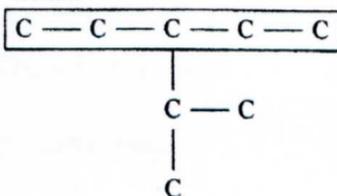


Apabila terdapat lebih dari 1 rantai terpanjang, pilih yang mempunyai cabang terbanyak.

Contoh :



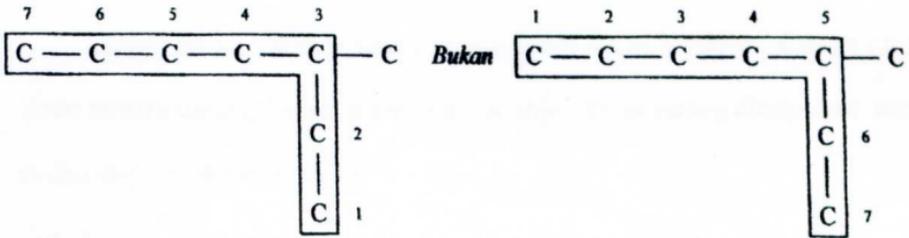
Bukan



2. Penomoran

Penomoran dimuali dari salah satu ujung rantai induk sehingga cabang-cabang mendapat nomor terkecil.

Contoh :



3. Penamaan Cabang

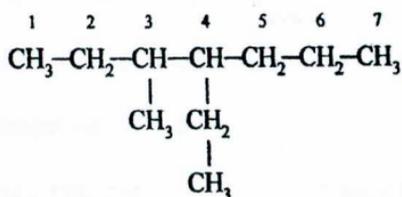
Cabang-cabang dianggap sebagai gugus pengganti dengan nama *alkil*. Gugus alkil mempunyai rumus empirik C_nH_{2n+1} dan biasa dilambangkan dengan mengganti akhiran *ana* menjadi *il*. Berikut ini disajikan gugus-gugus alkil terpenting :

CH_3-	Metil
CH_3-CH_2-	Etil
$CH_3-CH_2-CH_2-$	Normal-Propil atau <i>n</i> -Propil
$\begin{array}{c} CH_3-CH- \\ \\ CH_3 \end{array}$	Iso-Propil atau <i>i</i> -Propil
$CH_3-CH_2-CH_2-CH_2-$	Norma-Butil atau <i>n</i> -Butil
$\begin{array}{c} CH_3-CH-CH_2- \\ \\ CH_3 \end{array}$	Iso-Butil atau <i>i</i> -Butil
$\begin{array}{c} CH_3-CH_2-CH- \\ \\ CH_3 \end{array}$	Sekunder-Butil atau <i>s</i> -Butil
$\begin{array}{c} CH_3 \\ \\ CH_3-C- \\ \\ CH_3 \end{array}$	Tertier-Buti atau <i>t</i> -Butil

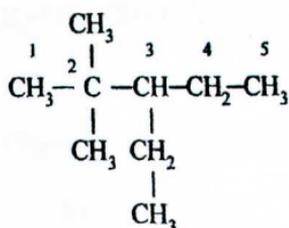
4. Penulisan Nama

Setiap *alkana* dinamai sesuai dengan nama rantai induknya. Cabang-cabang ditulis terlebih dahulu, disusun menurut urutan abjad. Posisi cabang ditunjukkan dengan awalan angka pada nama cabang.

Contoh :



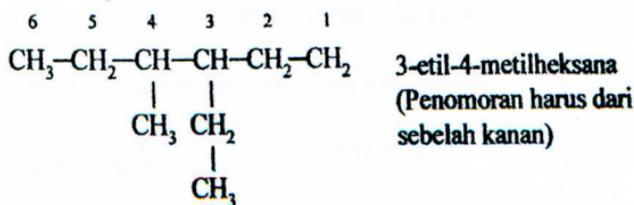
4-etil-3-metilheptana



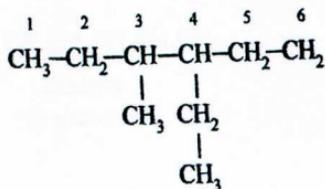
3-etil-2,2-dimetilpentana

Jika penomoran ekuivalen dari kedua ujung maka cabang yang akan ditulis terlebih dahulu harus mendapat nomor terkecil.

Contoh :

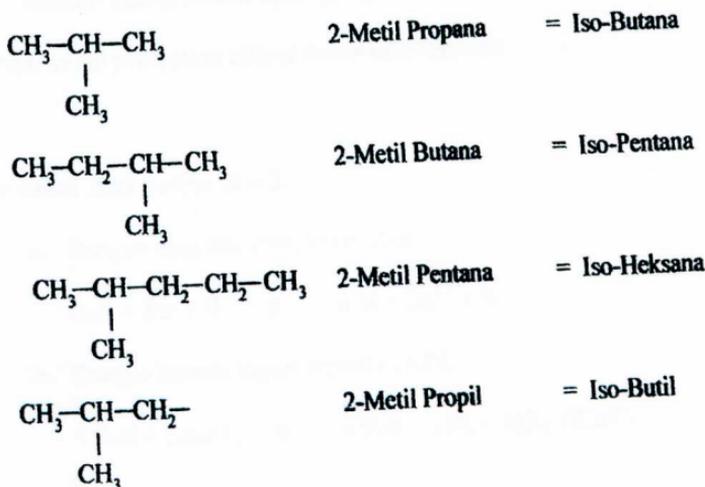


Bukan



4-etil-3-metilheksana

Isomer yang mempunyai 1 cabang metil pada atom C nomor 2 dari ujung rantai ditandai dengan *iso*. Contohnya :

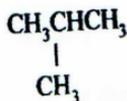


II. 4. 1. 2 Isomeri

Isomeri pada *alkana* adalah isomeri kerangka. Isomeri mulai terdapat pada butana. Butana mempunyai 2 isomeri, yaitu *n*-butana dan *i*-butana.



n-Butana



i-Butana

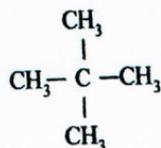
Pentana mempunyai 3 isomer, yaitu normal-pentana, iso-pentana, dan neo-pentana :



Pentana
(*n*-pentana)



2-metilbutana
(iso-pentana)



2,2-dimetilpropana
(neo-pentana)

II. 4. 2 Pembuatan Alkana

Sumber utama *alkana* ialah gas alam dan minyak bumi. Berikut ini diberikan beberapa reaksi pembuatan *alkana* dalam laboratorium.

1. Reduksi Alkil halida, R - X

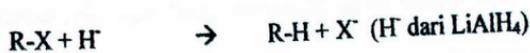
- a. Dengan seng dan asam keras encer :



- b. Dengan hibrida logam seperti LiAlH_4 :



atau



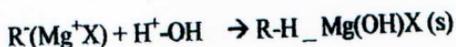
- c. Melalui senyawa organometalik (pereaksi Grignard) :

Alkil halida dapat bereaksi dengan Mg atau Li dalam eter kering menghasilkan suatu senyawa organometalik di mana gugus alkilnya bersifat basa karbanion (bermuatan negatif).

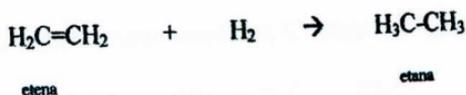
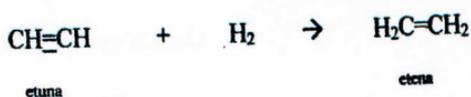


pereaksi Grignard

Hidrolisis perekat Grignard menghasilkan *alkana* :



2. Hidrogenasi Alkena Atau Alkuna, Dengan Katalisator Pt Atau Ni.



3. Sintesis Wurtz, dari alkilhalida dan logam natrium.

Dengan cara ini dihasilkan *alkana* dengan jumlah atom karbon yang lebih banyak dari senyawa asal :



Atau



II. 4. 3 Sifat-Sifat Alkana

Pada suhu kamar, 4 suku pertama berupa gas, suku-suku berikutnya berupa cairan sedangkan suku-suku tinggi berupa zat padat. Makin besar massa molekul relatif (M_r) makin besar titik cair, titik didih maupun massa jenisnya. Isomer rantai lurus mempunyai titik cair, titik didih serta massa jenis lebih besar dibandingkan isomer bercabang. Semua *alkana* sukar larut dalam air.



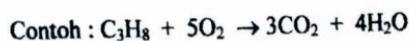
II. 4. 4 Reaksi-Reaksi Alkana

Alkana sukar bereaksi, oleh karena itu disebut *parafin* yang berarti *afinitas kecil*.

Reaksi-reaksi terpenting dari *alkana* adalah pembakaran, penggantian (substitusi) dan "cracking" (pengretakan).

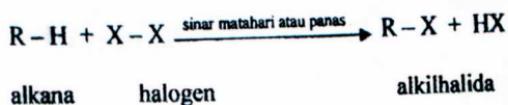
1. Pembakaran

Pada pembakaran sempurna, C menjadi CO_2 dan H menjadi H_2O .



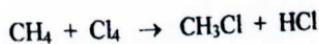
2. Penggantian (substitusi)

Dalam hal ini, atom H dari alkana digantikan oleh atom atau gugus atom lain : Yang terpenting dari substitusi alkana ialah halogenasi. Halogenasi dapat berlangsung bila campuran alkana dengan halogen dipanaskan dengan suatu katalisator atau bila mendapat radiasi ultra violet :

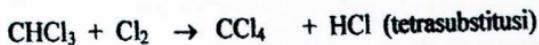
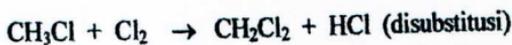


Urutan keaktifan halogen : $\text{F}_2 > \text{Cl}_2 > \text{Br}_2$, I_2 tidak bereaksi tetapi dapat digunakan sebagai katalisator pada halogenasi tersebut.

Contoh : Klorinasi metana



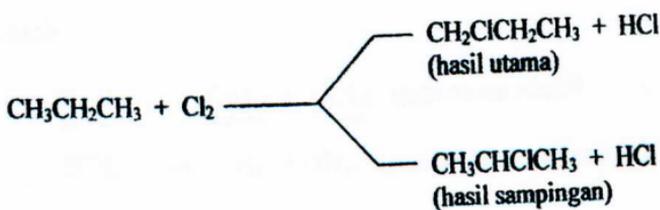
Halogenasi metana tidaklah berakhir dengan monosubstitusi tetapi dapat berlanjut sehingga semua atom H dari metana itu digantikan. Hasil halogenasi alkana juga dipengaruhi perbandingan mol antara alkana dan halogen yang direaksikan.



Untuk propana dan alkana yang lebih besar, yang mempunyai lebih dari sejenis atom H, hasil substitusi tergantung pada 3 faktor :

a. Faktor kemungkinan (probabilitas)

Misalnya, klorinasi propana lebih banyak menghasilkan 1-kloropropana daripada 2-kloropropana. Hal ini karena propana mempunyai 6 atom H primer dan hanya 2 atom H sekunder, jadi kemungkinan terjadinya tumbukan antara atom-atom Cl dengan atom H primer lebih besar daripada tumbukan antara atom-atom Cl dengan atom H sekunder :



b. Reaktivitas atom H

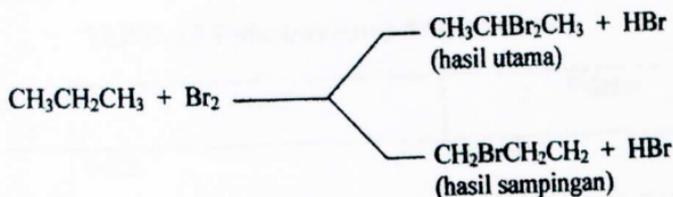
Urutan reaktivitas atom H pada alkana : $3^\circ > 2^\circ > 1^\circ$

Berarti atom H primer paling kurang reaktif, paling sukar digantikan.

c. Reaktivitas halogen

Telah disebutkan, urutan keaktifan halogen : $\text{F}_2 > \text{Cl}_2 > \text{Br}_2$

Br_2 yang kurang reaktif akan bersifat memilih-milih (selektif), akan mudah menggantikan atom H yang lebih reaktif. Brominasi propana lebih banyak menghasilkan 2-bromopropana :

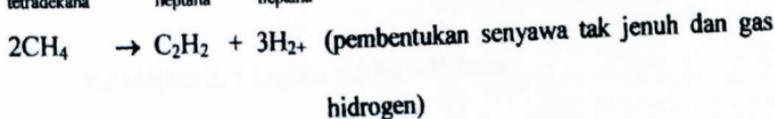
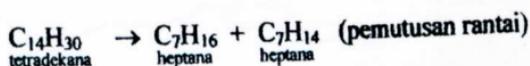


Fluor dan klor yang sangat reaktif tidak bersifat selektif, hasil reaksinya ditentukan faktor kemungkinan.

d. Cracking

Cracking adalah pemanasan dengan tekanan tinggi tanpa oksigen. Pada proses cracking dapat terjadi pemutusan rantai, pembentukan senyawa tak jenuh atau perubahan rantai lurus menjadi rantai bercabang. Proses cracking juga dapat menghasilkan gas hidrogen.

Contoh :



II. 5 Pointer Dan *Linked List*

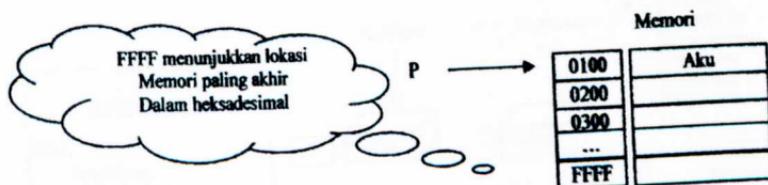
Tipe data pointer bersifat dinamis, variabel akan dialokasikan hanya pada saat dibutuhkan dan sesudah tidak dibutuhkan dapat dialokasikan kembali.

Tabel di bawah berikut ini diberikan perbedaan antara variabel bertipe array dengan variabel bertipe pointer.

TABEL 2.3 Perbedaan Array dengan Pointer

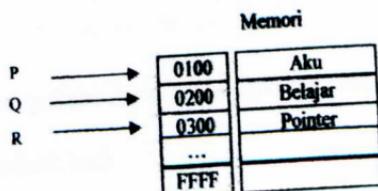
Kriteria	Array	Pointer
Sifat	Statis	Dinamis
Ukuran	Pasti	Sesuai kebutuhan
Alokasi variabel	Saat program dijalankan sampai selesai	Dapat diatur sesuai kebutuhan

Variabel pointer adalah suatu variabel yang menunjuk ke alamat memori yang digunakan untuk menampung data yang akan diproses, seperti digambarkan di bawah ini:

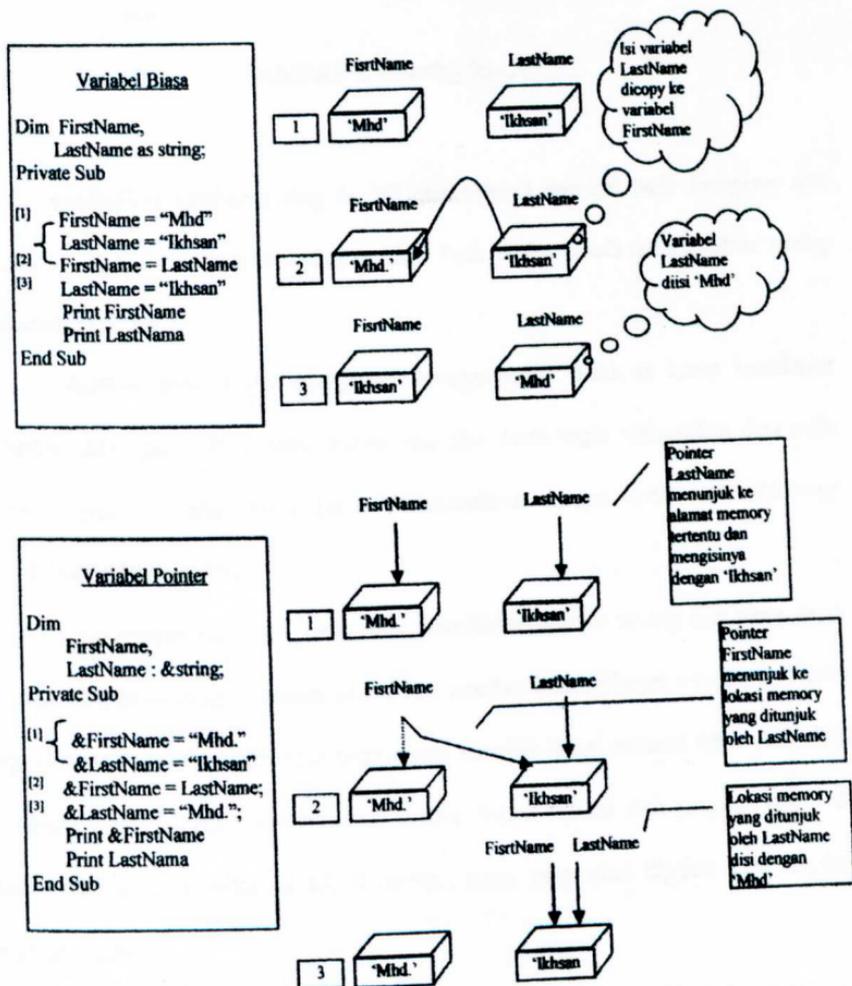


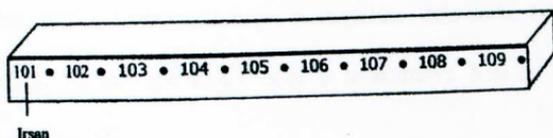
GAMBAR 2.2 Logika Variable Pointer

P adalah variabel pointer yang menunjuk ke alamat memori 100 yang berisi data bertipe string "Aku". Apabila Anda ingin menambahkan data dengan menggunakan variabel yang berbeda, maka Anda dapat mendeklarasikan variabel pointer baru misalnya Q dan R dan seterusnya sehingga tampak sebagai berikut :



Untuk memudahkan analogi antara variabel pointer dengan variabel biasa, perhatikan contoh berikut :





GAMBAR 2. 4 *Saving box* Awal

Analogikan gambar *saving box* di atas sebagai memori pada komputer dan nomor *saving box* sebagai alamat dari memori. Pada keadaan mula-mula keadaan *saving box* adalah kosong.

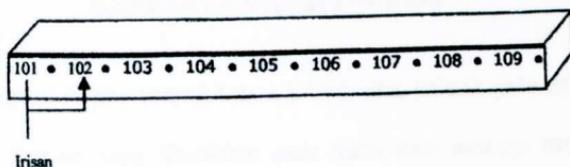
Misalkan Irsan ingin mengisinya (menyewanya) maka ia harus membayar sejumlah dana pada pihak bank, begitu pula jika Anda ingin mengisi data pada variabel pointer maka Anda harus membayarnya dengan melakukan deklarasi `New(<NamaVarPointer>);`

Baru setelah itu pihak bank akan memilihkan nomor *saving box* yang dapat dipakai oleh Irsan dengan catatan *saving box* tersebut belum disewa orang lain, begitu juga dengan variabel pointer akan bebas untuk memilih lokasi memori mana yang akan ia sewa (tunjuk) dengan catatan lokasi tersebut belum dipakai oleh program lain. Jadi Anda tidak usah memikirkan lokasi memori mana yang akan dipakai oleh variabel pointer tersebut.

Setelah mendapatkan nomor *saving box* dari bank yaitu A101, maka Irsan akan mulai memasukkan barang-barang berharga miliknya ke dalamnya, begitu pula dengan variabel pointer akan memasukkannya ke dalam tempat yang telah ditunjuknya. Ternyata *saving box* telah penuh namun masih ada barang berharga Irsan yang masih belum tersimpan, maka ia dapat menyewa *saving box* yang lain. Begitu pula dengan

variabel pointer, jika Anda masih ingin memasukkan data maka variabel pointer akan menunjuk lokasi memori lain seperti langkah sebelumnya. Kali ini Irsan mendapatkan *saving box* nomor A102.

Untuk lebih jelasnya perhatikan gambar berikut.



GAMBAR 2. 5 *Saving box* Perluasan

Jika tempatnya masih kurang memenuhi maka Irsan dapat menyewa kembali seperti sebelumnya sampai tidak ada lagi *saving box* yang tersisa, begitupula dengan variabel pointer akan menunjuk ke lokasi memori yang masih kosong sampai semua lokasi pada memori terisi penuh.

II. 5. 1 Single *Linked List*

Apabila setiap kali Anda ingin menambahkan data selalu dengan menggunakan variabel pointer yang baru, Anda akan membutuhkan banyak sekali variabel pointer (penunjuk).

Oleh karena itu ada baiknya jika Anda hanya menggunakan satu variabel pointer saja untuk menyimpan banyak data dengan metoda yang kita sebut *Linked List*. Jika diterjemahkan, maka berarti suatu daftar isi yang saling berhubungan. Untuk lebih jelasnya perhatikan gambar di bawah ini :



GAMBAR 2. 6 Struktur *Linked List*

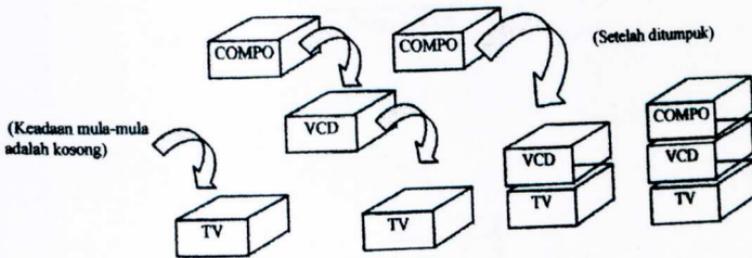
Pada gambar di atas tampak bahwa sebuah data terletak pada sebuah lokasi *memory area*. Tempat yang disediakan pada suatu area *memory* tertentu untuk menyimpan data dikenal dengan sebutan *node/simpul*. Pada setiap *node* memiliki pointer (penunjuk) yang menunjuk ke simpul berikutnya sehingga terbentuk suatu untai dan dengan demikian hanya diperlukan sebuah variabel pointer. Susunan berupa untai semacam ini disebut *Single Linked List*. (ket : Nil tak memiliki nilai apapun. Biasanya *Linked List* pada titik akhirnya akan menunjuk ke Nil).

Dalam pembuatan *Single Linked List* dapat menggunakan 2 metoda :

1. LIFO (Last In First Out), aplikasinya : Stack (Tumpukan).
2. FIFO (First In First Out), aplikasinya : Queue (Antrian).

II. 5. 1. 1 LIFO (Last In First Out)

LIFO adalah suatu metoda pembuatan *Linked List* dimana data yang masuk paling akhir adalah data yang keluar paling awal. Hal ini dapat dianalogikan (dalam kehidupan sehari-hari) pada saat Anda menumpuk barang, seperti digambarkan di bawah ini :

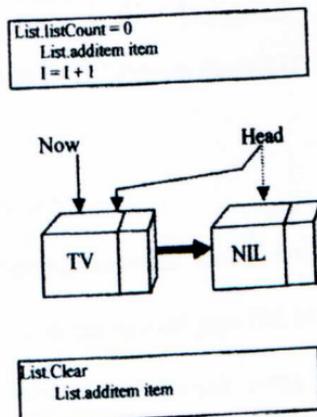


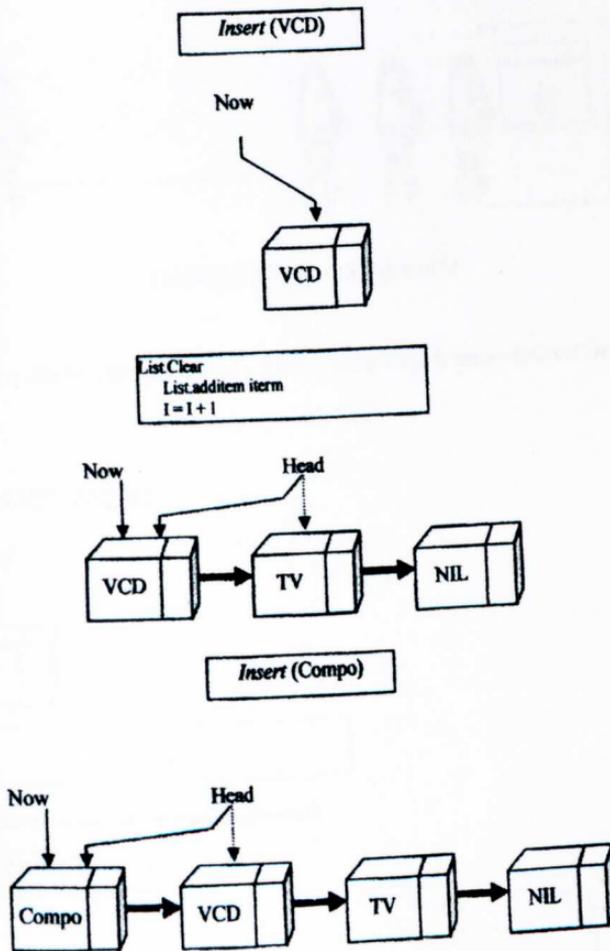
GAMBAR 2. 6 Analogi FIFO

Pembuatan sebuah simpul dalam suatu *Linked List* seperti digambarkan di atas disebut dengan istilah *INSERT*. Jika *Linked List* dibuat dengan metoda LIFO, maka terjadi penambahan/ *Insert* simpul di belakang.

- **PROCEDUR *INSERT***

Istilah *Insert* berarti menambahkan sebuah simpul baru ke dalam suatu *Linked List*. Berikut di bawah ini adalah prinsip kerja LIFO.

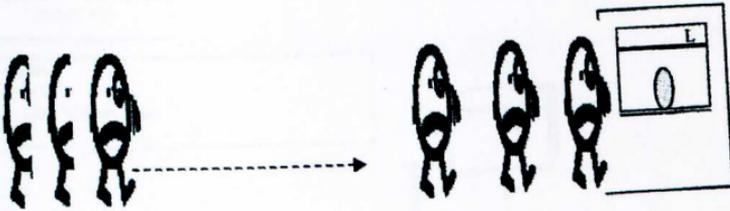




GAMBAR 2. 7 Prinsip Kerja LIFO

II. 5. 1. 2 FIFO (First In First Out)

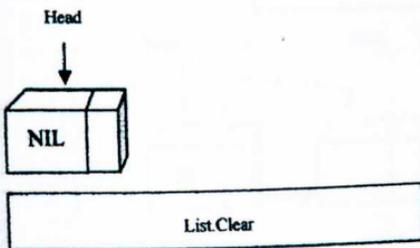
FIFO adalah suatu metoda pembuatan *Linked List* dimana data yang masuk paling awal adalah data yang keluar paling awal juga. Hal ini dapat dianalogikan (dalam kehidupan sehari-hari misalnya saat sekelompok orang yang datang (*ENQUEUE*) mengantri hendak membeli tiket di loket.



GAMBAR 2. 8 Analogi FIFO

Jika *Linked List* dibuat dengan metoda FIFO, maka terjadi penambahan/ *Insert* simbol di depan.

• **PROCEDURE INSERT**

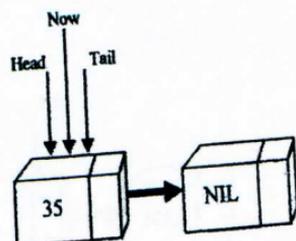
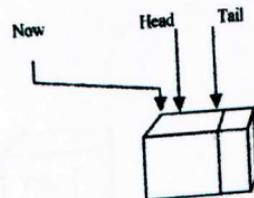


(head mula-mula selalu diidentifikasi sebagai nil)

Insert (35)

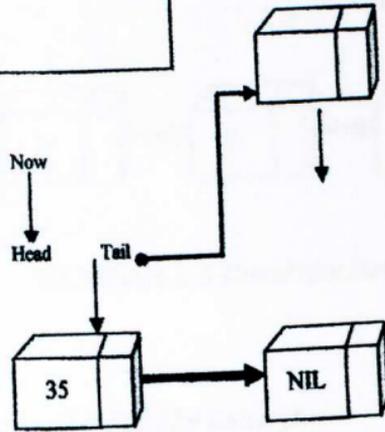
```
List.additem item
I = I + 1
List.list(i) <> VbNullString
List.additem item
```

```
List.additem VbNullString
I = I + 1
List.list(i).additem
```

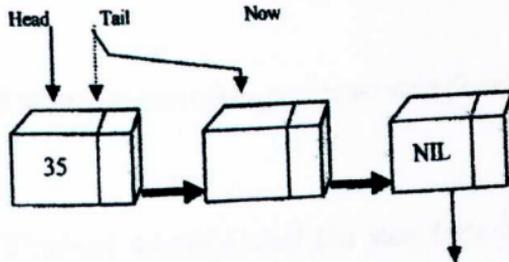


Insert (5)

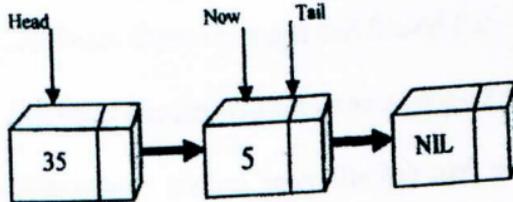
List.Clear
I=0
List.AddItem Item



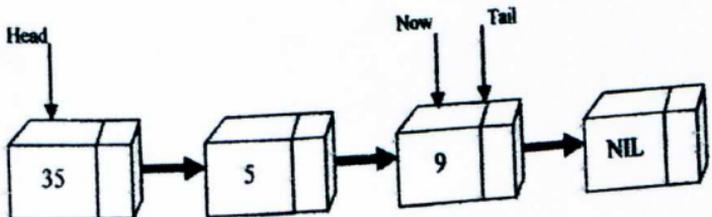
$I = I + 1$
List.AddItem Item

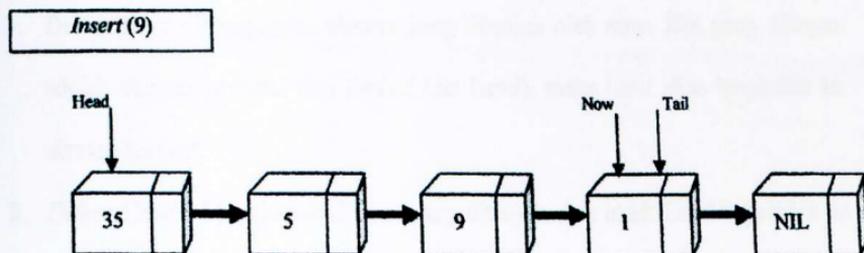


List.AddItem 35



Insert (9)





GAMBAR 2. 9 *Procedure Insert*

- *Procedure dan Function Linked List Lainnya*

Selain *procedure Insert* di atas, pada *Linked List* juga terdapat *procedure* serta *function* lainnya.

Di bawah ini diberikan *procedure-procedure* serta *function* umum dalam aplikasi *Linked List*.

1. *Create* : Membuat sebuah *Linked List* yang baru dan masih kosong. (ket : *procedure* ini wajib dilakukan sebelum menggunakan *Linked List*).
2. *Empty* : *Function* untuk menentukan apakah *Linked List* kosong atau tidak.
3. *Find First* : Mencari elemen pertama dari *Linked List*.
4. *Find Next* : Mencari elemen sesudah elemen yang ditunjuk *now*.
5. *Retrieve* : Mengambil elemen yang ditunjuk oleh *now*. Elemen tersebut lalu ditampung pada suatu variabel.
6. *Update* : Mengubah elemen yang ditunjuk oleh *now* dengan isi dari suatu variabel.

7. *Delete Now* : Menghapus elemen yang ditunjuk oleh *now*. Jika yang dihapus adalah elemen pertama dari *Linked List* (*head*), maka *head* akan berpindah ke elemen berikut.
8. *Delete Head* : Menghapus elemen yang ditunjuk oleh *head*. *Head* berpindah ke elemen sesudahnya.
9. *Clear* : Untuk menghapus *Linked List* yang sudah ada.

Wajib dilakukan bila ingin mengakhiri program yang menggunakan *Linked List*. Jika tidak data-data yang dialokasikan ke memori pada program sebelumnya akan tetap tertinggal di dalam memori.

II. 5. 2 *Double Linked List*

Salah satu kelemahan dari *single Linked List* adalah pointer (penunjuk) hanya dapat bergerak satu arah saja, maju atau mundur/kanan atau kiri. Sehingga pencarian data pada *single Linked List* hanya dapat bergerak dalam satu arah saja. Untuk mengatasi kelemahan tersebut, Anda dapat menggunakan metoda *Double Linked List*. *Linked List* ini dikenal dengan nama *Linked List* berpointer Ganda atau *Double Linked List*. Untuk jelasnya lihat gambar di bawah :



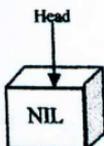
GAMBAR 2. 10 Struktur *Double Linked List*

II. 5. 2. 1 Operasi-Operasi Pada *Double Linked List*

• *Insert After*

Procedure Insert After berguna untuk menambah simpul di belakang (sebelah kanan) pada sebuah *Double Linked List*. Berikut penggalan *procedure Insert After* beserta penjelasannya.

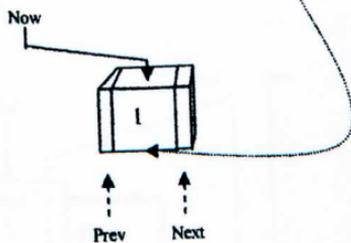
DeleteNow



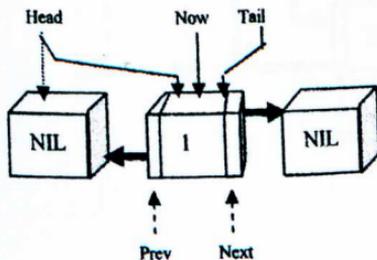
```
List.ListCount < 0 then
  List.AddItem Item
  I = I + 1
  List.AddTerm VbNullString
```

InsertAfter(1)

(buat simpul baru)



```
List.Clear
List.AddItem 1
```

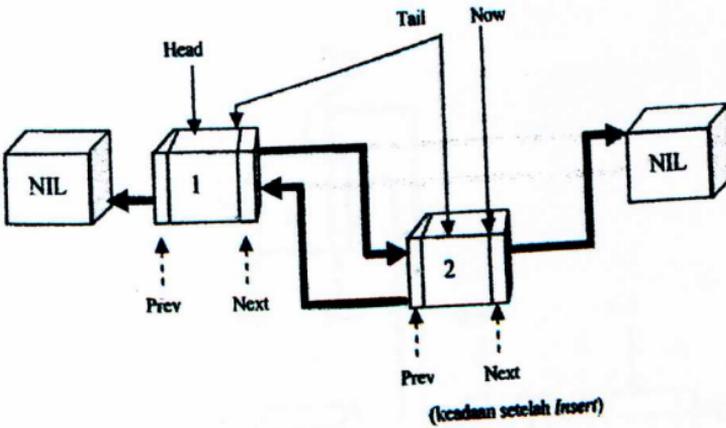
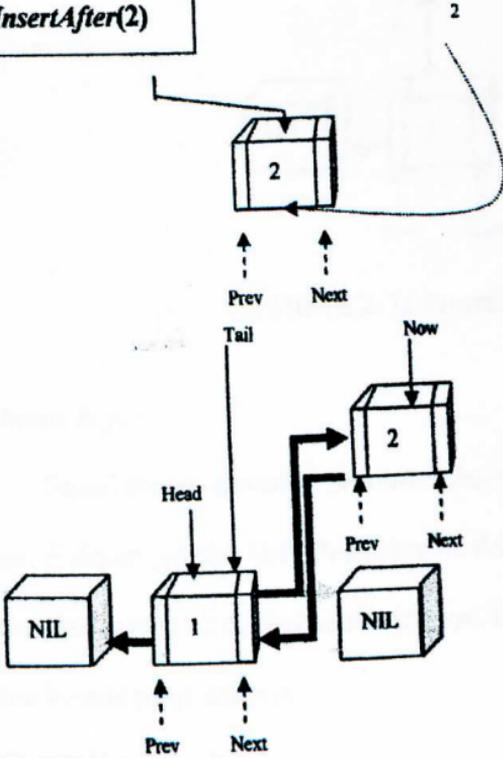


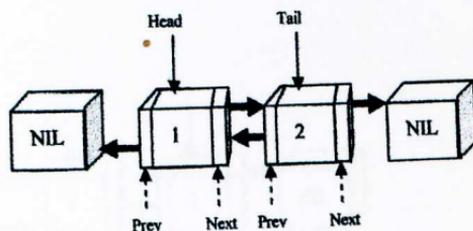
```
List.Clear
I = 0
List.AddItem = list(i)
I = I + 1
List.AddTerm = List(i)
List.AddItem = vbNullString
```

InsertAfter(2)

```

List.Clear
I = 0
List.AddItem = list(i)
I = I + 1
List.AddItem = List(i)
List.AddItem = vbNullString
    
```





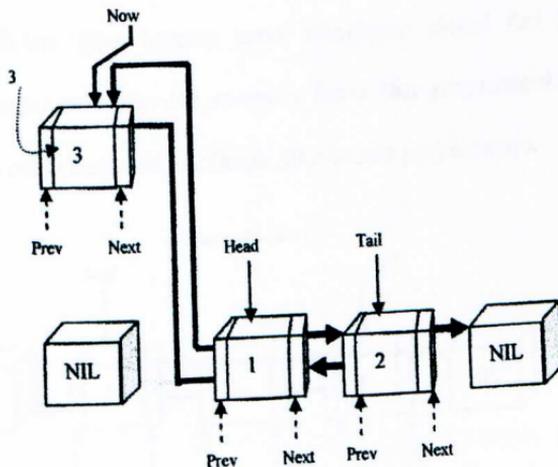
GAMBAR 2. 11 *Procedure Insert After*

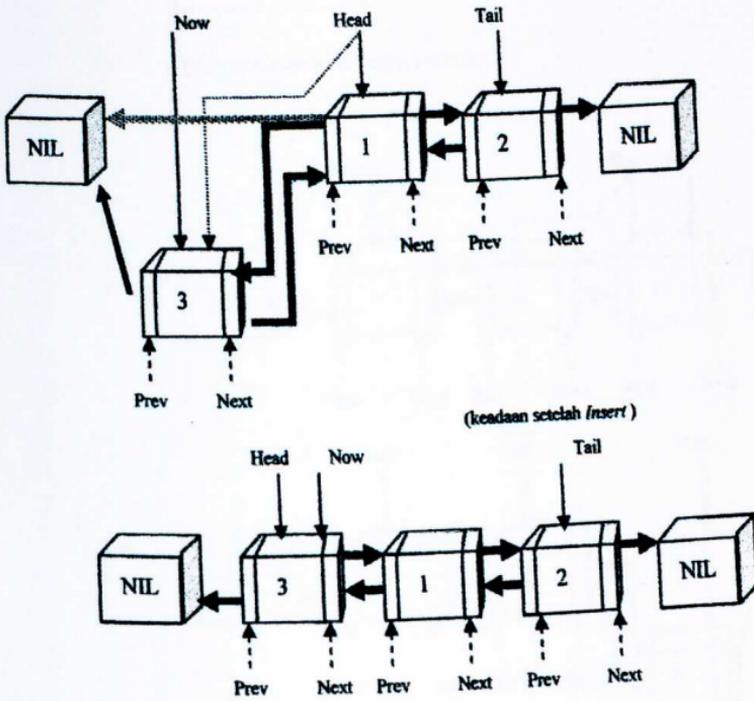
- ***Insert Before***

Sesuai dengan namanya, *procedure Insert Before* berguna untuk menambah simpul di depan (sebelah kiri). *Procedure* ini tidak berbeda jauh dengan *procedure Insert After* yang telah dijelaskan sebelumnya. Berikut penggalan *procedure Insert Before* beserta penjelasannya.

Insert Before(3)

(keadaan mula-mula merupakan lanjutan sebelumnya)

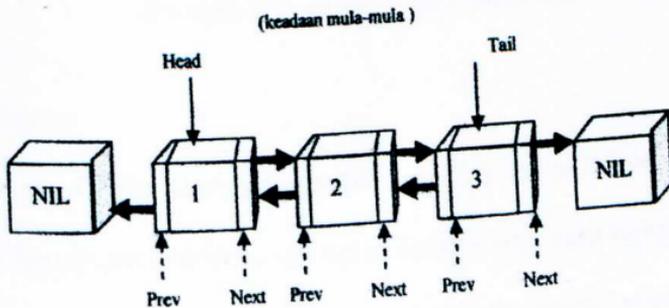




GAMBAR 2. 12 Procedure Insert Before

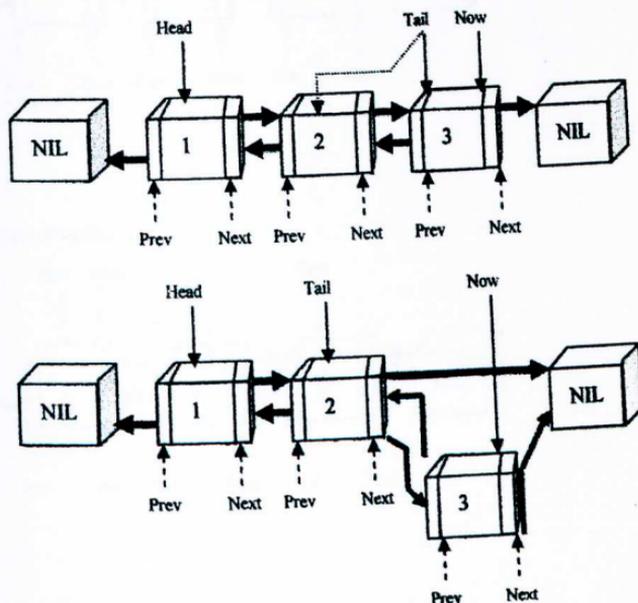
- **Delete After**

Procedure Delete After berguna untuk menghapus simpul dari belakang. *Procedure* ini merupakan kebalikan dari *procedure Insert After* yang menambah simpul di belakang. Berikut penggalan *procedure Delete After* beserta penjelasannya.

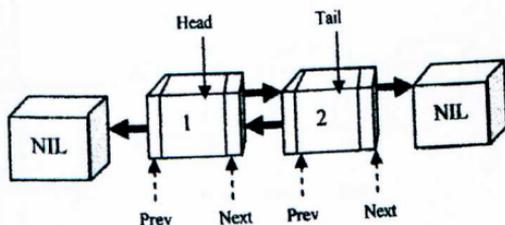


Delete After

(Now menunjuk simpul paling belakang)



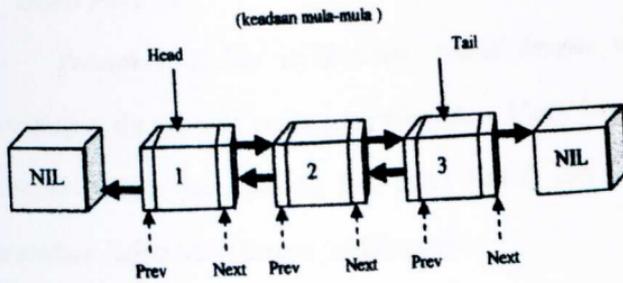
(keadaan setelah di- Delete



GAMBAR 2. 12 Procedure Delete After

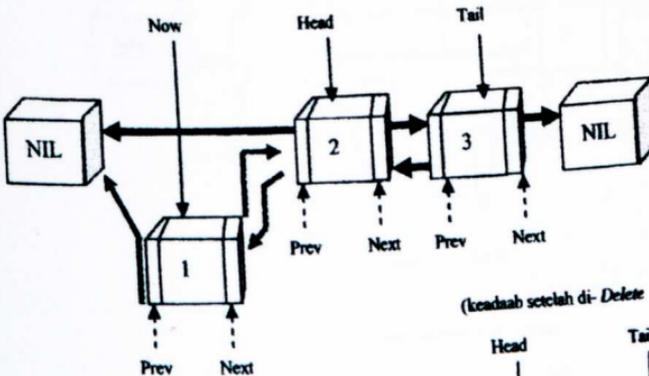
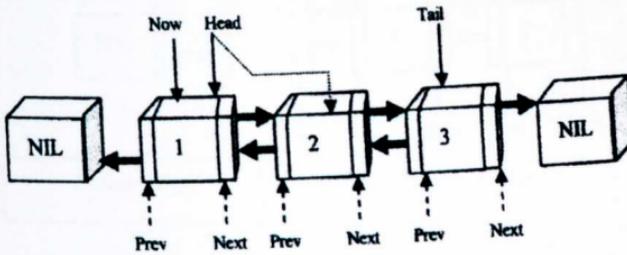
- Delete Before**

Procedure Delete Before merupakan kebalikan dari *procedure Delete After* yang menghapus simpul dari belakang, sedangkan *Delete Before* akan menghapus simpul dari depan (sebelah kiri). Berikut penggalan *procedure Delete Before* beserta penjelasannya.

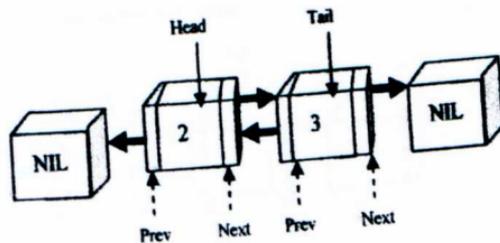


DeleteBefore

(Now menunjuk simpul paling depan)



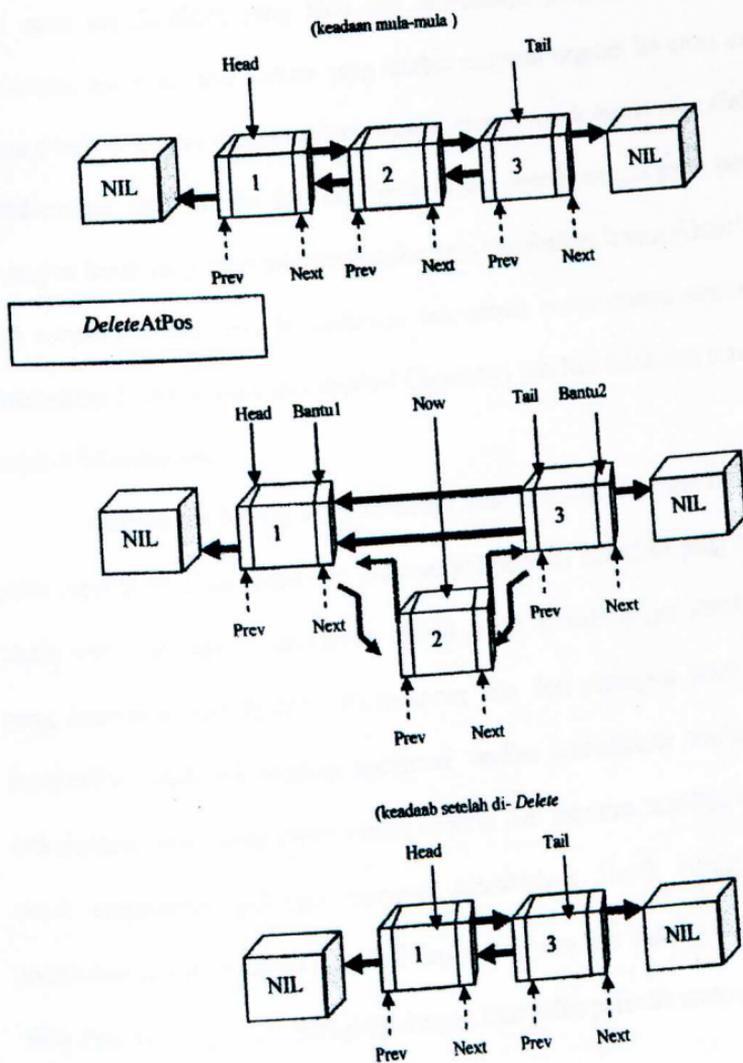
(keadaan setelah di-Delete)



GAMBAR 2. 13 Procedure Delete Before

- **Delete Position**

Procedure Delete at Position, sesuai dengan namanya, berguna untuk menghapus simpul pada posisi yang diinginkan. Untuk itu diperlukan bantuan 2 buah variabel pointer yang penulis beri nama Bantu1 dan Bantu2. Berikut penggalan *procedure DeleteAtPos* beserta penjelasannya.



GAMBAR 2. 14 *Procedure Delete Position*

BAB III

PERANCANGAN SISTEM

III. 1 Perancangan

Senyawa organik dari segi kuantitas cukup banyak terdapat di alam, juga karena sifat serta klasifikasinya yang khas dan peranannya yang sangat penting dalam kehidupan, maka senyawa karbon yang disebut senyawa organik itu terus diusahakan untuk dibuat dan ditiru dilaboratorium. Namun banyak aspek teknis yang dinilai cukup memberatkan, salah satunya dari segi biaya. Maka penulis merasa perlu adanya suatu perangkat lunak yang dapat menganalisa dan men-simulasikan kemungkinan hasil reaksi dari senyawa hidrokarbon, koefisiennya dan aturan penamaannya menurut IUPAC (*Internation Union of Pure and Applied Chemistry*) sebelum dilakukan pengujian lebih lanjut di laboratorium.

Perangkat lunak yang akan dirancang akan memiliki beberapa fitur diantaranya yaitu dapat menentukan penamaan senyawa hidrokarbon kompleks yang masih dibatasi hanya untuk golongan Hidrokarbon alifatik jenuh sesuai dengan standard penamaan yang ditentukan oleh IUPAC. Kemampuan lain dari perangkat lunak adalah dapat memberikan deskripsi lengkap mengenai analisa pembakaran sembarang senyawa hidrokarbon untuk menentukan rumus empirik dari senyawa tersebut. Hal ini berguna untuk menentukan golongan senyawa hidrokarbon. Untuk mempermudah dalam melakukan pengkalkulasian data dari tiap reaksi yang ada maka perangkat lunak yang akan dirancang juga akan dilengkapi dengan fitur daftar periodik unsur-unsur.

Untuk menghindari kesan teknik pada perangkat lunak maka Perangkat lunak nantinya akan dirancang agar memiliki desain interface yang menarik. Sehingga pemakai yang akan menggunakan perangkat lunak ini merasakan kemudahan dan keindahan yang ditawarkan oleh perangkat lunak.

III. 1. 1 Pemilihan Bahasa Pemrograman

Perangkat lunak yang akan dirancang harus dapat berjalan pada platform Windows apa saja, baik itu Windows versi 9x (Windows 95, 98 dan Me) dan Windows NT (Windows 2000 dan XP). Maka perangkat lunak juga harus dirancang dengan bahasa pemrograman 32-bit dan berbasis sistem operasi Windows.

Permasalahan yang akan dirancang perangkat lunaknya merupakan permasalahan yang bersifat teknik sehingga pendekatan bahasa pemrograman juga harus menggunakan bahasa yang mendukung beberapa aspek-aspek teknik.

Perangkat lunak direncanakan dapat berjalan dengan baik pada sistem perangkat keras dengan spesifikasi minimal *Pentium II* dan dengan Memory 64 MB, maka perangkat lunak ini harus dirancang dan di-*compile* dengan menggunakan bahasa pemrograman yang hasil file *executable*-nya menggunakan *resources* perangkat keras yang minimal.

Untuk memenuhi semua kriteria diatas maka penulis menggunakan bahasa pemrograman berorientasi objek Microsoft Visual Basic 6.0 yang merupakan bahasa pemrograman 32-bit, dan dapat digunakan untuk pemrograman yang terkait dengan aspek teknik, bisnis dan lainnya dan memiliki file *executable* yang sangat *low resources* juga dilengkapi dengan *library-library* yang mendukung pengaksesan ke sistem

Windows. Disamping itu semua bahasa pemrograman Microsoft Visual Basic terintegrasi dengan baik sekali dengan *platform* Windows versi apa saja dan dukungan kemudahannya untuk dapat memodifikasi perangkat lunak jika ada pengembangan perangkat lunak kedepannya.

III. 1. 2 Perancangan Perangkat Lunak

Dalam merancang perangkat lunak penulis menggunakan pendekatan terstruktur (*structured approach*) yang merupakan pendekatan di dalam merancang program yang sangat populer saat ini. Pendekatan ini dilakukan dengan cara memecah-mecah suatu masalah yang besar dan rumit menjadi beberapa masalah yang lebih kecil dalam bentuk modul-modul sehingga menjadi cukup mudah ditangani.

Suatu modul adalah kumpulan dari instruksi yang melakukan suatu fungsi dasar tertentu yang bersifat *independent*. Masing-masing modul harus mewakili suatu bagian masalah inti tertentu dari keseluruhan masalah.

Salah satu teknik dalam pendekatan perancangan terstruktur yang paling terkenal adalah *top-down design*. Akan lebih realistis bila rancangan suatu program dimulai dari atas menuju ke bawah (*top down*), yaitu dengan menentukan kebutuhan secara umum terlebih dahulu, memecahnya menjadi modul-modul yang lebih sederhana lagi dan seterusnya.

Dengan pendekatan terstruktur penulis mendapat beberapa keuntungan dibandingkan dengan pendekatan secara tradisional, yaitu :

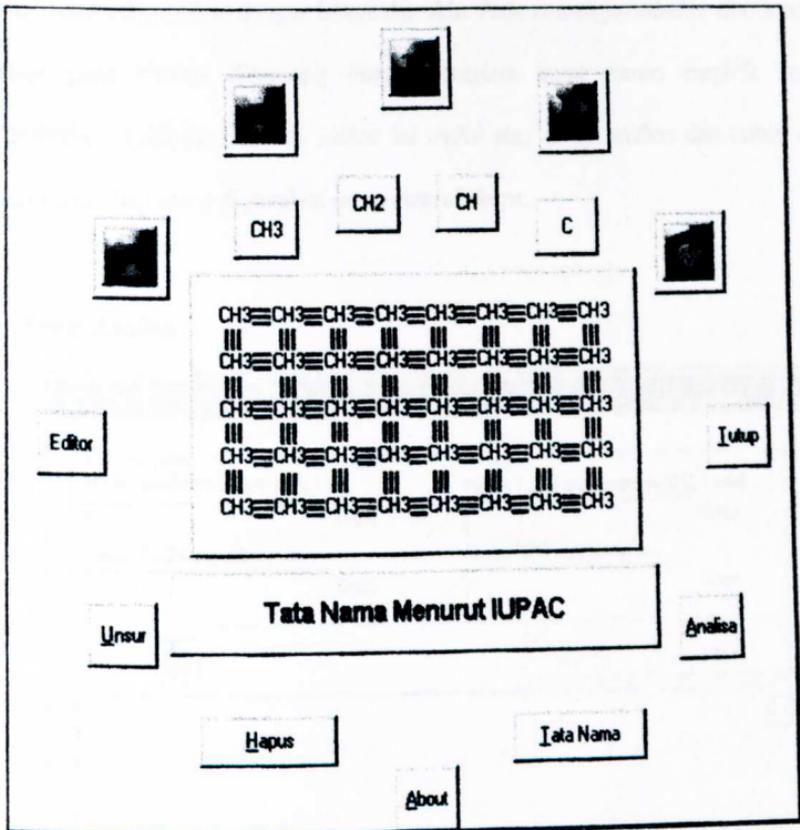
1. Kemampuan yang lebih untuk mempertemukan dengan kebutuhan pemakai secara keseluruhan.

2. Perkiraan waktu serta biaya dari pembuatan perangkat lunak dapat lebih mudah diperhitungkan
3. Memudahkan pengawasan dan pengendalian perangkat lunak
4. Memperkecil kesalahan dan permasalahan yang mungkin timbul selama pembuatan perangkat lunak.
5. Perencanaan ini bersifat fleksibel, karena tidak dipaksakan dari modul-modul, tetapi yang umum terlebih dahulu.
6. Merupakan pendekatan yang standard terhadap perancangan dan pembuatan program.

III. 1. 2. 1 Perancangan Interface

Perangkat lunak akan terdiri atas beberapa *window* yang tiap *window* akan terintegrasi dengan modul utama. Untuk merancang interface tiap *window* yang menarik maka perangkat lunak akan menggunakan *skin* yang penulis rancang dan dengan bantuan *Windows API* (Application and Programming Interface) untuk membuat tampilan *window* yang unik.

- Form Tata Nama



GAMBAR 3. 1 Layout Form Tata Nama

Pada saat *Run Time window* perangkat lunak tidak akan tampil berbentuk persegi seperti terlihat pada perancangan diatas, tetapi sesuai dengan image yang di-load pada picture box (sudut kiri atas form), hal ini dimungkinkan dengan bantuan fungsi grafik yang diberikan pada *Windows API Call Function*. Perangkat lunak akan dilengkapi dengan modul yang berisi deklarasi *API Call Function* dan *user define function* untuk menentukan *Window Region* mana yang diaktifkan.

Tipe Form adalah *1-None* yang berarti pada form tidak akan terdapat border, tetapi border digantikan dengan image dan skin. Pada rancangan *window* diatas terdapat *editor* yang khusus dirancang untuk menerima input rumus empirik senyawa hidrokarbon golongan alifatik. *Editor* ini terdiri atas *array textbox* dan untuk simbol ikatan antar tiap unsur digunakan *array control shape*.

- **Form Analisa**

The image shows a software window titled "Analisa Pembakaran Senyawa Hidrokarbon". Inside the window, there is a section labeled "Data Analisa" which contains four input fields arranged in a 2x2 grid. Each field is followed by the unit "Gram". The fields are: "Massa awal senyawa organik : Gram", "Massa 1 Liter uap senyawa (0 C, 1 atm) : Gram", "Massa CO2 yang dihasilkan : Gram", and "Massa H2O yang dihasilkan : Gram". Below the input fields, there are two buttons labeled "Analisa" and "Ulangi". At the bottom of the window, there is a large, empty text area with a vertical scroll bar on the right side.

GAMBAR 3. 2 Layout Form Analisa

Tipe form adalah 2-Fixed Dialog, sehingga *User* tidak dapat *meresize* form pada form terdapat empat *textbox* sebagai parameter input untuk melakukan fungsi analisa pembakaran senyawa hidrokarbon. Keempat *textbox* tersebut akan menerima nilai yang

sifatnya hanya numerik, jadi keempat textbox tersebut akan diberikan rutin yang akan mengizinkan *user* untuk menginput nilai numerik saja. Hal ini dilakukan untuk menghindari kesalahan pemasukan data.

Terdapat dua command button yang salah satunya akan digunakan untuk membangkitkan event perhitungan analisa. Hasil analisa akan tercetak pada picture box dan picture box dapat di-*scroll* dengan Vscroll disampingnya. Command Button kedua berupa tombol untuk mengulangi kembali proses analisa.

Untuk melakukan fungsi *scroll* ini, pada perancangan terdapat dua *picture box* dimana satu *picture box* merupakan *container* dari picture box lainnya. Dan nilai posisi *top* dari *picture box* kedua akan berubah sesuai dengan nilai *value* pada *Vscroll*.

- Form Periodik

Dalam Periodik Unsur

SUSUNAN BERKALA UNSUR-UNSUR

H																	He																												
Li	Be											B	C	N	O	F	Ne																												
Na	Mg											Al	Si	P	S	Cl	Ar																												
K	Ca	Sc	Ti	V	Cr	Mn	Fe	Co	Ni	Cu	Zn	Ga	Ge	As	Se	Br	Kr																												
Rb	Sr	Y	Zr	Nb	Mo	Tc	Ru	Rh	Pd	Ag	Cd	In	Sn	Sb	Te	I	Xe																												
Cs	Ba	La	Hf	Ta	W	Re	Os	Ir	Pt	Au	Hg	Tl	Pb	Bi	Po	At	Rn																												
Fr	Ra	Ac	Ku	Ha																																									
<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tbody> <tr> <td>Ce</td><td>Pr</td><td>Nd</td><td>Pm</td><td>Sm</td><td>Eu</td><td>Gd</td><td>Tb</td><td>Dy</td><td>Ho</td><td>Er</td><td>Tm</td><td>Yb</td><td>Lu</td> </tr> <tr> <td>Th</td><td>Pa</td><td>U</td><td>Np</td><td>Pu</td><td>Am</td><td>Cm</td><td>Bk</td><td>Cf</td><td>Es</td><td>Fm</td><td>Md</td><td>No</td><td>Lw</td> </tr> </tbody> </table>																		Ce	Pr	Nd	Pm	Sm	Eu	Gd	Tb	Dy	Ho	Er	Tm	Yb	Lu	Th	Pa	U	Np	Pu	Am	Cm	Bk	Cf	Es	Fm	Md	No	Lw
Ce	Pr	Nd	Pm	Sm	Eu	Gd	Tb	Dy	Ho	Er	Tm	Yb	Lu																																
Th	Pa	U	Np	Pu	Am	Cm	Bk	Cf	Es	Fm	Md	No	Lw																																

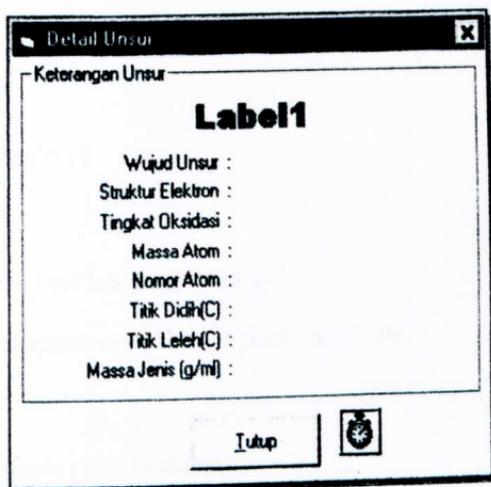


GAMBAR 3.3 Layout Form Periodik

Form bertipe 2-Fixed dialog, dan pada form terdapat kumpulan array command button dimana tiap command button tersebut diatas disusun berdasarkan nomor indeks tiap senyawa. Hal ini dilakukan untuk mempermudah dalam pemrograman kelak. Tiap indeks akan mewakili tiap unsur.

Bila salah satu command button ditekan maka dicari indeksnya dan indeks tersebut dikirim sebagai parameter untuk menentukan deskripsi dari unsur yang dicari lalu deskripsi dari unsur tersebut akan dikirim sebagai parameter ke form details untuk menampilkan detail mengenai unsur yang dimaksud.

- **Form Details**



GAMBAR 3. 4 Layout Form Details

Form Details bertipe 2-Fixed Dialog, form akan menampilkan semua keterangan mengenai unsur terpilih pada form periodik berdasarkan parameter yang dikirim ketika pemanggilan form.

III. 1. 2. 2 Algoritma Program

Algoritma sistem diwakili dalam bentuk *pseudocode* sehingga dapat mewakili urutan-urutan proses yang ada dalam sistem. Disamping itu dengan menggunakan Pseudocode dapat menjembatani antara bahasa alami dengan bahasa komputer sehingga mudah untuk dipahami.

- **Algoritma Form Tata Nama**

Form Tata Nama memiliki editor yang dirancang khusus sebagai input untuk senyawa hidrokarbon kompleks golongan alifatik jenuh dimana golongan ini memakai ikatan rangkap tunggal.

Algoritma Editor :

Mulai

Set Indeks Awal = 17

Awal :

Input pada array text boxt sesuai indeks

Validasi input apakah merupakan senyawa hidrokarbon

Jika tidak

Munculkan pesan kesalahan

Jika Ya

Simpan dalam array sesuai indeks textbox

Tunggu event tombol navigasi array

Jika Panah Kanan

Fokus ke textbox dengan indeks = indeks + 1

Tampilkan ikatan

Jika Panah Kiri

Fokus ke textbox dengan indeks = indeks - 1

Tampilkan Ikatan

Jika Panah Atas

Fokus ke textbox dengan indeks = indeks - 8

Tampilkan Ikatan

Jika Panah Bawah

Fokus ke textbox dengan indeks = indeks + 8

Tampilkan Ikatan

Kembali ke Awal

Selesai

Algoritma Tata Nama

Mulai

Validasi semua array pada textbox

Simpan dalam array semua nilai yang ada pada array text box

Panggil fungsi TataNama

Simpan Nilai kembali TataNama ke variable

Tulis Pada Label Tata Nama

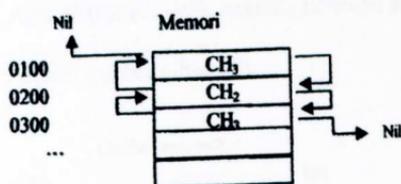
Selesai

Algoritma Fungsi Tata Nama

Sesuai kaidah yang ada pada tata cara penamaan menurut IUPAC maka algoritma untuk fungsi Tata Nama dipecah atas beberapa sub Fungsi yaitu :

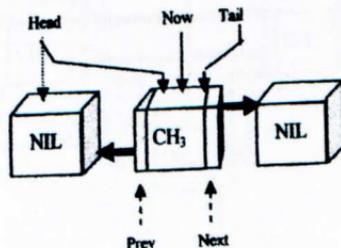
Algoritma Cari Rantai Terpanjang

Tiap nilai pada text box akan disimpan didalam stack dengan urutan stack sesuai indeks. Dan tiap stack juga akan memiliki 4 node dimana tiap node akan mengacu ke stack lain.



GAMBAR 3. 5 Struktur Stack Pada Algoritma Fungsi Tata Nama

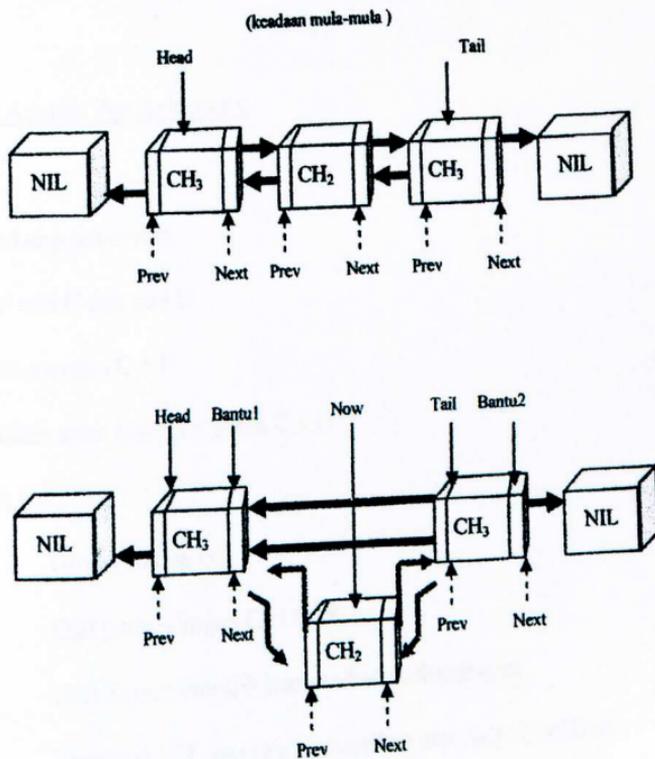
Proses pencarian rantai atom terpanjang adalah dengan mencari semua kemungkinan yang ada pada tiap cabang dengan node terhubung hanya satu. Hasil path yang dicari akan disimpan didalam stack baru dengan proses insert after

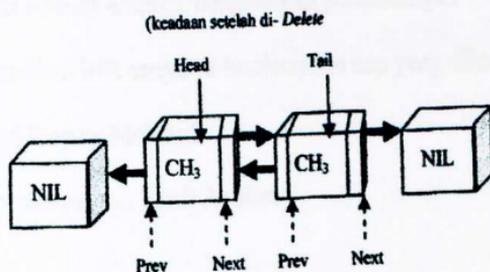


GAMBAR 3. 6 Insert After Pada Algoritma Fungsi Tata Nama

Algoritma Penomoran Rantai dan Gugus Alkil

Path pada stack akan diretrieve sesuai prinsip FIFO. Tiap elemen pada stack akan dihitung berapa panjangnya dan elemen terpanjang akan menjadi rantai atom induk. Setelah mencari rantai atom induk maka menentukan penomoran apakah dari kanan atau dari kiri. Untuk ini semua path kembali diuji agar memenuhi aspek sesuai standart IUPAC. Data akan diretrieve dari stack dan dicari gugusan alkilnya dan tiap data pada stack yang telah dicari gugus alkilnya akan dilakukan penghapusan agar tidak terjadi *unterminated looping*. Agar tiap stack tidak terputus nodenya maka penghapusan pada stack memiliki langkah seperti gambar di bawah.





GAMBAR 3. 7 Metode Penghapusan Stack

Setelah semua stack berhasil dicari gugus alkilnya maka string yang dipulangkan pada fungsi merupakan koleksi string yang ada pada stack.

Algoritma Analisa Pembakaran :

Mulai

Input tiap parameter

Cari mol C dan mol H

Hitung massa C + H

Apakah masa sampel < Masa C + H

Jika Ya

Hitung Massa O

Cari perbandingan C : O : H

Cari Rumus empirik berdasarkan perbandingan

Tentukan MR senyawa berdasarkan uap yang dihasilkan

Cari Rumus Molekul

Jika tidak

Cari perbandingan C : O

Cari Rumus empirik berdasarkan perbandingan

Tentukan MR senyawa berdasarkan uap yang dihasilkan

Cari Rumus Molekul

Tulis pada picture box Hasil Analisa

Selesai

Algoritma Form Periodik :

Mulai

Tunggu Event penekanan Tombol

Cari indeks dari tombol yang ditekan

Panggil form detail

Kirim indeks ke form detail

Selesai

Algoritma Form Details :

Mulai

Cari keterangan unsur sesuai parameter indeks

Tampilkan keterangan

Selesai

BAB IV

IMPLEMENTASI PROGRAM

IV. 1 Implementasi Sistem Perangkat Lunak

Setelah melalui tahap perancangan perangkat lunak maka kini perangkat lunak tersebut telah siap untuk diimplementasikan ke sistem. Hasil dari implementasi ini akan digunakan untuk menguji apakah perangkat lunak yang dibuat sudah sesuai dengan sistem yang telah dirancang pada sebelumnya.

IV. 1. 1 Pembuatan Perangkat Lunak

Pembuatan perangkat lunak berbasis Windows melalui tiga tahapan yaitu perancangan *interface*, pengaturan properties dari tiap objek kontrol dan penulisan kode program sesuai dengan sistem yang telah dirancang sebelumnya. Untuk membuat sistem perangkat lunak ini menggunakan software pendukung yaitu :

1. Microsoft Visual Basic 6.0

Bahasa pemrograman ini dipergunakan dalam pembuatan perangkat lunak karena memiliki pustaka yang lengkap dalam mengakses sistem dan terintegrasi dengan baik pada sistem operasi berbasis Windows serta memiliki compiler andal yang dapat menghasilkan file executable yang lebih cepat dan efisien dari sebelumnya.

2. Sistem Operasi Microsoft Windows Versi 9X dan NT

Sistem operasi merupakan syarat mutlak untuk dapat menjalankan atau menginstalasi program yang dirancang, tepatnya dalam lingkungan sistem

operasi berbasis Windows baik windows versi 9x (95, 98 dan Me) serta windows versi NT (200 dan XP).

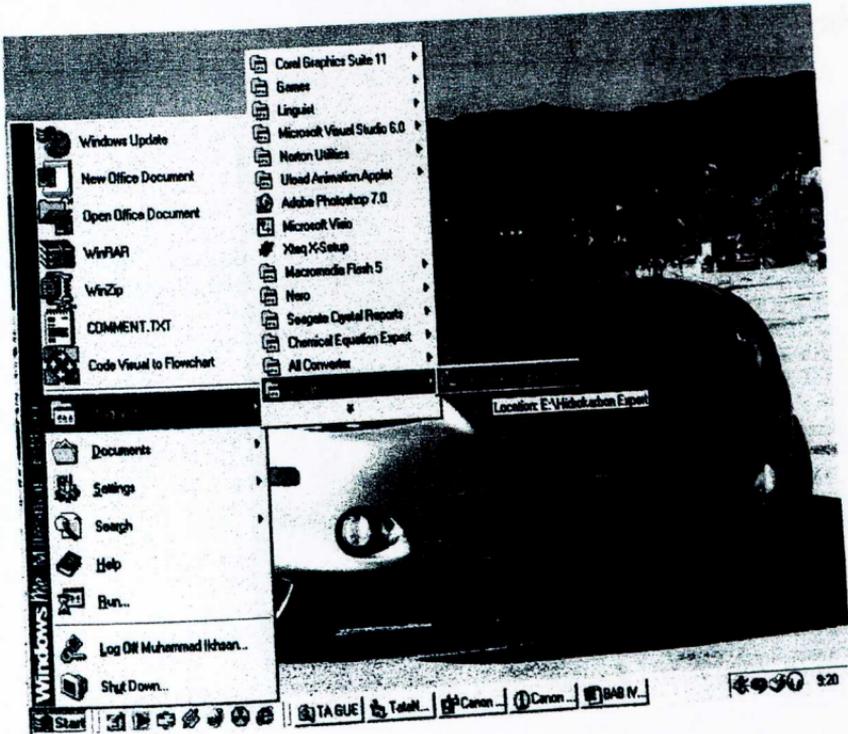
IV. 1. 2 Spesifikasi Hardware

Sistem perangkat lunak Hidrokarbon Expert dapat dijalankan dengan baik apabila telah dilakukan beberapa hal, yaitu proses instalasi perangkat lunak ke sistem operasi serta hardware harus memiliki spesifikasi yang mendukung dalam menjalankan perangkat lunak ini. Spesifikasi sistem hardware minimal yang dibutuhkan untuk mengimplementasikan sistem agar dapat berjalan dengan baik adalah sebagai berikut :

1. Komputer dengan processor Intel Pentium II ke atas.
2. Memory dengan SDR 64 Mb.
3. Harddisk 5 Gb.
4. VGA card 2 Mb dengan SVGA Monitor.
5. Mouse dan Keyboard.

IV. 1. 3 Membuka Program Aplikasi

Setelah program selesai diinstall kedalam sistem operasi maka tahap selanjutnya adalah menjalankan program tersebut dengan cara klik "Start Menu", "Programs" pilih "Organik" lalu klik "Hidrokarbon Expert". Untuk lebih jelasnya dapat dilihat gambar dibawah ini. Atau dapat menjalankannya dari *shortcut* yang ada di *desktop* yang secara otomatis akan dibuat ketika program di-install atau dapat pula melalui kotak dialog Menu Run, lalu pada kotak isiannya isikan path "c:\Program Files\Organik\Hidrokarbon Expert.



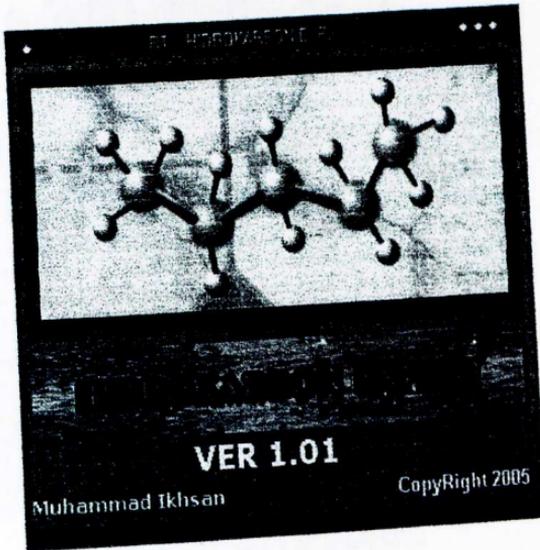
GAMBAR IV. 1 Cara Menjalankan Program

IV. 1. 4 Mengoperasikan Program Aplikasi

Setelah proses menjalankan program sukses maka program aplikasi akan terbuka seperti gambar IV.3, tetapi pertama sekali akan muncul *Message Box* berupa pesan ucapan Terima Kasih telah menggunakan perangkat lunak tersebut. Kemudian akan muncul *Splash Screen* yang berisikan animasi program atau terlihat pada gambar IV. 2.

Pada awal pertama sekali program dijalankan program akan menampilkan Form Tata Nama. Pada form ini terdapat beberapa command button yang memiliki fungsi untuk memanggil form lainnya, yaitu form analisa dan form periodik. Ketika form

muncul form tidak akan menampilkan editor sebelum user mengklik tombol toggle Editor.

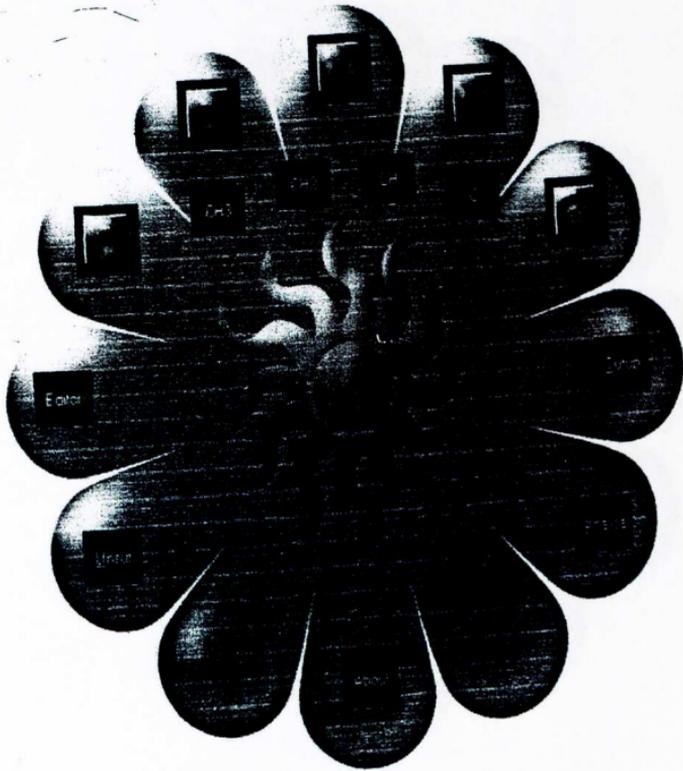


GAMBAR IV. 2 Splash Screen

IV. 4. 1 Form Tata Nama

Form ini merupakan form pertama yang ditampilkan oleh perangkat lunak. Pada form untuk menampilkan Editor hidrokarbon maka kita dapat meng-klik tombol Editor. Tombol editor bersifat Toggle button yang artinya untuk menampilkan atau menyembunyikan editor cukup mengklik tombol tersebut.

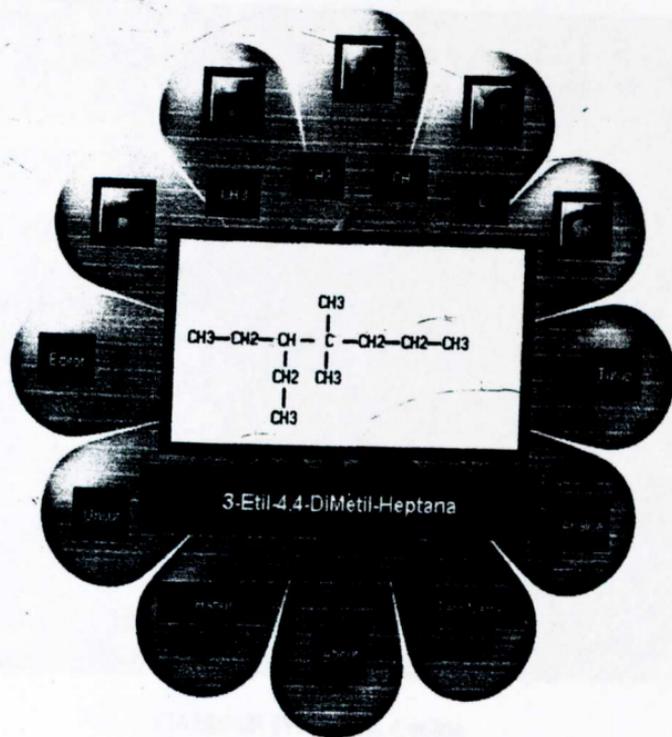
Untuk memanggil form Analisa kita dapat mengklik tombol Analisa maka form Analisa akan tampil Untuk memanggil form Periodik, maka kita dapat meng-klik tombol Unsur. Tombol About berfungsi untuk menampilkan Window About Program yang berisi deskripsi ringkas mengenai program dan pembuat program.



Gambar IV. 3 Form Tata Nama

IV. 4. 2 Menjalankan Editor

Untuk menjalankan editor klik tombol editor maka form tata nama akan terlihat seperti pada gambar IV. 4. Isikan pada kotak isian senyawa hidrokarbon lalu klik tombol navigasi panah atas, panah bawah, panah kiri dan panah kanan untuk berpindah ke textbox berikutnya. Untuk menghapus senyawa klik tombol undo. Kita dapat mengetikkan secara langsung senyawa atau dengan mengklik tombol cepat senyawa pada form yaitu CH_3 , CH_2 , CH dan C maka pada text box akan tertulis caption tombol tersebut. Klik tombol Tata Nama untuk melihat tata nama dari senyawa hidrokarbon yang kita input atau klik tombol hapus untuk menghapus editor.



GAMBAR IV. 4 Editor Form Tata Nama

IV. 4. 3 Form Analisa

Klik tombol Analisa untuk menampilkan form Analisa, pada form analisa isikan kotak isian data analisa sesuai dengan hasil percobaan. Lalu klik tombol Analisa untuk melihat hasil dari analisa program terhadap masukan tersebut. Pada picture box akan tampil hasil analisa dan kemungkinan pembakaran senyawa hidrokarbon. Klik tombol scroll untuk menggulung layar pada picture box hasil analisa . Untuk mengulangi proses klik tombol ulangi .

Detail Unsur

SUSUNAN BERKALA UNSUR-UNSUR

H																	He	
Li	Be											B	C	N	O	F	Ne	
Na	Mg											Al	Si	P	S	Cl	Ar	
K	Ca	Sc	Ti	V	Cr	Mn	Fe	Co	Ni	Cu	Zn	Ga	Ge	As	Se	Br	Kr	
Rb	Sr	Y	Zr	Nb	Mo	Tc	Ru	Rh	Pd	Ag	Cd	In	Sn	Sb	Te	I	Xe	
Cs	Ba	La	Hf	Ta	W	Re	Os	Ir	Pt	Au	Hg	Tl	Pb	Bi	Po	At	Rn	
Fr	Ra	Ac	Th	Pa														Ra
		Ce		Pr	Nd	Pm	Sm	Eu	Gd	Tb	Dy	Ho	Er	Tm	Yb	Lu		
		Th		Pa	U	Np	Pu	Am	Cm	Bk	Cf	Es	Fm	Md	No	Lr		

Tutup

GAMBAR IV. 6 Form Analisa

Detail Unsur

Peterangan Unsur

MANGAN

Wujud Unsur : Padat
 Struktur Elektron : (Ar)-3d⁵4s²
 Tingkat Oksidasi : 7
 Massa Atom : 54,938
 Nomor Atom : 25
 Titik Didih(C) : 2150
 Titik Leleh(C) : 1245
 Massa Jenis (g/ml) : 7,43

Tutup

GAMBAR IV. 7 Form Details

BAB V

KESIMPULAN DAN SARAN

V. 1 KESIMPULAN

Berdasarkan uraian dan pembahasan yang telah penulis lakukan, dapat ditarik beberapa kesimpulan yang berkaitan dengan perangkat lunak yang telah dirancang yaitu:

1. Dari segi kuantitas, jumlah senyawa organik lebih banyak daripada senyawa anorganik dan kehidupan manusia sebagai salah satu organisme bisa dikatakan tidak bisa terlepas dari pemakaian senyawa-senyawa organik, maka senyawa karbon yang disebut senyawa organik itu terus diusahakan untuk dibuat dan ditiru dilaboratorium.
2. Perangkat Lunak yang dirancang merupakan suatu sistem perangkat lunak yang dapat menentukan tatanama senyawa hidrokarbon serta menganalisa dan mensimulasikan kemungkinan hasil reaksi pembakaran dari senyawa hidrokarbon berdasarkan sifat serta karakteristik yang dimiliki tiap senyawa tersebut sebelum dilakukan pengujian lebih lanjut di laboratorium, untuk mengefisiensikan biaya, waktu serta tenaga.

V. 2 SARAN

Adapun saran-saran yang dapat diberikan di dalam pengembangan program ini kedepannya adalah :

1. Perangkat lunak yang dirancang masih terbatas kepada senyawa Hidrokarbon golongan alifatik jenuh, Untuk itu diperlukan pengembangan program lebih

lanjut agar dapat membahas secara luas semua senyawa hidrokarbon baik golongan siklik maupun aromatik.

2. Analisa masih terbatas kepada reaksi pembakaran hidrokarbon, proses analisa cracking, substitusi dan lain-lain belum tercakup pada perangkat lunak yang telah dirancang, diperlukan pengembangan perangkat lunak lebih lanjut agar proses analisa tersebut diatas dapat dilakukan secara komputerisasi juga.
3. Galat dari hasil analisa perangkat lunak belum memiliki presisi yang sangat tepat, masih banyak aspek yang mempengaruhi sistem manual yang diabaikan pada sistem komputerisasi. Untuk itu pengembangan program kearah sistem pakar membutuhkan analisa lebih lanjut dengan menambah kuantitas percobaan di laboratorium.

DAFTAR PUSTAKA

1. Halvorson, Michael. 1998. Microsoft Visual Basic Profesional (Step by Step). Jakarta. PT. Elex Media Komputindo.
2. Kurniadi, Adi. 1999. Pemrograman Microsoft Visual Basic 6.0. Jakarta. P.T Elex Media Komputindo.
3. Perry, Greg. 1997. Sams' Teach Yourself Visual Basic 5 in 24 Hours. Indianapolis. Macmillan Computer Publishing.
4. Pramono, Djoko. 2000. Mudah Menguasai Visual Basic 6.0. Jakarta. P.T Elex Media Komputindo.
5. Purba, Michael dan Hadi, Sarwono. 1990. Ilmu Kimia Karbon. Jakarta. Penerbit Erlangga.
6. R. Hadimoelyono dan Soetikno. 1979. Petunjuk Praktek Kimia Analisa. Jakarta. Departemen Pendidikan dan Kebudayaan
7. Sanjaya, Dwi. 2001. Berpertualang Dengan Struktur Data. Yogyakarta. J & J Learning.
8. Wanto, E. P. dan Soebagyo, Arief. 1979. Proses Industri Kimia. Jakarta. Departemen Pendidikan dan Kebudayaan.

Berbagai Sumber dari Internet, baik artikel, jurnal dan lain-lain.

