

D/IMT
2004
082

**APLIKASI PROGRAM LINEAR DENGAN
MICROSOFT VISUAL BASIC 6.0[®] DALAM
FORMULASI RANSUM UNGGAS**

SKRIPSI

Budi Aprilianto Kusnandar



**PROGRAM STUDI NUTRISI DAN MAKANAN TERNAK
DEPARTEMEN ILMU NUTRISI DAN MAKANAN TERNAK
FAKULTAS PETERNAKAN
INSTITUT PERTANIAN BOGOR
2004**

RINGKASAN

Budi Aprilianto Kusnandar. D02400071. 2004. **Aplikasi Program Linear dengan Microsoft Visual Basic 6.0® dalam Formulasi Ransum Unggas.** Skripsi. Program Studi Nutrisi dan Makanan Ternak. Departemen Ilmu Nutrisi dan Makanan Ternak. Fakultas Peternakan. Institut Pertanian Bogor.

Pembimbing Utama : Dr. Ir. Idat Galih Permana, M.Sc.
Pembimbing Anggota : Ir. Julio Adisantoso, M.Kom.
Ir. Widya Hermana, M.Si.

Biaya pakan merupakan biaya produksi terbesar dalam suatu usaha peternakan unggas yang dapat mencapai 68-70 % dari biaya produksi. Oleh karena itu dalam usaha untuk meningkatkan pendapatannya, peternak dituntut untuk mampu menyusun suatu formula ransum sendiri yang lebih ekonomis tanpa mengabaikan faktor kebutuhan nutrisi ternak.

Formulasi tersebut dapat dilakukan dengan berbagai metode, seperti metode coba-coba (*Trial and Error Method*), Metode Bujur Sangkar (*Square Method*), metode Program linear dan berbagai metode lain yang tujuannya sama yaitu untuk mendapatkan suatu komposisi ransum yang sesuai dengan kebutuhan ternak. Untuk mendapatkan suatu ransum yang ekonomis dan memenuhi kebutuhan ternak, maka metode program linear dapat digunakan sebagai metode formulasi. Akan tetapi, untuk penyelesaian secara manual, metode ini dirasa sangat rumit. Untuk mempermudahnya, metode ini dapat diterapkan dalam bentuk suatu program komputer.

Penelitian ini bertujuan untuk membuat suatu program komputer yang merupakan aplikasi dari metode program linear dalam mendapatkan suatu formula ransum unggas dengan bahasa pemrograman Microsoft Visual Basic 6.0®.

Penelitian ini dilaksanakan di Laboratorium Komputer Departemen Ilmu Nutrisi dan Makanan Ternak, Fakultas Peternakan, IPB yang dilaksanakan pada bulan Juni 2003 sampai dengan bulan Oktober 2004.

Penelitian ini dilakukan dengan 3 tahap. Tahap pertama adalah pembuatan model matematis formula ransum menggunakan metode program linear dengan fungsi tujuan minimalisasi biaya pakan. Secara umum fungsi tujuan dari program linear tersebut sebagai berikut :

$$Z_{\min} = C_1X_1 + C_2X_2 + C_3X_3 + \dots + C_nX_n$$

Dengan fungsi kendala :

$$\begin{aligned} A_{11}X_{11} + A_{12}X_{12} + A_{13}X_{13} + \dots + A_{1n}X_{1n} &\geq b_1 \\ A_{21}X_{21} + A_{22}X_{22} + A_{23}X_{23} + \dots + A_{2n}X_{2n} &\geq b_2 \end{aligned}$$

$$\vdots$$

$$A_{m1}X_{m1} + A_{m2}X_{m2} + A_{m3}X_{m3} + \dots + A_{mn}X_{mn} \geq b_m$$

Dengan Asumsi :

$$X_1, X_2, X_3, \dots, X_n \geq 0$$

Dimana :

Z_{\min} : fungsi tujuan (meminimalkan harga ransum)

C : harga bahan

X : bahan pakan

A : kandungan nutrisi bahan

b : kebutuhan nutrisi

Tahap kedua adalah merancang model yang akan ditampilkan dan diikuti dengan pembuatan program dengan menggunakan sistem pemrograman Microsoft Visual Basic 6.0® yang bersifat *Object Oriented Programming* (OOP). Untuk membuat *database* komposisi bahan pakan dan standar kebutuhan nutrisi digunakan Microsoft Access '97 (*.MDB). Data bahan pakan dan standar kebutuhan nutrisi unggas tersebut bersifat dinamis dan dapat diperbaiki oleh pengguna dengan mudah. Pada tahap akhir dilakukan pengujian program dan akhirnya dikompilasi dalam bentuk aplikasi (*.EXE) yang dapat digunakan pada komputer lain. Hasil formulasi dapat disimpan dalam bentuk file data (*.AAN) sehingga dapat dicetak maupun dilakukan formulasi ulang. Hasil pengujian menunjukkan bahwa program ini hasilnya dapat menyamai program formulasi lain seperti Feed Mania. Bahkan dalam pengujian, program ini dapat menghasilkan ransum dengan harga yang lebih murah. Program formulasi ransum ini diberi nama "EzFeed versi 1.0"

Kata kunci : formulasi ransum, ransum unggas, program linear, microsoft visual basic 6.0®

SUMMARY

Budi Aprilianto Kusnandar. D02400071. 2004. **Application of Linear Programming with Microsoft Visual Basic 6.0 for Poultry Feed Formulation.** Thesis. Study Program of Nutrition and Feed Science. Faculty of Animal Science. Bogor Agricultural University.

Advisor : Dr. Ir. Idat Galih Permana, M.Sc.
Co-Advisor : Ir. Julio Adisantoso, M.Kom.
Ir. Widya Hermana, M.Si.

Feed cost is the highest production cost in a poultry production which can reach up to 68-70 % of the production cost. Therefore an order to increase their benefit, farmers should be able to formulate economic ration.

The feed formulation can be using various methods, such as Trial and Error Method, Square Method, Linear Programming Method, and various others. To make economic ration that fulfill livestock requirement, the linear program method can be used for feed formulation. However, to solved linear programming manually is very difficult, therefore an application computer program can be used to solve this problem.

The objective of this experiment was to design an application computer program using linear programming method for poultry ration formulation with Microsoft Visual Basic 6.0 programming language.

This experiment was conducted in the Computer Laboratory of Department of Nutrition and Feed Science, Faculty of Animal Science, Bogor Agricultural University from June 2003 until October 2004.

This experiment divided by 3 steps. The first step is to design the mathematical model of Linear Programming to minimize feed cost. In generally, the mathematical model is mentioned as following:

$$Z_{\min} = C_1X_1 + C_2X_2 + C_3X_3 + \dots + C_nX_n$$

With constraint function :

$$A_{11}X_{11} + A_{12}X_{12} + A_{13}X_{13} + \dots + A_{1n}X_{1n} \geq b_1$$

$$A_{21}X_{21} + A_{22}X_{22} + A_{23}X_{23} + \dots + A_{2n}X_{2n} \geq b_2$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots$$

$$A_{m1}X_{m1} + A_{m2}X_{m2} + A_{m3}X_{m3} + \dots + A_{mn}X_{mn} \geq b_m$$

With Assumption :

$$X_1, X_2, X_3, \dots, X_n \geq 0$$

Which :

Z_{\min} : main function (to minimize ration cost)

C : feed material cost

X : feed material

A : nutrient of feed material

b : nutrient requirement

The second step is to design the mathematical model by using Microsoft Visual Basic 6.0[®] programming language. The feed ingredient database was created using Microsoft Access '97. The database can be updated by the user dynamically. The last step is to evaluate and compiled in to an executed application program. The result of formulation can be saved in data file (*. AAN), and it can be printed or reformulated. This Feed Formulation Program called " EzFeed Ver. 1.0".

Keywords: feed formulation, poultry ration, linear programming, microsoft visual basic 6.0[®]

**APLIKASI PROGRAM LINEAR DENGAN
MICROSOFT VISUAL BASIC 6.0® DALAM
FORMULASI RANSUM UNGGAS**

**Budi Aprilianto Kusnandar
D02400071**

**Skripsi ini merupakan salah satu syarat untuk
memperoleh gelar Sarjana Peternakan
pada Fakultas Peternakan
Institut Pertanian Bogor**

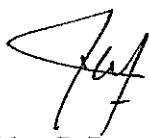
**PROGRAM STUDI NUTRISI DAN MAKANAN TERNAK
DEPARTEMEN ILMU NUTRISI DAN MAKANAN TERNAK
FAKULTAS PETERNAKAN
INSTITUT PERTANIAN BOGOR
2004**

**APLIKASI PROGRAM LINEAR DENGAN
MICROSOFT VISUAL BASIC 6.0® DALAM
FORMULASI RANSUM UNGGAS**

Oleh
Budi Aprilianto Kusnandar
D02400071

Skripsi ini telah disetujui dan disidangkan di hadapan
Komisi Ujian Lisan pada Tanggal 14 Desember 2004

Pembimbing Utama



Dr. Idat G. Permana, M.Sc.

Pembimbing Anggota



Ir. Julio Adisantoso, M.Kom

Pembimbing Anggota



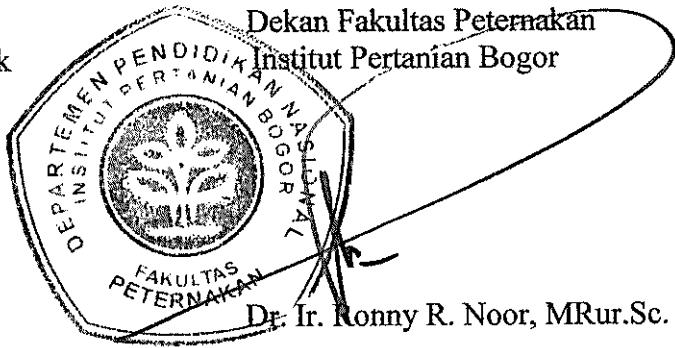
Ir. Widya Hermana, M.Si.

Ketua Departemen
Ilmu Nutrisi dan Makanan Ternak
Fakultas Peternakan
Institut Pertanian Bogor



Dr. Ir. M. Ridla, M.Agr.

Dekan Fakultas Peternakan
Institut Pertanian Bogor



Dr. Ir. Ronny R. Noor, MRur.Sc.

RIWAYAT HIDUP

Penulis dilahirkan di Desa Masaran, Kecamatan Masaran, Kabupaten Sragen, Propinsi Jawa Tengah pada tanggal 2 April 1981. Penulis merupakan anak pertama dari tiga bersaudara dari pasangan Bapak Budiyono dan Ibu Siti Kusiyah.

Penulis lulus dari jenjang pendidikan dasar di SDN 1 Masaran pada tahun 1988 dan kemudian dilanjutkan di pendidikan menengah SLTPN 1 Sragen sampai dengan tahun 1997. Penulis menamatkan pendidikan menengah atas di SMUN 3 Sragen pada tahun 2000 dan pada tahun yang sama penulis diterima di Institut Pertanian Bogor (IPB) melalui jalur USMI.

Setelah menjalani Tingkat Persiapan Bersama (TPB) IPB 2000-2001, penulis terdaftar sebagai mahasiswa Fakultas Peternakan pada tahun 2001 dan sebagai mahasiswa Departemen Ilmu Nutrisi dan Makanan Ternak pada tahun 2002.

Selama menjalani pendidikan di IPB, penulis pernah aktif di organisasi KOPMA IPB dan beberapa kegiatan lainnya seperti Gelar Ilmiah IPB 2001 dan Seminar Nasional Pengembangan Peternakan Berwawasan lingkungan pada tahun 2004.

Pada tahun 2003-2004, penulis aktif sebagai asisten mahasiswa pada Laboratorium Komputer Departemen Ilmu Nutrisi dan Makanan Ternak untuk mata ajaran Aplikasi Komputer dalam Ilmu Nutrisi dan Makanan Ternak. Selain itu, penulis juga pernah memberikan pelatihan penyusunan ransum dengan menggunakan program computer Feed Mania dan Feed Live.

KATA PENGANTAR

Assalamualaikum Wr. Wb.

Alhamdulillahi rabbil 'alamin, segala puji bagi Allah SWT yang telah memberikan rahmat, hidayah serta karunia-Nya sehingga tulisan ini bisa diselesaikan. Skripsi ini disusun dalam rangka melengkapi tugas akhir akademik sebagai salah satu syarat meraih gelar Sarjana Peternakan (S.Pt.) dari Fakultas Peternakan, Institut Pertanian Bogor.

Pada kesempatan kali ini, penulis ucapan terima kasih sedalam-dalamnya kepada Ir. Lily Aboenawan yang telah menjadi pembimbing akademik selama penulis belajar di Fakultas Peternakan IPB. Selanjutnya penulis ucapan terima kasih kepada Dr. Ir. Idat Galih Permana, M.Sc., Ir. Julio Adisantoso, M.Kom. dan Ir. Widya Hermana, M.Si. atas bimbingan, arahan dan dorongannya dalam menyelesaikan tulisan ini serta kepada Dr.Ir. Ibnu Katsir Amrullah, M.Sc. selaku dosen penguji seminar, Prof. Dr. Ir. Wiranda G. Pilliang dan Ir. Burhanuddin, MM selaku dosen penguji ujian lisan, terima kasih atas masukkannya.

Penulis ucapan terima kasih yang sebesar-besarnya dan sedalam-dalamnya kepada bapak dan ibu yang telah membesar dan mendidik penulis sampai saat ini, semoga Allah senantiasa membalas dan memberikan perlindungan-Nya. Kepada dua orang adik tercinta Iin dan Ani terima kasih atas semangat dan doanya semoga Allah mengumpulkan kita sekeluarga ke dalam hamba-Nya yang sholeh.

Tidak terlupa kepada seluruh teman sepenelitian Piter Andre, teman-teman pemberi semangat Anto, Eko, Dinda, Wira, Atun, Isna, Triyadi, Chandra Bayu, Ahkam, Anjar dan Mbak Upi atas dorongannya semoga sukses dan keberhasilan selalu menyertai kita. Amin

Wassalamu'allaikum Wr. Wb.

Bogor, Desember 2004

Penulis

DAFTAR ISI

	Halaman
RINGKASAN	i
SUMMARY	iii
RIWAYAT HIDUP.....	v
KATA PENGANTAR	vi
DAFTAR TABEL.....	ix
DAFTAR GAMBAR.....	x
DAFTAR LAMPIRAN.....	xi
PENDAHULUAN	
Latar Belakang	1
Tujuan	2
Manfaat	2
TINJAUAN PUSTAKA	
Ransum.....	3
Bahan Makanan ternak.....	3
Kebutuhan Nutrisi Ternak Unggas.....	3
Formulasi Ransum	4
Teknik Formulasi Ransum	4
Program Linear.....	5
Program Linear dalam Formulasi Ransum	6
Penyelesaian Program Linear dalam Program Komputer.....	7
Bahasa Pemrograman Microsoft Visual Basic [®]	7
MATERI DAN METODE	
Tempat dan Waktu	9
Materi	9
Metode.....	12
HASIL DAN PEMBAHASAN	
Metode Program Linear	15
Pembuatan Program	17
Pemasukan Data dalam Perhitungan.....	18
Manajemen Basis Data.....	20
Pengujian Program	22
Faktor Pembatas	22
Fasilitas dan Tampilan Program.....	24

Formulasi Ransum Secara <i>Least Cost</i>	32
Formulasi Ransum Secara <i>Feed Mix</i>	36
KESIMPULAN DAN SARAN	
Kesimpulan	37
Saran.....	37
DAFTAR PUSTAKA	38
LAMPIRAN	40

DAFTAR TABEL

Nomor	Halaman
1. Standar Kebutuhan Nutrisi Unggas (NRC '94)	10
2. Data Komposisi Beberapa Bahan Pakan Berdasarkan NRC (1994).....	11
3. Bentuk Umum Iterasi Program Linear Fase I	15
4. Bentuk Umum Iterasi Program Linear Fase II.....	16
5. Struktur Basis Data FEED.MDB	21

DAFTAR GAMBAR

Nomor	Halaman
1. Skema Pembuatan Program Formulasi Ransum.....	12
2. Skema Pembuatan Program	18
3. Tampilan <i>Form</i> Pembuka.....	24
4. Tampilan dari <i>Form</i> Utama.....	26
5. Tampilan dari <i>List</i> Bahan Terpilih.....	26
6. Tampilan dari <i>Form</i> Edit Data Nutrisi.....	27
7. Tampilan dari <i>Form</i> Edit Data Bahan Pakan.....	28
8. <i>Form</i> Batasan Penggunaan Bahan	29
9. <i>Form</i> Batasan Nutrisi	29
10. Tampilan dari <i>Form Least Cost</i>	30
11. Tampilan dari <i>Form Feed Mix</i>	31
12. Form Data Tersimpan	31
13. Contoh Formula Ransum Ayam Broiler dengan Menggunakan 7 Bahan Pakan.....	33
14. Standar Kebutuhan Nutrisi Ayam Broiler (NRC '94)	33
15. Hasil Formulasi Ransum Setelah Diubah Batasan Maksimalnya	34
16. Komposisi Ransum Hasil Formulasi dengan Feed Mania	34
17. Komposisi Nutrisi Hasil Formulasi dengan Feed Mania	35
18. <i>Form</i> Formulasi Ransum secara <i>Feed Mix</i>	36

DAFTAR LAMPIRAN

Nomor	Halaman
1. <i>Listing Program EzFeed V 1.0</i>	40
2. Petunjuk Penggunaan Program EzFeed V 1.0.	69

PENDAHULUAN

Latar Belakang

Dalam suatu usaha peternakan, efisiensi produksi merupakan salah satu faktor penentu keberhasilannya. Efisiensi produksi tersebut dapat diwujudkan dengan menggunakan pakan yang memiliki harga murah namun berkualitas. Biaya pakan merupakan sumber biaya produksi terbesar (70-80 %) dari keseluruhan biaya produksi dalam suatu usaha peternakan unggas.

Untuk mendapatkan suatu pakan yang murah dan berkualitas, diperlukan suatu teknik atau metode formulasi ransum yang mudah untuk digunakan, cepat, akurat dalam penentuan komposisi bahan (perhitungan) dan yang paling utama adalah memperhatikan faktor biaya pakan dalam perhitungannya. Metode formulasi tersebut adalah metode program linear. Selain metode program linear, ada beberapa metode lain yang dapat digunakan, antara lain metode Bujur Sangkar (*Square Method*), metode eliminasi Gauss Jordan, dan beberapa metode lain. Diantara metode-metode tersebut, metode program linear adalah yang paling sesuai untuk diterapkan sebagai metode formulasi ransum karena harga ransum dapat dimasukkan sebagai peubah (fungsi tujuan) dalam perhitungan, akan tetapi dalam perhitungannya secara manual metode ini masih dirasa sangat sulit. Oleh karena itu, diperlukan suatu bentuk penerapan metode ini agar dapat digunakan dengan mudah oleh peternak atau pengusaha pakan. Salah satu bentuk penerapan tersebut adalah dalam bentuk suatu program komputer yang lebih bersifat interaktif dan mudah untuk digunakan (*user friendly*).

Tujuan

Penelitian ini bertujuan untuk membuat suatu program komputer (*software*) formulasi ransum unggas yang menggunakan metode program linear sebagai metode perhitungan dasar. Program ini dibuat dengan menggunakan sistem pemrograman Microsoft Visual Basic 6.0[®] sehingga program yang dihasilkan akan mudah untuk digunakan.

Manfaat

Penelitian ini dapat menghasilkan *software* formulasi ransum ternak unggas yang berbasis Windows™ sehingga pengoperasiannya menjadi lebih mudah. *Software* ini diharapkan dapat memformulasikan ransum dengan harga semurah mungkin sehingga dapat menekan biaya produksi yang berasal dari pakan serta mampu meningkatkan efisiensi waktu dan biaya formulasi ransum unggas.

TINJAUAN PUSTAKA

Ransum

Ransum merupakan satu atau beberapa jenis pakan yang diberikan untuk memenuhi kebutuhan ternak selama sehari semalam (Ensminger, 1992). Rasyaf (1984) menyatakan bahwa bahan-bahan ransum harus memiliki syarat-syarat teknis yang dicerminkan dengan nilai gizi bahan-bahan makanan yang selaras dengan nilai gizi yang dibutuhkan ternak. Ketidakseimbangan akan menyebabkan terjadinya gangguan fisiologis pada unggas atau ternak lain yang mengkonsumsinya.

Bahan-Bahan Makanan Ternak Unggas

Menurut Rasyaf (1994), bahan makanan ternak terbagi atas: 1) Sumber Protein (protein hewani dan nabati), 2) Sumber Energi dan 3) Sumber Vitamin dan Mineral. Menurut Williamson dan Payne (1993), kebanyakan makanan ternak dapat dikelompokkan menjadi dua jenis, yaitu hijauan dan konsentrat. Secara umum, hanya ruminansia seperti kerbau, sapi, domba, kambing dan unta yang dapat memanfaatkan makanan hijauan dengan baik. Hal ini disebabkan ternak ruminansia berbeda dengan ternak non ruminansia seperti unggas. Ternak ruminansia mempunyai saluran pencernaan yang kompleks yang mampu mencerna hijauan yang berserat tinggi, sedangkan ternak non ruminansia seperti unggas tidak dapat mencerna serat dengan baik.

Kebutuhan Nutrisi Ternak Unggas

Unggas terdiri dari bermacam-macam jenis, karena itu kisaran tingkat energi yang dibutuhkan juga berbeda-beda. Kebutuhan unggas terhadap asam-asam amino, mineral, vitamin dan zat-zat nutrisi lain tentu berbeda pula untuk tiap jenis unggas (Williamson dan Payne, 1993). Kandungan zat-zat nutrisi dari bahan pakan dalam tubuh unggas akan dicerna, kemudian diserap oleh cairan tubuh dan disebarluaskan ke seluruh jaringan tubuh (Ensminger, 1992). Zat-zat nutrisi tersebut digunakan untuk pertumbuhan, penggemukan atau produksi telur yang juga dapat disebut sebagai kebutuhan produksi unggas.

Kebutuhan unggas terhadap suatu zat gizi tertentu ditentukan dengan cara mendapatkan jumlah minimum dari unsur tersebut untuk dapat diketahui fungsi fisiologis dan sifat ekonomis dari unsur tersebut (Ensminger, 1992).

Formulasi Ransum

Formulasi atau menyusun ransum adalah menyamakan kandungan nutrisi (protein, asam amino, energi, vitamin dan mineral) beberapa bahan pakan yang terpilih dengan kebutuhan nutrisi ternak. Formulasi ransum dilakukan agar ransum yang diberikan kepada ternak memenuhi kebutuhan zat-zat nutrisi dan sesuai dengan kemampuan konsumsinya (Rasyaf, 1994). Menurut Ensminger *et al.* (1990), hal yang perlu diketahui dalam menyusun ransum adalah pengetahuan tentang kebutuhan zat-zat makanan dari ternak yang bersangkutan. Dengan suatu metode tertentu kebutuhan ayam harus terpenuhi seoptimal mungkin.

Dalam menyusun suatu ransum, diperlukan beberapa informasi mengenai 1) kebutuhan nutrisi ternak, 2) bahan pakan yang tersedia, 3) jenis ransum, serta 4) konsumsi yang diharapkan. Menurut Ensminger (1992), sebelum melakukan formulasi ransum unggas dengan benar, perlu diketahui: 1) kebutuhan nutrisi dari unggas, yang sering disebut juga sebagai *feeding standards*; 2) ketersediaan, kandungan nutrisi dan harga dari bahan-bahan pakan; 3) palatabilitas dan sifat fisik dari bahan-bahan pakan; dan 4) kandungan unsur-unsur berbahaya yang juga menetukan kualitas dari bahan .

Teknik Formulasi Ransum

Menurut Rasyaf (1994), cara menyusun ransum yang dikenal antara lain :

1. Metode MPS (Metode Pendugaan Sederhana);
2. Metode persamaan simulat;
3. Metode linear programming; dan
4. Metode program tujuan ganda.

Selain metode-metode tersebut, juga dikenal metode-metode formulasi ransum lain seperti metode coba-coba (*Trial and Error Method*), metode Bujur Sangkar (*Square Method*) dan metode Matriks (Ensminger *et al.*, 1990).

Metode coba-coba dilakukan dengan cara mencoba-coba sehingga memerlukan pengalaman dan latihan yang cukup lama (Suharti, 1997). Selain itu, semakin banyak kendala nutrisi yang dimasukkan, maka akan semakin sulit dalam penyusunannya.

Metode bujur sangkar hanya dapat digunakan untuk menyusun ransum dengan satu kendala nutrisi saja dari dua jenis pakan yang diformulasikan (Suharti, 1997), sehingga metode ini hanya digunakan untuk mencukupi kebutuhan satu macam nutrisi saja, tanpa mempertimbangkan kebutuhan nutrisi lainnya.

Metode yang paling banyak digunakan dalam formulasi ransum adalah metode program linear. Metode ini mampu memperhitungkan banyak kendala nutrisi dan variabel bahan tanpa mengesampingkan faktor biaya ransum.

Program Linear

Metode program linear pertama kali dirumuskan oleh ahli matematika Rusia yang bernama L.V. Kantorovich dalam bukunya yang berjudul “The Mathematical Methods in The Organization and Planning of Production”, namun ide ini rupanya tidak berkembang di Rusia (Supranto, 1983). Pada tahun 1947 seorang ahli matematika dari Amerika Serikat bernama George B. Dantzing (Bapak *Linear Programming*) menemukan suatu cara untuk memecahkan persoalan linear programming tersebut dengan metode yang disebut “*simplex method*”. Metode ini mula-mula digunakan dalam bidang kemiliteran (untuk penyusunan strategi perang, persoalan *bombing pattern*) dan dalam bidang bisnis (Supranto, 1983).

Persoalan Linear Programming (L.P.) sendiri ialah suatu persoalan untuk menentukan besarnya nilai-nilai variabel sedemikian rupa sehingga nilai fungsi tujuan atau obyektif (*objective function*) yang linear menjadi *optimum* (maksimum atau minimum) dengan memperhatikan pembatasan-pembatasan yang ada yaitu pembatasan mengenai inputnya (Supranto, 1983). Kata *Programming* didefinisikan sebagai suatu perencanaan (*planning*) dan bukan sebagai “program” seperti halnya dalam istilah komputer (Optimization Technology Center, 2004).

Metode program linear dapat diterapkan pada berbagai bidang, seperti dalam bidang pertanian, pengairan, transportasi, kesehatan, manajemen produksi, pembangunan dan sebagainya. Metode ini banyak digunakan didasarkan pada

kelebihan-kelebihan yang dimiliki oleh metode ini. Kelebihan-kelebihan tersebut antara lain (Soekartawi, 1992):

- a. Mudah dilaksanakan, apalagi bila dibantu dengan komputer;
- b. Dapat menggunakan banyak variabel, sehingga berbagai kemungkinan untuk memperoleh pemanfaatan sumberdaya yang optimum dapat dicapai; dan
- c. Fungsi tujuan (*objective function*) dapat difleksibelkan sesuai dengan tujuan penelitian atau berdasarkan data yang tersedia. Misalnya bila ingin meminimumkan biaya atau memaksimumkan keuntungan dengan data yang terbatas.

Selain kelebihan, metode ini juga memiliki kelemahan. Kelemahan tersebut adalah bila alat bantu seperti komputer tidak tersedia, maka penggunaan banyak variabel dalam metode ini akan sulit untuk diselesaikan dan bahkan tidak mungkin untuk dikerjakan dengan cara manual saja.

Suatu persoalan program linear bisa diklasifikasikan menjadi tiga kategori yaitu tidak ada pemecahan yang fisibel (*there is no feasible solution*), ada pemecahan *optimum (maximum / minimum)* atau fungsi obyektif tidak ada batasnya (*unbounded solution*) (Supranto, 1983).

Program Linear dalam Formulasi Ransum

Program linear dapat digunakan untuk menentukan campuran makanan ternak yang efisien, praktis dan relatif mudah digunakan (Soekartawi, 1992). Sesuai dengan definisinya, program linear adalah suatu teknik untuk menentukan kombinasi terbaik diantara bahan makanan yang tersedia, yang mempunyai kandungan nutrisi dan harga yang berbeda, dalam rangka untuk mendapatkan ransum dengan harga yang serendah mungkin (Crampton dan Harris, 1969). Hasil dari formulasi tersebut tergantung pada nilai yang digunakan untuk: 1) kandungan nutrisi dan spesifikasi lainnya yang diperlukan dalam ransum, 2) komposisi nutrisi dari bahan makanan yang dipilih, dan 3) unit harga dari tiap bahan pakan yang digunakan.

Aplikasi program linear dalam formulasi ransum akan menggunakan banyak variabel, sehingga dalam aplikasinya diperlukan alat bantu komputer (Soekartawi, 1992).

Penyelesaian Program Linear dalam Program Komputer

Program komputer adalah suatu urutan instruksi untuk dilaksanakan oleh komputer secara berturutan (Wikipedia, 2004). Suatu program komputer terdiri dari beberapa instruksi yang dapat dikenali oleh komputer.

Penyelesaian program linear dengan bantuan paket komputer (*software*) diperlukan persiapan-persiapan yang terinci (Soekartawi, 1992). Persiapan-persiapan tersebut antara lain menetapkan fungsi tujuan dalam bentuk linier, menetapkan fungsi dalam setiap faktor pembatas dan menetapkan bahwa setiap koefisien adalah positif atau non negatif.

Bahasa Pemprograman Microsoft Visual Basic®

Visual Basic dikembangkan dari bahasa QuickBasic yang berjalan di atas sistem operasi DOS (Dewobroto, 2003). Visual Basic menggunakan bahasa BASIC (*Beginners All-purpose Symbolic Instruction Code*) yang sejak awal dirancang untuk mudah dipelajari. Versi awal diciptakan oleh Alan Cooper yang kemudian menjualnya ke Microsoft dan mengambil alih pengembangan produk dengan memberi nama sandi “Thunder”. Sampai saat ini, sudah dirilis beberapa versi dari Microsoft Visual Basic mulai dari versi 1.0 yang dirilis pada bulan Mei 1991, versi 2.0 pada Oktober 2002, versi 3.0 pada tahun 1993, versi 4.0 pada tahun 1995, versi 5.0 pada tahun 1997, versi 6.0 pada tahun 1998 dan yang terakhir 6.0 .NET pada tahun 2001 (Dewobroto, 2003).

Tidak seperti bahasa pemrograman lain, seperti pascal misalnya, di mana *programmer* harus menuliskan kode program untuk segala sesuatunya, Visual Basic mampu menambahkan sendiri sebagian kode program secara otomatis ke dalam program sehingga pekerjaan *programmer* semakin mudah (Pandia, 2002).

Kelebihan dari Visual Basic antara lain : 1) Kurva pembelajaran dan pengembangan lebih singkat/mudah dibandingkan bahasa pemrograman lain seperti C/C++, Delphi atau PowerBuilder sekalipun, 2) Visual Basic Application dapat diintegrasikan dengan program lainnya seperti Office (Word, Excel, Access, Power Point) dan beberapa software lainnya, 3) dapat membuat kontrol ActiveX sehingga dapat memakai fungsi-fungsi aplikasi lain yang mendukung teknologi tersebut seperti Word, Excel dan sebagainya. Selain itu, Visual Basic termasuk salah satu

sistem pemrograman yang bersifat *Object Oriented Programming* (konsep pemrograman yang memberikan perhatian utama pada obyek yang akan diolah oleh program dengan cara mengenali semua obyek yang akan diolah dan mencari kaitan antara satu obyek dengan lainnya). Akan tetapi, Visual Basic juga mempunyai kekurangan antara lain program ini bukan sepenuhnya bahasa pemrograman yang berorientasi obyek seperti C/C++ atau Borland Delphi (Dewobroto, 2003).

MATERI DAN METODE

Tempat dan Waktu

Penelitian ini dilaksanakan di Laboratorium Komputer Departemen Ilmu Nutrisi dan Makanan Ternak, Fakultas Peternakan, IPB pada bulan Juni 2003 sampai dengan bulan Oktober 2004.

Materi

Alat

Perangkat yang digunakan dalam penelitian ini antara lain sebuah komputer dengan spesifikasi sebagai berikut : Prosessor AMD Athlon XP 1700+, memori (RAM) 256 MB, Harddisk dengan kapasitas 40 Gigabyte, monitor 17“, serta ditunjang dengan printer Canon S200SPx dengan menggunakan sistem operasi Microsoft Windows XP®.

Sistem Pemprograman

Sistem pemrograman Microsoft Visual Basic 6.0® dengan sistem operasi dasar Microsoft Windows XP® digunakan dalam pembuatan program ini. Program ini nantinya dapat digunakan pada semua komputer dengan spesifikasi minimal: komputer dengan prosessor pentium 100, memori 32 *MegaByte* dengan sistem operasi Microsoft Windows ‘98 atau yang lebih baru. Selain itu, dalam pembuatan program ini juga digunakan program-program pendukung lainnya seperti :

- Adobe Photoshop 6.0 digunakan untuk *editing* gambar,
- Adobe Acrobat 6.0 untuk menguji hasil cetakan dalam file PDF,
- Snag-It versi 7.0 untuk memindai (*capture*) tampilan program,
- Icon Cool Editor 4.0 untuk membuat *icon-icon* yang digunakan dalam program,
- Macromedia Dreamweaver MX untuk membuat *file* bantuan dalam bentuk HTML,
- HTML Help Workshop untuk mengkompilasi file bantuan dari *file* HTML menjadi *file Help (*.chm)*,
- InstallShield Express V 5.0 + SP2 untuk membuat *file* instalasi.

Data Standar Kebutuhan Nutrisi

Data yang digunakan untuk mengisi *database* standar kebutuhan nutrisi merupakan data standar kebutuhan nutrisi menurut rekomendasi *National Research Council* 1994 (NRC '94) seperti yang disajikan pada Tabel 1.

Tabel 1. Standar Kebutuhan Nutrisi Unggas (NRC '94)

Jenis Unggas	Umur / Fase Produksi (Minggu)	Kebutuhan Nutrisi						
		Energi Metabolis (kkal/kg)	Protein Kasar (%)	Lemak Kasar (%)	Kalsium (%)	Phospor tersedia (%)	Asam Amino Lisin (%)	Asam Amino Metionin (%)
Ayam Petelur	0 – 6	2800	17	1,0	0,90	0,40	0,80	0,28
	6 – 12	2800	17	1,0	0,90	0,40	0,80	0,28
	12 – 18	2800	15	1,0	0,80	0,35	0,56	0,23
	18 – bertelur pertama	2850	16	1,0	1,80	0,35	0,49	0,21
Ayam Broiler	0 – 3	3200	23	1,0	1,00	0,45	1,10	0,50
	3 – 6	3200	20	1,0	0,90	0,35	1,00	0,38
	6 – 8	3200	18	1,0	0,80	0,30	0,85	0,32
Itik	0 – 2	2900	22	-	0,65	0,40	0,90	0,40
	2 – 7	3000	16	-	0,60	0,30	0,65	0,30
	Breeding	2900	15	-	2,75	0,30	0,60	0,27
Burung Puyuh	Starter dan Grower	2900	24	1,0	0,80	0,30	1,30	0,50
	Breeding	2900	20	1,0	2,50	0,35	1,00	0,45
Kalkun	0 – 4	2800	28	1,0	1,20	0,60	1,60	0,55
	4 – 8	2900	26	1,0	1,00	0,50	1,50	0,45
	8 – 12	3000	22	0,8	0,85	0,42	1,30	0,40
	12 – 16	3100	19	0,8	0,75	0,38	1,00	0,35
	16 – 20	3200	16	0,8	0,65	0,32	0,80	0,25
	20 – 24	3300	14	0,8	0,55	0,28	0,65	0,25
	Holding	2900	12	0,8	0,50	0,25	0,50	0,20
	Laying Hens	2900	14	1,1	2,25	0,35	0,60	0,20

Data Komposisi Kimia Bahan Pakan

Bahan pakan yang dimasukkan sebagai data acuan adalah data komposisi kimia bahan pakan unggas menurut NRC '94 (Tabel 2).

Tabel 2. Data Komposisi Beberapa Bahan Pakan Berdasarkan NRC (1994)

No.	Nama Bahan	BK (%)	EMn (Kcal/kg)	Protein (%)	LK (%)	SK (%)	Ca (%)	P tersedia (%)	Cl (%)	Fe (mg/kg)	Mg (%)	Na (%)
1	Tepung darah	94	2830	81,1	-	0,50	0,55	0,42	0,27	2,02	0,16	0,32
2	Bungkil kelapa	92	1525	19,2	-	14,40	0,17	0,22	0,03	-	0,31	0,04
3	Jagung kuning	90	3350	8,5	2,20	2,20	0,02	0,08	0,04	460,00	0,29	0,15
4	Bungkil kedelai	89	2230	44,0	0,40	7,00	0,29	0,65	0,05	120,00	0,27	1,10
5	MBM	93	2150	50,4	0,36	2,80	10,3	5,10	0,69	490,00	1,12	0,70
6	Tepung bulu	93	2360	81,0	-	1,00	0,33	0,55	0,28	76,00	0,20	0,84
7	CaCO ₃	100	-	-	-	-	38,00	-	-	-	-	-
8	DCP	100	-	-	-	-	22,00	18,70	-	-	-	-
9	CGM	90	3720	62,0	-	1,30	-	0,14	0,05	400,00	0,15	0,02
10	Tepung daging	92	2195	54,4	0,28	2,70	8,27	4,10	0,69	440,00	0,58	1,15

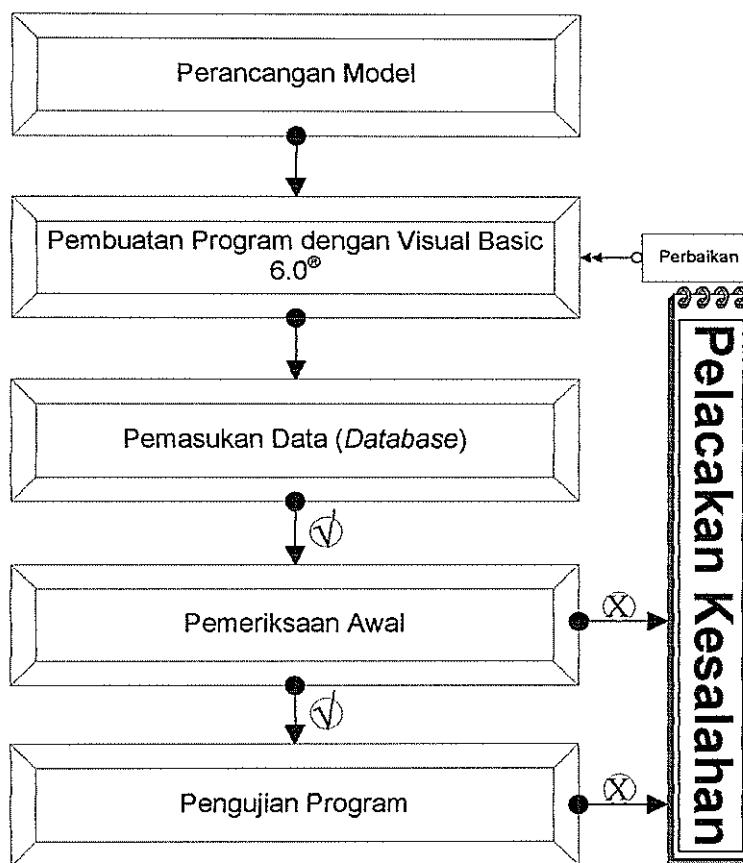
Tabel 2. Lanjutan

No.	Nama Bahan	Vit. B12 (µg/kg)	Vit. E (mg/kg)	Lisin (%)	Metionin (%)	Sistin (%)	Triptophan (%)
1	Tepung darah	44,00	-	7,05	0,55	0,52	1,29
2	Bungkil kelapa	-	-	0,50	0,28	0,28	0,12
3	Jagung kuning	-	15,00	0,63	0,45	0,51	0,10
4	Bungkil kedelai	-	2,00	2,69	0,62	0,66	0,74
5	MBM	70,00	1,00	12,61	0,69	0,69	0,27
6	Tepung bulu	78,00	-	2,28	0,57	4,34	0,55
7	CaCO ₃	-	-	-	-	-	-
8	DCP	-	-	-	-	-	-
9	CGM	-	24,00	1,03	1,49	1,10	0,36
10	Tepung daging	68,00	1,00	3,00	0,75	0,66	0,36

Metode

Tahapan Penelitian

Pembuatan program ini terbagi dalam beberapa tahap seperti tampak pada gambar 1. Tahapan-tahapan tersebut meliputi: 1) perancangan model, 2) pembuatan program dengan Visual Basic 6.0®, 3) pemasukan data sementara, 4) pemeriksaan dan pengujian awal program 5) pelacakan dan perbaikan kesalahan, 6) uji coba dengan melakukan formulasi ransum unggas di Laboratorium Komputer Departemen Ilmu Nutrisi dan Makanan Ternak, Fakultas Peternakan, Institut Pertanian Bogor.



Gambar 1. Skema Pembuatan Program Formulasi Ransum

Perancangan Model

Dalam perancangan model, metode program linear yang menjadi inti dari program ini diterjemahkan kedalam bahasa pemrograman Visual Basic. Model program linear yang digunakan adalah program linear dengan sistem *simplex* yang mempunyai bentuk umum sebagai berikut :

Fungsi tujuan (meminimumkan biaya) :

$$Z_{\min} = C_1X_1 + C_2X_2 + C_3X_3 + \dots + C_nX_n$$

Dengan fungsi kendala :

$$A_{11}X_{11} + A_{12}X_{12} + A_{13}X_{13} + \dots + A_{1n}X_{1n} \geq b_1$$

$$A_{21}X_{21} + A_{22}X_{22} + A_{23}X_{23} + \dots + A_{2n}X_{2n} \geq b_2$$

$$A_{m1}X_{m1} + A_{m2}X_{m2} + A_{m3}X_{m3} + \dots + A_{mn}X_{mn} \geq b_m$$

Serta dengan Asumsi :

$$X_1, X_2, X_3, \dots, X_n \geq 0$$

Dimana :

- Z : harga ransum yang diperoleh
- C : harga bahan makanan yang digunakan
- X : bahan makanan yang digunakan
- A : Kandungan nutrisi bahan makanan
- b : standar kebutuhan nutrisi
- m : 1, 2, 3, dst
- n : 1, 2, 3, dst

Pembuatan Program

Metode tersebut kemudian diterjemahkan dalam bentuk kode perintah-perintah yang dikenali oleh Visual Basic. Apabila perintah-perintah tersebut benar, Visual Basic akan menerjemahkan dan menjalankannya. Akan tetapi, bila perintah tersebut salah, Visual Basic akan segera menampilkan pesan dan menunjukkan letak kesalahannya sehingga lebih mudah untuk memperbaikinya.

Pemasukan Data

Komposisi kimia bahan pakan dan standar kebutuhan nutrisi ternak unggas dimasukkan dalam sebuah *database*. Sebagai data sementara, data yang diisikan hanya terdiri dari sedikit bahan pakan dan hanya beberapa jenis unggas saja yang dimasukkan standar kebutuhan nutrisinya. Tujuan dari pengisian data sementara ini adalah untuk melakukan pengujian awal program. Data tersebut meliputi 8 faktor pembatas utama yaitu kebutuhan unggas terhadap energi metabolismis, protein kasar, lemak kasar, serat kasar, kalsium, phospor tersedia, asam amino lisin dan asam amino Metionin. Selain faktor pembatas diatas, juga juga beberapa mineral (Fe, Mg, Cl dan Na), asam amino essensial dan semi essensial (tryptophan dan sistin) dan beberapa vitamin (vitamin B12 dan E) untuk diperhitungkan dalam hasil akhir

sebagai informasi tambahan, bukan sebagai faktor pembatas. Data yang diperlukan untuk standar kebutuhan nutrisi unggas antara lain jenis unggas lengkap dengan umur atau fase produksinya dan kebutuhan unggas terhadap 8 faktor pembatas utama minimal dan maksimal.

Pengujian

Dalam pengujian program baik awal maupun akhir, apabila ditemukan suatu kesalahan akan segera dilakukan perbaikan. Kesalahan tersebut akan ditunjukkan langsung oleh Visual Basic dalam bentuk pesan *debug* (pada pengujian awal) atau pesan *error* bila program sudah dalam bentuk aplikasi (pada pengujian akhir).

Suatu program aplikasi dalam Visual Basic terdiri dari *form-form* (*.FRM) yang tergabung dalam suatu file induk yang disebut dengan *project file* (*. VBP) (Alam, 1999). File *project* tersebut nantinya akan dikompilasi dengan Microsoft Visual Basic® menjadi suatu file aplikasi yang siap untuk dijalankan.

Untuk memudahkan dalam instalasi program, dalam Visual Basic juga telah disediakan fasilitas *Package & Deployment Wizard* yang digunakan untuk membuat file instalasi (SETUP.EXE). File instalasi ini dapat didistribusikan dalam bentuk kepingan *Compact Disc* (CD).

Kebutuhan Operasional

Untuk dapat menggunakan program ini diperlukan komputer dengan spesifikasi minimal Pentium 100, memori 32 MB, ruang *Hard Disk* kosong sebesar 50 MB, monitor yang mampu mendukung resolusi 800 x 600 pixels dengan sistem Operasi Microsoft Windows 98®.

HASIL DAN PEMBAHASAN

Metode Program Linear

Metode program linear yang digunakan dalam program ini adalah metode program linear dua fase. Tahapan – tahapan dalam iterasi program linear fase I tersebut adalah sebagai berikut:

1. Pembacaan data dari basis data (*database*) berdasarkan bahan-bahan yang digunakan dalam formulasi dan jenis unggas yang standar kebutuhan nutrisinya dijadikan pembatas dalam formulasi;
2. Memasukkan variabel *slack* (S) / *surplus* (-S) dan *artifisial* (A) sesuai dengan data batasan yang telah dimasukkan;
3. Memasukkan nilai fungsi tujuan (C_j) yang merupakan tujuan utama dari formulasi;
4. Data beserta *slack*, *surplus* dan *artifisial* dimasukkan dalam bentuk tabel yang meliputi kolom C_j, C_b, Z_j, H dan X seperti pada Tabel 3 berikut ini.

Tabel 3. Bentuk Umum Iterasi Program Linear Fase I

	C _j		0	0	0	0	0	0	-1	-1
C _b	vdb	H	X ₁	X ₂	S ₁	S ₂	A ₁
0	A ₁									
-1	S ₂									
0	A ₂									
-1	S ₄									
.....									
	Z _j									
	Z _j - C _j									

Keterangan :

C_j : Baris ini diisi dengan nilai 0 kecuali untuk variabel artifisial (A) diberi nilai -1,

Vdb : Variabel dalam basis yang merupakan variabel buatan dari data yang nilainya ada di C_j.

H : Kolom ini merupakan nilai dari kendala atau batasan nutrisi yang digunakan.

X_n : Kolom ini diisi dengan nilai kandungan nutrisi dari bahan X_n.

S_n : Kolom ini merupakan variabel *slack* (bernilai +1) / *surplus* (bernilai -1) dari tiap kendala atau batasan nutrisi.

A_n : Kolom ini merupakan variabel *Artifisial* dari tiap kendala atau batasan nutrisi.

5. Pada tahap selanjutnya, kolom terpilih (k) ditentukan dengan mencari nilai Z_j-C_j negatif terkecil;

6. Menentukan baris terpilih yang merupakan nilai terkecil dari rasio (r). Rasio sendiri merupakan hasil bagi antara nilai-nilai pada kolom H dengan nilai-nilai pada kolom terpilih (k);
7. Membuat tabel baru dengan jumlah baris dan kolom sama dengan tabel sebelumnya. Nilai-nilai kolom baru diperoleh dengan cara :
 - Membuat vektor *transformasi* dimana $T(n) = -((N(n,k) / N(r,k))$ kecuali bila $n = r$ maka $T(n) = (1/N(r,k)) - 1$. Setelah selesai kemudian menentukan nilai kolom baru dimana nilai kolom baru merupakan penjumlahan antara nilai N kolom lama dengan perkalian antara vektor *transformasi* ($T(n)$) dengan nilai (N) pada kolom ke- n baris terpilih (r).

$$N(n)' = N(n) + (T(n) \times N(n,r))$$

8. Dilakukan pengulangan (*Looping*) terhadap langkah 5, 6 dan 7 sampai nilai $Z_j - C_j \geq 0$ sebagai ketentuan berakhirnya iterasi Fase I.

Pada fase I ini semua variable buatan dibuat menjadi nol (0), kemudian baru menyelesaikan fungsi tujuan yang sebenarnya (Z), dimulai dari suatu pemecahan dasar yang fisibel baik yang memuat vektor buatan dengan nilai variable pada tingkat nol atau tidak memuat vektor buatan sama sekali. Proses ini disebut dengan Fase II. Tahapan dari Fase II tersebut sebagai berikut:

1. Setelah semua nilai $Z_j - C_j$ pada iterasi Fase I terpenuhi, kemudian dilanjutkan dengan fase II dengan memasukkan harga dari tiap bahan yang digunakan (hX_n) sesuai dengan yang ditunjukkan pada Tabel 4;

Tabel 4. Bentuk Umum Iterasi Program Linear Fase II

	C_j		- hX ₁	- hX ₂	0	0	0
C_b	vdb	H	X₁	X₂	S₁	S₂
0	A1							
-1	S2							
0	A2							
-1	S4							
.....							
	Z_j							
	Z_j - C_j							

2. Sama halnya dengan fase I, selanjutnya kolom terpilih (k) ditentukan dengan mencari nilai $Z_j - C_j$ negatif terkecil;
3. Menentukan baris terpilih yang merupakan nilai terkecil dari rasio (r). Rasio sendiri merupakan hasil bagi antara nilai-nilai pada kolom H dengan nilai-nilai pada kolom terpilih (k);
4. Mulai membuat tabel baru dengan jumlah baris dan kolom sama dengan tabel sebelumnya. Nilai-nilai kolom baru diperoleh dengan cara :
 - a. Membuat vektor transformasi dimana $T(n) = -(N(n,k) / N(r,k))$ kecuali bila $n = r$ maka $T(n) = (1/N(r,k)) - 1$. Setelah selesai kemudian menentukan nilai kolom baru dimana nilai kolom baru merupakan penjumlahan antara nilai N kolom lama dengan perkalian antara vektor transformasi ($T(n)$) dengan nilai (N) pada kolom ke-n baris terpilih (r).

$$N(n)' = N(n) + (T(n) \times N(n,r))$$

5. Dilakukan pengulangan (*Looping*) terhadap langkah 2, 3 dan 4 sampai nilai $Z_j - C_j \geq 0$ sebagai ketentuan berakhirnya iterasi Fase II.

Setelah iterasi selesai, kemudian data yang ada ditampilkan pada *form Least Cost* dalam program formulasi ransum unggas agar dapat dilihat oleh pengguna.

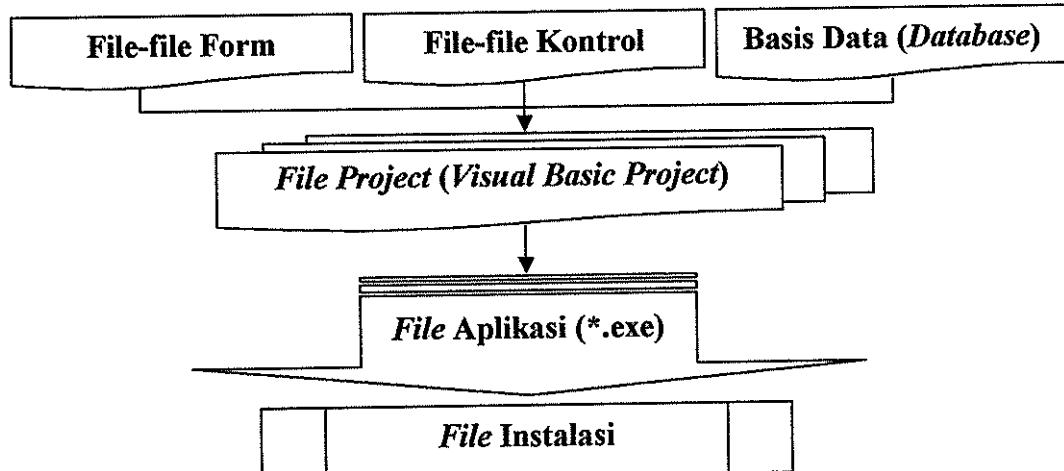
Pembuatan Program

Program EzFeed (dibaca "Easy Feed") versi 1.0 merupakan aplikasi dari metode program linear dalam formulasi ransum unggas. Program ini dibuat dengan menggunakan sistem pemrograman Microsoft Visual Basic 6.0®. Sistem ini dipilih sebagai sistem pemrograman karena selain termasuk sebagai sistem pemrograman yang bersifat *visual*, sistem ini juga termasuk mudah untuk digunakan terutama bagi programmer-programmer baru. Sistem pemrograman Visual Basic memadukan konsep antara sistem pemrograman yang berorientasi obyek (*Object Oriented Programming*) dan sistem pemrograman yang berorientasikan *event* (Dewobroto, 2003). Program yang dibangun dengan sistem pemrograman Visual Basic 6.0 dapat bekerja pada lingkungan kerja Microsoft Windows™.

Program formulasi ransum unggas ini dibuat untuk mempermudah, mempercepat dan memperkecil kesalahan dalam formulasi ransum unggas. Dengan digunakannya metode program linear sebagai metode dasar formulasi ransum,

program ini diharapkan dapat menghasilkan suatu *formula ransum* yang sesuai dengan keinginan pengguna tanpa mengabaikan faktor harga.

Pembuatan program ini dimulai dari pembuatan *file project* yang merupakan *file* induk dari *file-file form*, *file* kontrol dan basis data (*database*). *File project* ini kemudian dikompilasi menjadi suatu *file* aplikasi (*.exe) dan untuk selanjutnya dibuatkan *file* instalasi. Skema pembuatan program EZFeed seperti yang ditampilkan pada gambar 2:



Gambar 2. Skema Pembuatan Program

Pemasukan Data Dalam Perhitungan

Dalam memformulasikan sebuah ransum, diperlukan data komposisi bahan pakan dan batasan penggunaan maksimal dan minimal dari setiap bahan tersebut. Semua data-data tersebut harus sudah terdapat dalam basis data. Apabila data dalam basis data tidak lengkap atau formatnya tidak benar, maka perhitungan tentunya tidak akan berjalan dengan benar pula. Untuk mencegah terjadinya kesalahan tersebut, ditambahkan beberapa kotak pesan yang berisi peringatan bila terjadi kesalahan dalam pengisian data. Selain berupa kotak pesan, pencegahan juga dilakukan dengan tidak diperkenakkannya pengisian karakter selain karakter numerik pada kolom yang memerlukan data *numerik*. Adapun petikan kode pencegahan tersebut adalah sebagai berikut :

```
Private Sub txtFields_KeyPress(Index As Integer, KeyAscii As Integer)
    If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii =
vbKeyBack Or KeyAscii = Asc(".")) Then
        KeyAscii = 0
    End If
End Sub
```

Kode untuk mengambil data bahan pakan yang berupa komposisi kimia, harga dan penggunaan minimal serta maksimal bahan dalam formulasi ransum adalah seperti berikut ini:

```
Function getBahan(nama_bahan As String) As ADODB.Recordset
    Dim rs As New ADODB.Recordset
    Dim sql As String
    sql = "select "
    sql = sql & " nama_bahan as nama "
    sql = sql & ", -1*harga as harga "
    sql = sql & ", energi_metabolis as em"
    sql = sql & ", energi_metabolis as emMax"
    sql = sql & ", protein_kasar as pk"
    sql = sql & ", protein_kasar as pkMax"
    sql = sql & ", lemak_kasar as lk"
    sql = sql & ", lemak_kasar as lkMax"
    sql = sql & ", serat_kasar as sk"
    sql = sql & ", serat_kasar as skMax"
    sql = sql & ", kalsium as ca "
    sql = sql & ", kalsium as caMax "
    sql = sql & ", phospor_tersedia as p_tesedia"
    sql = sql & ", phospor_tersedia as p_Max"
    sql = sql & ", lisin as lys "
    sql = sql & ", lisin as lysMax "
    sql = sql & ", metionin as met "
    sql = sql & ", metionin as metMax "
    sql = sql & ", bm_min/100 as bmMin "
    sql = sql & ", bm_max/100 as bmMax "
    sql = sql & " from bahan "
    sql = sql & " where nama_bahan = '" & nama_bahan & "'"
    rs.Open sql, cn, adOpenKeyset
    Set getBahan = rs
    Set rs = Nothing
End Function
```

Kode untuk mengambil data mengenai unggas beserta standar kebutuhan nutrisinya adalah seperti berikut ini :

```
Function getUnggas(unggas As String) As ADODB.Recordset
    Dim rs As New ADODB.Recordset
    Dim sql As String
    sql = "select "
    sql = sql & " unggas as nama "
    sql = sql & ", em"
    sql = sql & ", emMax"
    sql = sql & ", pk"
    sql = sql & ", pkMax"
    sql = sql & ", lk"
    sql = sql & ", lkMax"
    sql = sql & ", sk"
    sql = sql & ", skMax"
    sql = sql & ", ca"
    sql = sql & ", caMax "
    sql = sql & ", p_tersedia"
    sql = sql & ", p_Max"
    sql = sql & ", lys"
    sql = sql & ", lysMax "
    sql = sql & ", met"
    sql = sql & ", metMax "
    sql = sql & ", O as bmMin "
    sql = sql & ", O as bmMax "
    sql = sql & " from batnut "
    sql = sql & " where unggas = '" & unggas & "'"
    rs.Open sql, cn, adOpenKeyset
    Set getUnggas = rs
    Set rs = Nothing
End Function
```

Pengguna dapat membaiki basis data untuk mendapatkan *formula* pakan yang sesuai dengan kebutuhan unggas dengan leluasa.

Manajemen Basis Data

Semua data mengenai bahan pakan dan standar kebutuhan nutrisi unggas dimasukkan dalam basis data yang dibuat dengan Microsoft Access '97 dengan nama file Feed.MDB. Dalam *database* tersebut terdapat percabangan data (*recordset*), yaitu *recordset* BAHAN yang berisi data mengenai bahan pakan beserta komposisi kimia, *informasi* dan batasan penggunaan bahan tersebut. *Recordset* kedua yaitu *recordset* BATNUT berisi standar kebutuhan nutrisi unggas. Selain *database* tersebut juga terdapat satu tambahan *database* yang berisi mengenai tips-tips dalam penggunaan program, *database* tersebut diberi nama Tooltips.MDB.

Tabel 5. Struktur Basis Data FEED.MDB

Nama Field	Tipe	Nama Field	Tipe	Keterangan
Kode	Text	Unggas*	Text	Unggas beserta fasenya
Nama_bahan	Text	EM	Number	Energi Metabolis
Harga	Currency	Emmax**	Number	EM maksimum
Bahan_kering	Number	PK	Number	Protein Kasar
Energi_metabolis	Number	Pkmax**	Number	PK maksimum
Protein_kasar	Number	LK	Number	Lemak Kasar
Lemak_kasar	Number	Lkmax**	Number	LK maksimum
Serat_kasar	Number	SK	Number	Serat Kasar
Kalsium	Number	Skmax**	Number	SK maksimum
Phospor_tersedia	Number	Ca	Number	Kalsium
Chlor	Number	CaMax**	Number	Ca maksimum
Zat_besi	Number	P_tersedia	Number	Phospor tersedia
Magnesium	Number	P_max**	Number	P. tersedia maksimum
Natrium	Number	Lys	Number	Asam amino Lisin
Vitamin_B12	Number	LysMax**	Number	Lys. Maksimum
Vitamin_E	Number	Met	Number	Asam amino Metionin
Lisin	Number	MetMax**	Number	Met. Maksimum
Metionin	Number			
Sistin	Number			
Triptophan	Number			
Info	Text			
Bm_min	Number			
Bm_max	Number			

Keterangan :

* : Diisi unggas beserta umur atau fase produksinya.

** : Standar kebutuhan maksimum unggas.



Pengujian Program

Pengujian program dilaksanakan pada saat pembuatan program dan setelah program selesai dibuat. Kesalahan pada saat pengujian awal (saat pembuatan program) ditandai dengan keluarnya pesan kesalahan (debug) yang disertai dengan ditampilkannya petunjuk kesalahan dan letak kesalahan. Pengujian tahap akhir (setelah program dibuat) dilakukan dengan melakukan formulasi ransum unggas dengan menggunakan bahan-bahan yang telah tersedia dalam basis data bahan pakan.

Faktor Pembatas

Batasan kebutuhan nutrisi dan batasan penggunaan bahan sebagai faktor pembatas dibuat dalam bentuk suatu kisaran minimal dan maksimal, sehingga dalam iterasi faktor-faktor pembatas tersebut selalu diperhitungkan.

Dalam perhitungan, harga yang digunakan merupakan harga *absolut* dan bukan merupakan harga relatif bahan terhadap suatu kandungan nutrisi bahan tertentu. Hal ini dilakukan karena dalam unggas terdapat dua standar harga relatif bahan, yaitu harga relatif bahan terhadap energi metabolismis dan harga relatif bahan terhadap kandungan protein bahan. Penggunaan harga relatif tersebut tergantung pada jenis unggas yang digunakan.

Komposisi kimia bahan yang diperhitungkan sebagai faktor pembatas antara lain energi metabolismis, protein kasar, serat kasar, lemak kasar, kalsium, fosfor tersedia, asam amino lisin dan asam amino metionin. Energi metabolismis diperhitungkan sebagai faktor pembatas karena energi metabolismis merupakan nilai yang digunakan dalam perhitungan-perhitungan untuk ransum aneka ternak unggas (Anggorodi, 1995). Protein diperhitungkan sebagai faktor pembatas karena protein adalah unsur pokok alat tubuh dan jaringan lunak tubuh ternak (Anggorodi, 1995). Zat tersebut diperlukan untuk pertumbuhan, pengelolaan dan produksi telur serta merupakan bagian dari enzim dalam tubuh. Protein dan lemak merupakan unsur yang penting dalam tubuh, karena saat kekurangan energi, unggas dapat mengubah protein dan lemak menjadi energi (Rasyaf, 1984).

Bahasan mengenai protein tentunya tidak pernah terlepas dari pembahasan mengenai asam amino. Menurut Rasyaf (1984), ada empat asam amino yang seharusnya diperhitungkan sebagai faktor pembatas, yaitu asam amino lisin,

metionin, sistin dan triptophan. Untuk memudahkan formulasi ransum, maka hanya asam amino lisin dan metionin saja yang diperhitungkan sebagai faktor pembatas.

Serat kasar bagi unggas merupakan faktor pembatas. Hal ini disebabkan karena pada unggas tidak memiliki kemampuan untuk mencerna serat kasar (Anggorodi, 1995).

Mineral dapat digunakan dalam jumlah sedikit dalam ransum, karena zat mineral merupakan zat nutrisi yang jumlahnya sangat sedikit dalam tubuh(Anggorodi, 1995). Mineral yang sangat penting bagi unggas antara lain kalsium dan phospor, karena kalsium dan phospor merupakan lebih dari 70 % abu tubuh. Lebih kurang 99 % kalsium dan 80 % phospor terdapat dalam tulang (Anggorodi, 1995).

Selain kedelapan faktor pembatas tersebut, juga terdapat beberapa nutrisi yang ikut diperhitungkan dalam *formula* ransum tetapi bukan sebagai faktor pembatas melainkan sebagai informasi tambahan saja. Zat-zat nutrisi tersebut antara lain Chlor, Zat besi, Magnesium, Natrium, Vitamin B₁₂, Vitamin E, Sistin dan Triptophan. Perhitungan dari kedelapan zat nutrisi tersebut dilakukan setelah formulasi selesai.

Fasilitas dan Tampilan Program

Program ini dilengkapi dengan beberapa fasilitas yang dapat digunakan oleh pengguna program ini melakukan formulasi ransum unggas. Fasilitas tersebut terbagi menjadi dua, yaitu fasilitas utama berupa Edit Data dan Formulasi secara *Least Cost*. Fasilitas kedua yang merupakan fasilitas pendukung terdiri dari *Feed Mix*, menu tips, kalkulator, bantuan, buka data tersimpan dan status basis data.

1. *Form Pembuka*

Pada *form* pembuka, pengguna diminta untuk menuliskan namanya. Tujuan dari pengisian ini adalah ketika pengguna mencetak atau menyimpan hasil formulasi, nama pengguna akan ikut tercetak dalam hasil formulasi tersebut lengkap dengan waktu pembuatan. Tampilan dari *form* pembuka tersebut adalah sebagai berikut :



Gambar 3. Tampilan *Form* Pembuka

2. *Form Utama*

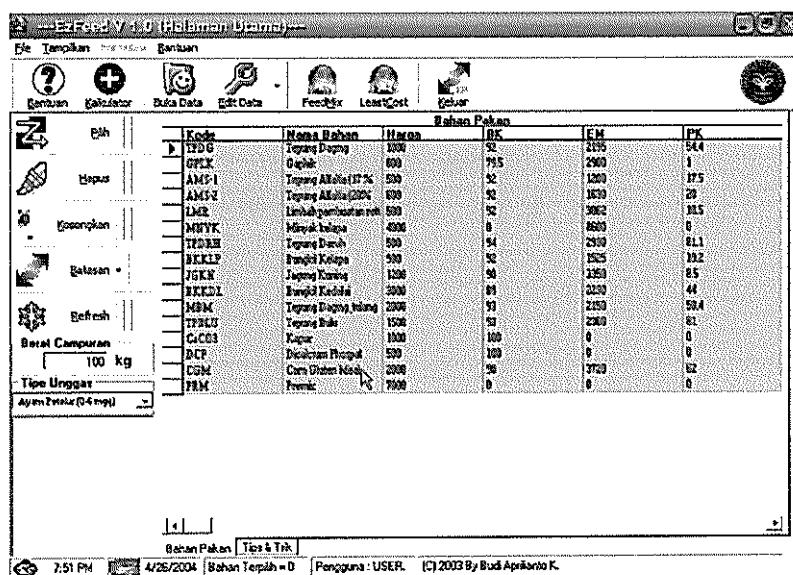
Pada *form* utama terdapat beberapa menu dan tombol. Menu-menu tersebut antara lain menu *File* yang terdiri dari submenu Edit Data [Bahan Pakan dan Kebutuhan Nutrisi], Submenu Buka Data dan Submenu Keluar. Submenu Edit Data [Bahan Pakan] digunakan untuk menambah, mengubah atau menghapus data bahan pakan dalam basis data sedangkan menu Edit Data [Kebutuhan Nutrisi] digunakan untuk menambah, mengubah atau menghapus data dalam basis data standar kebutuhan nutrisi unggas. Submenu Buka Data seperti namanya digunakan untuk

membuka data hasil formulasi yang disimpan untuk dilakukan formulasi ulang (*Re-Formulate*) ataupun untuk dicetak kembali.

Selain itu juga terdapat menu Tampilkan yang terdiri dari submenu Bahan, Tips & Trik, Status, Batasan [Nutrisi dan Penggunaan Bahan] dan Kalkulator. Submenu bahan berfungsi untuk menampilkan tabel bahan pakan yang dapat digunakan dalam formulasi. Bahan dalam tabel tersebut dapat dipilih dengan cara meng-klik ganda bahan yang dipilih atau dengan cara meng-klik sekali pada bahan yang dipilih, kemudian tekan tombol Pilih di sebelah kiri tabel bahan. Submenu Tips & Trik digunakan untuk menampilkan bagian Tips dan Trik dalam penggunaan program ini. Submenu status berisi mengenai informasi jumlah masukan data dalam basis data bahan pakan maupun unggas. Submenu Batasan Nutrisi digunakan untuk mengubah standar kebutuhan nutrisi dari unggas yang dipilih dan submenu Batasan Penggunaan bahan digunakan untuk mengubah batasan penggunaan bahan yang dipilih. Submenu Kalkulator berfungsi untuk menampilkan kalkulator yang dapat digunakan untuk melakukan perhitungan tersendiri.

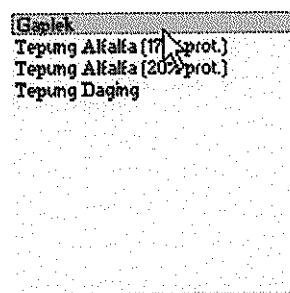
Selain menu *File* dan *Tampilkan*, juga terdapat menu *Formulasi* dan *Menu Bantuan*. Menu *Formulasi* berisi submenu *Feed Mix* dan *Least Cost*. Menu bantuan berfungsi untuk menampilkan fasilitas bantuan operasional dari program ini. Sub menu *Feed Mix* digunakan untuk melakukan formulasi pakan dengan metode coba-coba (*Trial and Error Method*), sedangkan submenu *Least Cost* yang merupakan inti dari program ini digunakan untuk melakukan formulasi pakan unggas dengan metode program linear. Kedua submenu ini dapat berfungsi jika pengguna sudah memilih bahan terlebih dahulu untuk digunakan dalam formulasi.

Tampilan dari *form Utama* diperlihatkan pada gambar 4:



Gambar 4. Tampilan dari *Form* Utama

Selain menu-menu tersebut juga terdapat beberapa tombol yang baik nama maupun fungsinya sama dengan pada bagian menu-menu tersebut. Tombol-tombol tersebut berfungsi untuk mempermudah pengguna dalam menjalankan program "EzFeed" ini. Pada bagian kiri bawah dari *form* ini terdapat *list* bahan yang telah dipilih untuk selanjutnya digunakan dalam formulasi pakan baik secara *Feed Mix* maupun *Least Cost*. Tampilan dari bagian ini adalah seperti pada gambar 5:



Gambar 5. Tampilan dari *List* Bahan Terpilih

3. Menu Edit Data

Menu ini terbagi menjadi dua, yaitu edit data nutrisi dan edit data bahan pakan.

➤ Edit Data Nutrisi :

Menu ini digunakan untuk menambah, mengubah maupun menghapus data standar kebutuhan nutrisi unggas yang dapat

digunakan dalam penyusunan ransum. Dalam *form* ini, pengguna diminta untuk menuliskan secara lengkap data mengenai jenis unggas lengkap dengan fase produksi atau umurnya, standar kebutuhan nutrisi minimal dan maksimal unggas terhadap delapan faktor pembatas (Energi metabolismis, Protein kasar, Lemak kasar, Serat kasar, Kalsium, Phosphor tersedia, Lisin dan Metionin). Jika data tidak diisi secara lengkap dapat mengakibatkan terjadinya kesalahan (*error*) dalam formulasi secara *Least Cost*. Tampilan dari *form* ini seperti tampak pada gambar 6:

	Minimal	Maksimal	Kkal/Kg
Energi Metabolis	2800	2005	%
Protein Kasar	16	16.5	%
Lemak Kasar	1	1.1	%
Serat Kasar	0	4	%
Kalsium	0.9	0.95	%
Phosphor tersedia	0.44	0.45	%
Asam Amino Lisin	1.6	1.65	%
Asam Amino Metionin	2.8	3	%

Gambar 6. Tampilan dari *form* Edit Data Nutrisi

➤ Edit Data Bahan Pakan :

Seperti halnya dengan menu Edit Data Nutrisi, menu ini juga dapat digunakan untuk menambah, mengubah atau menghapus data bahan pakan pada basis data. Pada *form* Edit Data bahan Pakan ini, pengguna selain diminta untuk mengisikan nama bahan pakan dan komposisi kimia bahan tersebut juga diminta untuk mengisikan kode bahan, batasan penggunaan minimal dan maksimal bahan tersebut dalam ransum serta informasi fungsi dari bahan tersebut. Tampilan dari *form* Edit Data Bahan Pakan seperti diperlihatkan pada gambar 7:

Komposisi Bahan Pakan

Kode	TPDG	
Nama Bahan	Tujuan Daging	Data : 1/16
Harga	1000 Rp/kg	Zat Besi 440 mg/kg
Bahan Kering	92 %	Magnesium 0.58 %
Energi Metabolis	2195 kcal/kg	Natrium 1.15 %
Protein Kasar	54.4 %	Vitamin B-12 68 µg/kg
Lemak Kasar	0.28 %	Vitamin E 1 mg/kg
Serat Kasar	2.7 %	Lisin 3 %
Kalsium	8.27 %	Metionin 0.75 %
Phospor Tersedia	4.1 %	Sistin 0.66 %
Chlor	0.69 %	Triptofan 0.36 %
Penggunaan	Min (%) 0 Max (%) 100	Informasi Sumber Protein

[Tambah Data] [Hapus] [Refresh] [Simpan] [Tutup]

(C) 2003 By Budi Aprilianto Kusnandar

Gambar 7. Tampilan dari *Form* Edit Data Bahan Pakan

4. Menu Batasan

Menu ini digunakan untuk membatasi penggunaan dari bahan pakan yang akan digunakan dalam formulasi ransum maupun membatasi kebutuhan nutrisi (Batasan Nutrisi) sesuai dengan ransum yang akan disusun. Pengguna cukup mengubah data yang ditampilkan (setelah sebelumnya memilih bahan dan jenis ransum) dan kemudian menekan tombol “Simpan Perbaikan” untuk menyimpan perubahan data. Berikut tampilan dari *form* batasan penggunaan bahan dan nutrisi (Gambar 8 dan 9).

EzFeed V 1.0 (Batasan Bahan Makanan)

BATASAN PENGGUNAAN BAHAN

Nama Bahan Pakan	Min (%)	Max (%)
Jagung Kuning	0	30
Limbah pembuatan roti	0	50
Tepung Daging & tulang	0	15

Simpan Perbaikan

Update

Bahan	Nutrisi
-------	---------

Refresh FeedMix LeastCost

(C) 2003 by Budi Aprilianto Kusnandar

Gambar 8. Form Batasan Penggunaan Bahan

EzFeed (batasan nutrisi)

STANDAR NUTRISI

Unggas (+ Fase Produksi/Umur) Ayam Petelur (0-6 minggu)

	Minimal	Maksimal	
Energi Metabolis	2800	2805	kcal/kg
Protein Kasar	16	16.5	%
Lemak Kasar	1	1.1	%
Serat Kasar	0.9	1.4	%
Kalsium	0.9	0.95	%
Phosphor tersedia	0.44	0.45	%
Asam Amino Lisin	1.6	1.65	%
Asam Amino Metionin	2.8	3	%

Simpan Perbaikan

Update

Bahan	Nutrisi
-------	---------

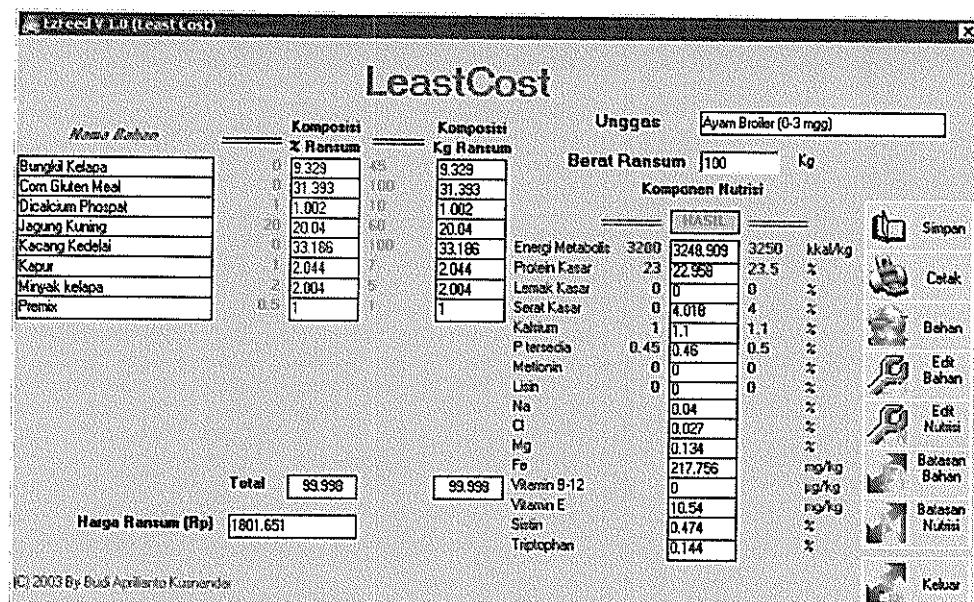
Refresh FeedMix LeastCost

(C) 2003 by Budi Aprilianto Kusnandar

Gambar 9. Form Batasan Nutrisi

5. Form Least Cost

Bagian ini merupakan bagian inti dari program formulasi ransum EzFeed. Pada bagian ini pengguna dapat melihat hasil formulasi ransum dengan metode *Least Cost*. Setelah menyelesaikan formulasi ransum, pengguna dapat langsung mencetak hasilnya maupun menyimpannya dalam *file* data (*.Aan) sehingga pada kesempatan lain dapat dilakukan formulasi ulang (*Re-Formulate*). Tampilan dari *form Least Cost* seperti yang ditampilkan pada gambar 10:



Gambar 10. Tampilan dari *Form Least Cost*

6. Form Feed Mix

Fasilitas ini merupakan fasilitas tambahan untuk melakukan formulasi ransum dengan metode coba-coba (*Trial and Error Method*). Pengguna cukup memasukkan angka pada kolom "Komposisi % Ransum" maka secara otomatis program akan langsung menghitungkan komposisi nutrisi dalam ransum tersebut.

Sama halnya dengan fasilitas *Least Cost*, pengguna juga dapat langsung mencetak hasilnya atau menyimpannya dalam *file* data (*.Aan) untuk dilakukan formulasi ulang (*Re-Formulate*) pada kesempatan lain. Gambar 11 merupakan tampilan dari *Form Feed Mix*.

The screenshot shows the FeedMix software interface. At the top, it says "FeedMix". Below that, there's a table for "Komposisi Ransum" (Feed Composition) with columns for "Nama Bahan", "Kompositi % Ransum", and "Kompositi Kg Ransum". The table includes rows for "Gulek", "Tepung Daging & bahan", and "Tepung Durian". To the right of the table, there's a section for "Berat Ransum" (Weight of Ransum) set to 100 kg, and a "Komponen Nutrisi" (Nutrient Components) table with rows for Energi Metabolis, Protein Kotor, Lemak Kotor, Serat Kotor, Kalsium, Phosforida, Na, Cl, Mg, Fe, Vitamin E-12, Vitamin E, Nitrogen, Lisis, Sistin, and Triptofan. On the far right, there are several buttons: "Simpan" (Save), "Cetak" (Print), "Bahan" (Ingredients), "Edit Bahan" (Edit Ingredients), "Edit Nutrisi" (Edit Nutrients), "Bahan-Bahan" (Ingredients-Ingredients), "Bahan-Nutrisi" (Ingredients-Nutrients), and "Keluar" (Exit). At the bottom left, there are fields for "Total Harga Ransum" (Total Feed Cost) and "Rp/Kg". A copyright notice "(C) 2003 By Budi Apriyanto Kusumawardhani" is at the bottom.

Gambar 11. Tampilan dari *Form Feed Mix*

7. *Form Data Tersimpan*

Sesuai dengan namanya, fasilitas ini digunakan untuk menampilkan hasil formulasi yang disimpan dalam bentuk *file* data (*.Aan). Pengguna dapat kembali mencetaknya maupun melakukan formulasi ulang ransum tersebut. Apabila ada satu atau beberapa bahan tidak lagi terdapat dalam basis data, maka secara otomatis program akan menghilangkannya dari daftar bahan terpilih. Gambar 12 merupakan tampilan dari *Form Data Tersimpan*.

This screenshot shows the "Data Tersimpan" (Stored Data) screen of the FeedMix software. It has a title bar "FeedMix v1.0 (Data Tersimpan)". The main area contains a table for "Komposisi Ransum (Kg)" with columns for "Nama Bahan", "Kompositi % Ransum", and "Kompositi Kg Ransum". The table lists various ingredients like "Junkoil Kadela", "Junkoil Kelapa", "Gulek", "Mengkil kelapa", "Tepung Alpafa (17% prot.)", "Tepung Ragi (20% prot.)", and "Tepung Daging". To the right, there's a "Komponen Nutrisi" (Nutrient Components) table with rows for Energi Metabolis, Protein Kotor, Lemak Kotor, Serat Kotor, Kalsium, Phosforida, Na, Cl, Mg, Fe, Vitamin E-12, Vitamin E, Nitrogen, Lisis, Sistin, and Triptofan. Above the nutrient table, it says "Pengguna : Budi Apriyanto, Rabu, 28 Januari 2004 13.44". On the right side, there are buttons for "Cetak" (Print), "Reformulate", and "Input". At the bottom left, there are fields for "Harga Ransum" (Feed Cost) and "Rp/Kg". A copyright notice "(C) 2003 By Budi Apriyanto Kusumawardhani" is at the bottom.

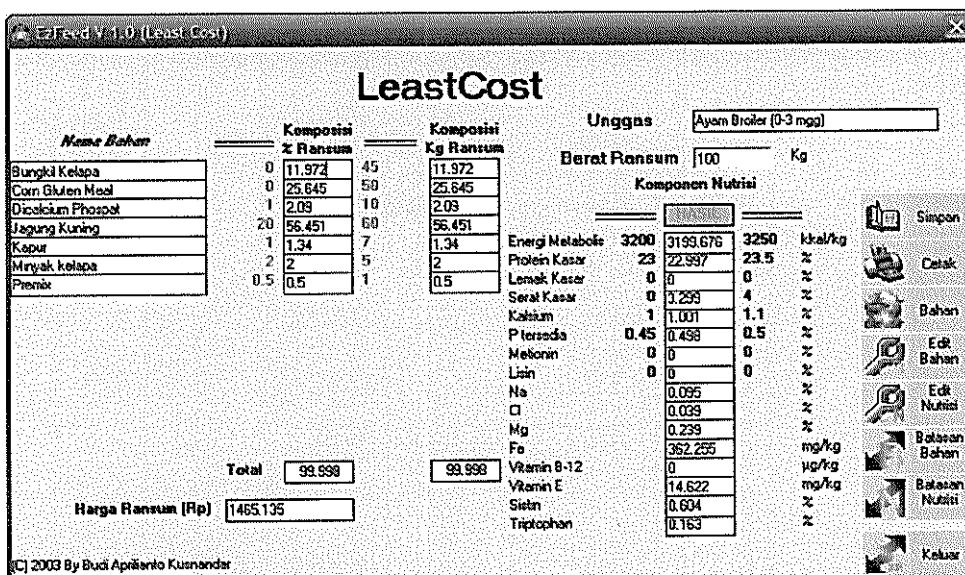
Gambar 12. *Form Data Tersimpan*

Formulasi Ransum Secara *Least Cost*

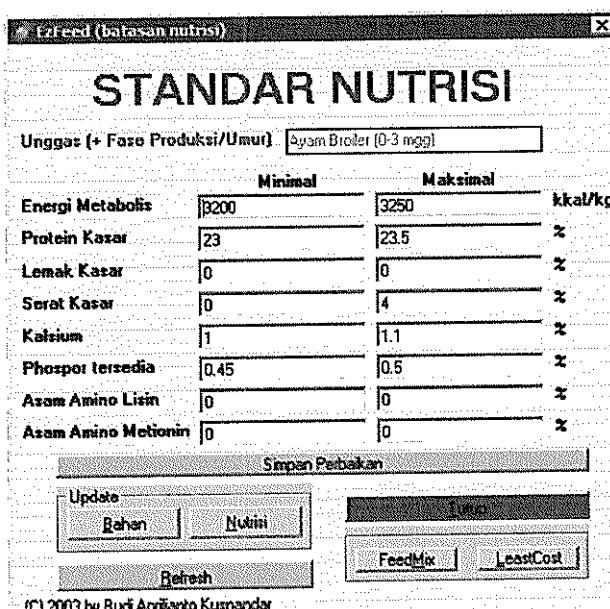
Metode formulasi ransum secara *Least Cost* merupakan inti dari program ini. Pada dasarnya, ransum diformulasikan menggunakan metode *linear programming* dengan tujuan meminimalkan harga ransum.

Gambar 13 merupakan contoh tampilan hasil formulasi ransum dengan menggunakan program EzFeed. Ransum tersebut untuk memenuhi kebutuhan nutrisi ayam broiler dengan energi metabolismis sebesar 3200 kkal/kg, protein kasar 23 %, serat kasar 4 % (maksimal), kalsium 1% dan phospor tersedia sebesar 0,45 % dari 7 macam bahan pakan. Dari ketujuh bahan tersebut, diperoleh suatu ransum dengan kandungan nutrisi antara lain : energi metabolismis sebesar 3199 kkal/kg, protein kasar sebesar 22,997 %, serat kasar sebesar 3,299 %, kalsium sebesar 1,001 % dan phospor tersedia sebesar 0,498 %. Sedangkan harga termurah yang berhasil dicapai adalah sebesar Rp 1.465,135 tiap kg *as fed* ransum.

Gambar 14 dan 15 merupakan standar kebutuhan nutrisi ayam broiler yang digunakan dalam formulasi, dimana semakin ketat jarak (*range*) kebutuhan yang dimasukkan, maka semakin kecil kemungkinan kebutuhan tersebut akan terpenuhi. Oleh karena itu, standar kebutuhan tersebut diberi jarak yang agak longgar agar program dapat dengan lebih mudah dalam menghitungnya. Selain itu, batasan minimal dan maksimal juga harus diisi untuk membatasi program agar susunan ransum yang terbentuk seimbang dan untuk mengantisipasi efek negatif pada ternak apabila kelebihan salah satu nutrisi tertentu.



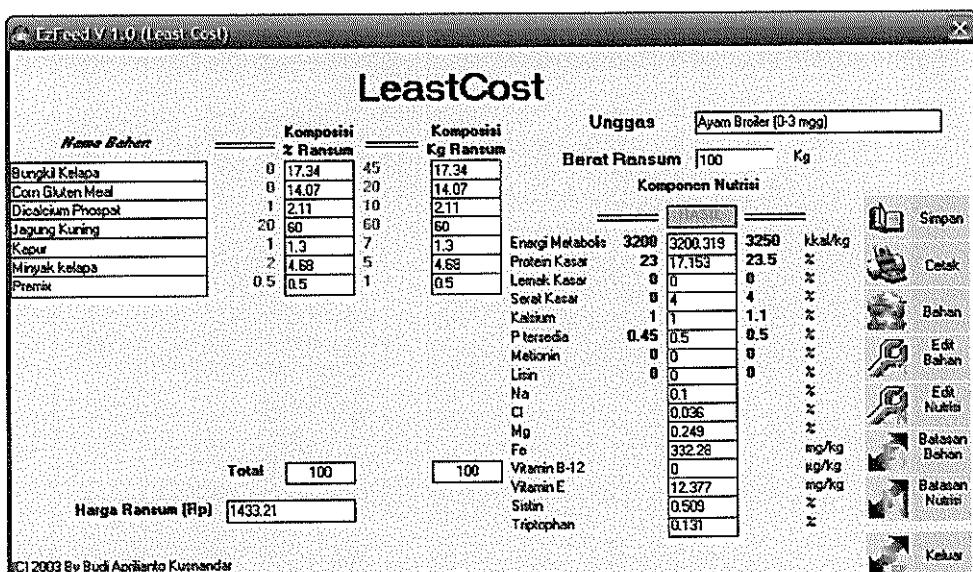
Gambar 13. Contoh Formula Ransum Ayam Broiler dengan Menggunakan 7 Bahan Pakan



Gambar 14. Standar Kebutuhan Nutrisi Ayam Broiler (NRC '94)

Pada saat batasan penggunaan maksimal *Corn Gluten Meal* (CGM) menjadi 20 %, tampak terjadi perubahan yang sangat berarti dalam komposisi ransum tersebut. Terutama pada bungkil kelapa yang awalnya sebesar 11,972 % menjadi sebesar 17,34 % dan CGM yang awalnya sebesar 56,451 % maka setelah dibatasi menjadi maksimal 20 % yang tercapai adalah sebesar 14,307 %. Selain itu, dari segi nutrisi juga berubah dimana kandungan energi metabolismis, serat kasar, kalsium dan fosfor tersedia mengalami sedikit peningkatan sedangkan kandungan protein kasar

menurun dari 22,997 % menjadi 17,153 % sehingga kebutuhan nutrisi tidak terpenuhi (perlu ditambahkan bahan pakan sumber protein). Selain itu, harga yang tercapai juga mengalami penurunan dari Rp. 1.465,135 tiap kg ransum menjadi hanya Rp.1.433,21 tiap kg ransum.



Gambar 15. Hasil Formulasi Ransum Setelah Diubah Batasan maksimalnya

Untuk menguji program formulasi ransum ini, maka program formulasi ransum Feed Mania digunakan sebagai program pembanding. Dengan bahan dan kebutuhan nutrisi yang sama, hasil yang diperoleh tidak terlalu jauh berbeda (Gambar 16 dan 17).

DIET SPEC : AYAM1						Standard mix weight = 1.000t		
INCLUDED	Incl%	Min%	Max%	Amount	\$	LOW.....cents/kg.....	HIGH	
CGM	26.55			265.514 kg	5310.28	560.031	2000.000	2875.785
KELAP	12.14			45.00 121.41 kg	1092.69	644.555	900.000	1320.001
DCP	2.10	1.00	10.00	20.953 kg	104.76	-46066.940	500.000	687.843
KAP	1.34	1.00	7.00	13.351 kg	133.51	673.721	1000.000	816546
NINYA	2.00	2.00	5.00	20.0 kg	800.00	2533.630	4000.000	<14.664>
PREMI	0.50	0.50	1.00	5.0 kg	350.00	261.590	7000.000	<67.384>
JKK	55.38	20.00	60.00	553.773 kg	6922.16	624.125	1250.000	1681.377
TOTAL MIX 1000.0 kg						14713.40	LC/PK	=14713.395

Space Bar = next screen Esc = exit
Auto bag, Display, Formulate, G, Ing. rest., List ing., Nut. rest., Print, S, V

Gambar 16. Komposisi Ransum Hasil Formulasi dengan Feed Mania

DIET SPEC : AYAM1		Standard mix weight = 1.000t		
Nutrient	Amount	RATIO	Marg \$/unit	Restrictions
ME	3200 KC/kg		15091.56	3200 < ME < 3250
PK	23.500 %		1184.40	23.00 < PK < 23.50
LK	0.000 %			
SK	3.312 %			SK < 4.000
CA	1.000 %		19.43	1.000 < CA < 1.100
P	0.500 %		10.11	0.450 < P < 0.500
LYS	0.000 %			
MET	0.000 %			

Space Bar = next screen Esc = exit
Auto bag, Display, Formulate, G, Ing. rest., List ing., Nut. rest., Print, S, V

Gambar 17. Komposisi Nutrisi Hasil Formulasi dengan Feed Mania

Formulasi Ransum Secara *Feed Mix*

Formulasi ransum secara *Feed Mix* merupakan fasilitas yang dapat digunakan untuk menyusun ransum sederhana dengan menggunakan metode perhitungan coba-coba (*Trial and Error Method*).

Lain halnya dengan *Least Cost* yang menggunakan metode program linier, fasilitas *Feed Mix* ini tidak mempergunakan faktor pembatas yang ada dalam basis data. Pengguna cukup memilih bahan yang akan digunakan dalam formulasi, kemudian klik tombol *Feed Mix* maka akan muncul *form Feed Mix*. Selanjutnya pengguna dapat mengisikan berat ransum (dalam kg) dan mengisikan berapa persen (%) komposisi bahan tersebut dalam ransum, maka program akan langsung menghitung secara otomatis kandungan nutrisi ransum, berat bahan dalam ransum dan harga tiap kilogram ransum. Tampilan dari *form Feed Mix* adalah seperti tampak pada gambar 18:

FeedMix

Nama Bahan	Komposisi % Ransum	Komposisi Kg Ransum	Berat Ransum [100 Kg]
Bungkil Kedelai	24	24.000	
Dicalcium Phospat	2	2.000	
Jagung Kuning	43	42.000	
Limbah pembuatan roti	8	8.000	
Minyak kelapa	3,5	3.500	
Premix	0,5	0.500	
Tepung Allata (17 % protein)	20	20.000	
Total	100	100.000	
Sisaan	0.000		
Harga Ransum	1549,00	Rp/Kg	

Komponen Nutrisi	
Energi Metabolis	2792,16 kkal/kg
Protein Kasar	18,47%
Lemak Kasar	2,455%
Serat Kasar	7,52%
Kalsium	0,8164%
Phosfor	0,6076%
Na	0,2452%
Cl	0,2212%
Mg	0,2874%
Fe	320,24 mg/kg
Vitamin B-12	1,12 µg/kg
Vitamin E	41,78 mg/kg
Metionin	0,391%
Zinc	1,091%
Sistina	0,4242%
Tryptophan	0,2736%

 Simpan
 Cetak
 Bahan
 Edit Bahan
 Edit Nutrisi
 Batasan Bahan
 Batasan Nutrisi
 Keluar

(C) 2003 By Budi Apriyanto Kusmendar

Gambar 18. Form Formulasi Ransum Secara Feed Mix

Setelah formulasi selesai, seperti halnya pada fasilitas *Least Cost* pengguna dapat mencetak ataupun menyimpan hasil formulasi untuk selanjutnya dapat dilakukan perbaikan.

KESIMPULAN DAN SARAN

Kesimpulan

Metode program linear merupakan salah satu metode yang dapat digunakan dalam formulasi ransum unggas. Dengan bantuan program komputer, penyusunan ransum dapat dilakukan dengan lebih mudah, cepat dan akurat dibandingkan dengan penyelesaian secara manual.

Program formulasi ransum ternak unggas ini diberi nama EzFeed Versi 1.0. Program ini dibuat berdasarkan prinsip perhitungan program linier dengan fungsi tujuan meminimalkan harga ransum (*Least Cost*). Program ini dapat memformulasikan beberapa bahan pakan yang sesuai dengan keinginan pengguna.

EzFeed dibuat dengan menggunakan sistem pemrograman Microsoft Visual Basic 6.0 yang bersifat *Object Oriented Programming* (OOP), sehingga hasil atau tampilan dari program ini dapat lebih mudah dikenali oleh pengguna sehingga pengoperasiannya akan dirasakan lebih mudah.

Saran

Untuk perbaikan dan pengembangan dari program ini, perlu dilakukan beberapa hal :

1. Penambahan fasilitas opsi dimana pengguna dapat dengan lebih leluasa menentukan faktor pembatas yang diperhitungkan dalam formulasi ransum;
2. Mengintegrasikan sistem formulasi ransum ternak unggas ini dengan sistem formulasi ransum ternak lain seperti ternak ruminansia; dan
3. Menambahkan fasilitas analisa dimana didalamnya terdapat informasi mengenai *shadow price* dan beberapa informasi lainnya.

DAFTAR PUSTAKA

- Alam, M. A. J. 1999. Manajemen Database dengan Microsoft Visual Basic 6.0. PT. Gramedia Pustaka Utama. Jakarta.
- Anggorodi, H. R. 1995. Nutrisi Aneka Ternak Unggas. PT. Gramedia Pustaka Utama. Jakarta.
- Crampton, E. W. and L. E. Harris. 1969. Applied Animal Nutrition. 2nd Ed. W.H. Freeman and Company. San Fransisco.
- Dewobroto, W. 2003. Aplikasi Sain dan Teknik dengan Visual Basic 6.0. Elex Media Komputindo. Jakarta.
- Ensminger, M. E. 1992. Poultry Science (Animal Agriculture Series). Interstate Publishers, Inc. Danville. Illinois.
- Ensminger, M. E., J. E. Oldfield and W. W. Heinemann. 1990. Feed and Nutrition : Formerly, Nutrition, Complete. 2nd Ed. The Ensminger Publishing Company. California. USA.
- National Research Council. 1994. Nutrient Requirement of Poultry, 9th Revised Ed. National Academy Press. Washington.
- Pandia, H. 2002. Visual Basic Tingkat Lanjut. Penerbit ANDI. Yogyakarta.
- Rasyaf, M. 1984. Program Liniar untuk Industri Ransum Ternak. Yayasan Kanisius. Yogyakarta.
- Rasyaf, M. 1994. Beternak Ayam Petelur. Penebar Swadaya. Jakarta.
- Scott, M. L., M. C. Nesheim and R. J. Young. 1969. Nutrition of The Chicken. M. L. Scott and Associates. Ithaca, New York.
- Soekartawi. 1992. Linear Programming, Teori dan Aplikasinya Khususnya dalam Bidang Pertanian. Rajawali Press. Jakarta.
- Supranto, J. 1983. Linear Programming. Lembaga Penerbit Fakultas Ekonomi Universitas Indonesia. Jakarta.

- Suharti, S. 1997. Formulasi Ransum Sapi Perah dengan Visual Basic. Skripsi. Fakultas Peternakan. Institut Pertanian. Bogor.
- Williamson, G. dan W. J. A. Payne. 1993. Pengantar Peternakan di Daerah Tropis. Gadjah Mada University Press. Yogyakarta.
- Wikipedia. 2004. Computer Program. http://www.ezresult.com/article/Computer_program. [5 Mei 2004].

LAMPIRAN 1
Listing Program EzFeed V. 1.0

FORM MASUK (LOGIN)

```
Private Sub Form_Load()
    Text1.Text = "USER"
End Sub

Private Sub batal_Click()
    End
End Sub

Private Sub Command1_Click()
    masuk
End Sub

Private Sub Text1_KeyPress(KeyAscii As Integer)
    If (KeyAscii = vbKeyReturn) Then
        masuk
    End If
End Sub

Private Sub Timer1_Timer()
    ProgressBar1.Value = ProgressBar1.Value + 5
    Label1.Caption = ProgressBar1.Value + 5 & "%"
    If ProgressBar1.Value = 100 Then
        Timer1.Enabled = False
        Picture1.Visible = True
        Label2.Visible = True
    End If
End Sub

Private Sub Timer2_Timer()
    If Label2.ForeColor = &HC0& Then
        Label2.ForeColor = &HFFFFFF
    ElseIf Label2.ForeColor = &HFFFFFF Then
        Label2.ForeColor = &HC0&
    End If
End Sub

Private Sub masuk()
    MousePointer = vbHourglass
    MDIForm1.Show
    MDIForm1.StatusBar1.Panels(4).Text = "Pengguna : " & Text1.Text & ". "
    Unload Me
End Sub
```

FORM UTAMA

```
Private Sub MDIForm_load()
    Adodc1.Refresh
    Adodc2.Refresh
    MousePointer = 0
    If Lstterpilih.ListCount = 0 Then
        mbatasan.Enabled = False
        mformulasi.Enabled = False
    End If
    SSTabl.Tab = 0
    cmbounggas.Text = "" & (Adodc2.Recordset.Fields("unggas"))
End Sub

Private Sub berat_KeyPress(KeyAscii As Integer)
    If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii = vbKeyBack Or KeyAscii = Asc(".")) Then
        KeyAscii = 0
    End If
End Sub
```

```
Private Sub DataGrid1_dblClick()
    pilih_bahan
End Sub

Private Sub ipb_Click()
    RetVal = ShellExecute(Me.hWnd, vbNullString, "http://www.ipb.ac.id/", vbNullString, Left$(CurDir$, 3), SW_SHOWNORMAL)
End Sub

Private Sub Lstterpilih_DblClick()
    Hapus_Pilihan
End Sub

Private Sub mbatbah_Click()
    frmbatbah.Show
End Sub

Private Sub mbatnut_Click()
    frmbatnut.Show
End Sub

Private Sub mfeedmix_Click()
    frmfeedmix.Show
End Sub

Private Sub minform_Click()
    HHShowContents Me.hWnd
End Sub

Private Sub minformasi_Click()
    status.Show
End Sub

Private Sub mingredient_Click()
    frmredit.Show
End Sub

Private Sub mkalkulator_Click()
    Calculator.Show
End Sub

Private Sub mkeluar_Click()
    get_out
End Sub

Private Sub mleastcost_Click()
    If MDIForm1.Lstterpilih.ListCount = 1 Then
        Beep
        MsgBox "Bahan yang dipilih minimal 2", vbCritical, "EzFeed V 1.0"
    Else
        frmLeastCost.Show
    End If
End Sub

Private Sub mNutrient_Click()
    frmbatnutupdate.Show
End Sub

Private Sub mopen_Click()
    buka_data_tersimpan
End Sub

Private Sub mseputar_Click()
    frmtentang.Show
End Sub

Private Sub mtabbahan_Click()
    SSTabl.Tab = 0
End Sub
```

```

Private Sub mtool_Click()
    SSTab1.Tab = 1
End Sub

Private Sub tblmaju_Click()
    Adodc3.Recordset.MoveNext
    If Adodc3.Recordset.EOF Then
        Adodc3.Recordset.MovePrevious
        End If
    End Sub

Private Sub tblmundur_Click()
    Adodc3.Recordset.MovePrevious
    If Adodc3.Recordset.BOF Then
        Adodc3.Recordset.MoveNext
        End If
    End Sub

Private Sub Toolbar1_ButtonClick(ByVal Button As MSComctlLib.Button)
    Dim i As Integer
    Select Case Button.Key
        Case Is = "tbbantuan"
            HHShowContents Me.hWnd
        Case Is = "tbkalkulator"
            Calculator.Show
        Case Is = "openfile"
            buka_data_tersimpan
        Case Is = "tbnutrisi"
            frmbatnutupdate.Show
        Case Is = "tbbahan"
            frmedit.Show
        Case Is = "tbfeedmix"
            If
                MDIForm1.Lstterpilih.ListCount = 0 Then
                    Beep
                    MsgBox "Anda belum memilih bahan !", vbCritical, "EzFeed V 1.0"
                    MDIForm1.Show
                Else
                    frmfeedmix.Show
                End If
            Case Is = "tbleastcost"
                If
                    MDIForm1.Lstterpilih.ListCount = 0 Then
                        Beep
                        MsgBox "Anda belum memilih bahan !", vbCritical, "EzFeed V 1.0"
                        MDIForm1.Show
                    Else
                        frmLeastCost.Show
                    End If
                Case Is = "tbkeluar"
                    get_out
                End Select
            End Sub

Private Sub Toolbar2_ButtonClick(ByVal Button As MSComctlLib.Button)
    Select Case Button.Key
        Case Is = "pilih"
            pilih_bahan
        Case Is = "refresh"
            Adodc1.Refresh
            Adodc2.Refresh
        Case Is = "hapus"
            Hapus_Pilihan
        Case Is = "kosong"
            mbatasan.Enabled = False
            mformulasi.Enabled = False
            Lstterpilih.Clear
    End Select
End Sub

Private Sub Toolbar2_ButtonMenuClick(ByVal ButtonMenu As MSComctlLib.ButtonMenu)
    Select Case ButtonMenu.Key
        Case "ingredient"
            If
                MDIForm1.Lstterpilih.ListCount = 0 Then
                    Beep
                    MsgBox "Anda belum memilih bahan untuk formulasi", 1049104, "EzFeed V 1 (Data)"
                Else
                    frmbatbah.Show
                End If
            Case "nutrient"
                frmbatnut.Show
        End Select
    End Sub

Private Sub Toolbar1_ButtonMenuClick(ByVal ButtonMenu As MSComctlLib.ButtonMenu)
    Select Case ButtonMenu.Key
        Case "batnut"
            frmbatnutupdate.Show
        Case "bahan"
            frmedit.Show
    End Select
End Sub

Private Sub DataGrid1_HeadClick(ByVal ColIndex As Integer)
    Dim rs As ADODB.Recordset
    Set rs = Adodc1.Recordset

    If rs.Sort <>
        DataGrid1.Columns(ColIndex).DataField & " DESC" Then
            rs.Sort =
        DataGrid1.Columns(ColIndex).DataField & " DESC"
    Else
        rs.Sort =
        DataGrid1.Columns(ColIndex).DataField & " ASC"
    End If
End Sub

Private Sub pilih_bahan()
    Dim i, j As Integer
    Dim ketemu As Integer
    ketemu = 0
    For i = 0 To Lstterpilih.ListCount - 1
        DataGrid1.Col = 1
        Lstterpilih.ListIndex = i
        If DataGrid1.Text = Lstterpilih.Text Then
            ketemu = 1
            Exit For
        End If
    Next i
    If Lstterpilih.ListCount >= 15 Then
        Beep
        MsgBox "Bahan Yang Dapat Dipilih Maksimal 15 !", vbCritical, "EzFeed V 1.0"
    ElseIf ketemu = 1 Then
        Beep
        MsgBox "Bahan Yang Dipilih Tidak Boleh Sama!", vbCritical, "EzFeed V 1.0"
    Else
        DataGrid1.Col = 1
        Lstterpilih.AddItem DataGrid1.Text
    End If
End Sub

```

```

        StatusBar1.Panels(3).Text = "Bahan
Terpilih = " & Lstterpilih.ListCount
        End If
        mbatasan.Enabled = True
        mformulasi.Enabled = True
End Sub

Private Sub Hapus_Pilihan()
        Dim pesan, judul_pesan
        If Lstterpilih.ListIndex = -1
Then
        Beep
        pesan = "Tidak Ada Bahan
Terpilih Untuk Dihapus !"
        judul_pesan = "EzFeed V 1.0"
        MsgBox pesan, vbCritical,
judul_pesan
        Exit Sub
        End If
        Lstterpilih.RemoveItem
Lstterpilih.ListIndex
        If Lstterpilih.ListCount = 0
Then
        mbatasan.Enabled = False
        mformulasi.Enabled = False
        End If
End Sub

Private Sub buka_data_tersimpan()
On Error GoTo salah
Dim m As Integer

List1.Clear
For m = 0 To 14
frmsaves.Bahan(m).Text = ""
frmsaves.Komposisi(m).Text = ""
frmsaves.Kg(m).Text = ""
Next m
For m = 0 To 15
frmsaves.Nut(m).Text = ""
Next m
frmsaves.judul(1).Caption = ""
frmsaves.User.Caption = ""
frmsaves.jam.Caption = ""
frmsaves.tgl.Caption = ""
frmsaves.jumpiltxt.Text = ""
jumpiltxt.Text = ""
frmsaves.fase.Caption = ""
frmsaves.berat.Text = ""
frmsaves.Harga_ransum.Text = ""
Dim file As String
cdlg.DialogTitle = "Buka Data."
cdlg.MaxFileSize = 16384
cdlg.FileName = ""
cdlg.Filter = "File EzFeed|*.Aan"
cdlg.ShowOpen      = 1
If cdlg.FileName = "" Then Exit Sub
file = cdlg.FileName
Dim A As String
Dim x As String
Dim i As Integer
Dim j As Integer
Dim k As Integer
Dim jmp As String
Dim Bhn As String
Dim Kmp As String
Dim Klg As String
On Error GoTo error
Open file For Input As #1
Do Until EOF(1)
Input #1, A$
List1.AddItem A$
Loop
Close 1

For i = 0 To 15
frmsaves.Nut(i).Text =
List1.List(i)
Next i
frmsaves.jumpiltxt.Text =
List1.List(16)
jumpiltxt.Text = List1.List(16)
jmp = Val(jumpiltxt.Text)
frmsaves.judul(1).Caption =
List1.List(17)
frmsaves.berat.Text = List1.List(18)
frmsaves.Harga_ransum.Text =
List1.List(19)
frmsaves.User.Caption =
List1.List(20)
frmsaves.tgl.Caption =
List1.List(21)
frmsaves.jam.Caption =
List1.List(22)
frmsaves.fase.Caption =
List1.List(23)

frmsaves.Bahan(0).Text =
List1.List(24)
frmsaves.Komposisi(0).Text =
List1.List(25)
frmsaves.Kg(0).Text = List1.List(26)
frmsaves.Bahan(1).Text =
List1.List(27)
frmsaves.Komposisi(1).Text =
List1.List(28)
frmsaves.Kg(1).Text = List1.List(29)
frmsaves.Bahan(2).Text =
List1.List(30)
frmsaves.Komposisi(2).Text =
List1.List(31)
frmsaves.Kg(2).Text = List1.List(32)
frmsaves.Bahan(3).Text =
List1.List(33)
frmsaves.Komposisi(3).Text =
List1.List(34)
frmsaves.Kg(3).Text = List1.List(35)
frmsaves.Bahan(4).Text =
List1.List(36)
frmsaves.Komposisi(4).Text =
List1.List(37)
frmsaves.Kg(4).Text = List1.List(38)
frmsaves.Bahan(5).Text =
List1.List(39)
frmsaves.Komposisi(5).Text =
List1.List(40)
frmsaves.Kg(5).Text = List1.List(41)
frmsaves.Bahan(6).Text =
List1.List(42)
frmsaves.Komposisi(6).Text =
List1.List(43)
frmsaves.Kg(6).Text = List1.List(44)
frmsaves.Bahan(7).Text =
List1.List(45)
frmsaves.Komposisi(7).Text =
List1.List(46)
frmsaves.Kg(7).Text = List1.List(47)
frmsaves.Bahan(8).Text =
List1.List(48)
frmsaves.Komposisi(8).Text =
List1.List(49)
frmsaves.Kg(8).Text = List1.List(50)
frmsaves.Bahan(9).Text =
List1.List(51)
frmsaves.Komposisi(9).Text =
List1.List(52)
frmsaves.Kg(9).Text = List1.List(53)
frmsaves.Bahan(10).Text =
List1.List(54)

```

```

        frmsaves.Komposisi(10).Text =
List1.List(55)
        frmsaves.Kg(10).Text =
List1.List(56)
        frmsaves.Bahan(11).Text =
List1.List(57)
        frmsaves.Komposisi(11).Text =
List1.List(58)
        frmsaves.Kg(11).Text =
List1.List(59)
        frmsaves.Bahan(12).Text =
List1.List(60)
        frmsaves.Komposisi(12).Text =
List1.List(61)
        frmsaves.Kg(12).Text =
List1.List(62)
        frmsaves.Bahan(13).Text =
List1.List(63)
        frmsaves.Komposisi(13).Text =
List1.List(64)
        frmsaves.Kg(13).Text =
List1.List(65)
        frmsaves.Bahan(14).Text =
List1.List(66)
        frmsaves.Komposisi(14).Text =
List1.List(67)
        frmsaves.Kg(14).Text =
List1.List(68)
        For j = 0 To Val(jumpiltxt.Text) - 1
            frmsaves.Bahan(j).Visible = True
            frmsaves.Komposisi(j).Visible =
True
            frmsaves.Kg(j).Visible = True
        Next j
        If frmsaves.judul(1).Caption =
"FeedMix" Then
            frmsaves.tblfeedmix.Visible =
True
        Else
            frmsaves.tblleastcost.Visible =
True
        End If
        frmsaves.Show
        Exit Sub
error:
salah:
End Sub

Private Sub get_out()
    Beep
    response = MsgBox("Anda Yakin
Ingin Keluar", 1048612, "EzFeed V 1.0
(Keluar)")
    If response = 6 Then
        MDIForm1.WindowState = 0
        Do
            MDIForm1.Top = MDIForm1.Top +
100
            MDIForm1.Move MDIForm1.Left,
MDIForm1.Top
            DoEvents
            Loop Until MDIForm1.Top >
Screen.Height - 100
        End
    Else
        MDIForm1.Show
    End If
End Sub

```

UPDATE NUTRISI

```

Private Sub cmdAdd_Click()
    cmdAdd.Enabled = False
    cmdDelete.Enabled = False
    cmdRefresh.Enabled = False
    txtFieldsunggas(0).Enabled = True
    Dim i As Single
    silahkan.Visible = True
    Adodc1.Recordset.AddNew
    For i = 1 To 16
        txtFields(i).Text = 0
    Next i
    txtFieldsunggas(0).SetFocus
End Sub

Private Sub cmdDelete_Click()
    Adodc1.Recordset.Delete
    Adodc1.Recordset.MoveNext
    Adodc1.Refresh
End Sub

Private Sub cmdRefresh_Click()
    Adodc1.Refresh
End Sub

Private Sub cmdUpdate_Click()
On Error GoTo salah
    If Val(txtFields(1).Text) >
Val(txtFields(2).Text) Then
        MsgBox "EM Minimal Harus Lebih
Kecil dibandingkan Maksimal !",
vbCritical, "EzFeed V 1.0"
        txtFields(2).Text =
Val(txtFields(1).Text) + 50
    ElseIf Val(txtFields(3).Text) >
Val(txtFields(4).Text) Then
        MsgBox "PK Minimal Harus Lebih
Kecil dibandingkan Maksimal !",
vbCritical, "EzFeed V 1.0"
        txtFields(4).Text =
Val(txtFields(3).Text) + 0.1
    ElseIf Val(txtFields(5).Text) >
Val(txtFields(6).Text) Then
        MsgBox "LK Minimal Harus Lebih
Kecil dibandingkan Maksimal !",
vbCritical, "EzFeed V 1.0"
        txtFields(6).Text =
Val(txtFields(5).Text) + 0.01
    ElseIf Val(txtFields(7).Text) >
Val(txtFields(8).Text) Then
        MsgBox "SK Minimal Harus Lebih
Kecil dibandingkan Maksimal !",
vbCritical, "EzFeed V 1.0"
        txtFields(8).Text =
Val(txtFields(7).Text) + 0.01
    ElseIf Val(txtFields(9).Text) >
Val(txtFields(10).Text) Then
        MsgBox "Ca Minimal Harus Lebih
Kecil dibandingkan Maksimal !",
vbCritical, "EzFeed V 1.0"
        txtFields(10).Text =
Val(txtFields(9).Text) + 0.01
    ElseIf Val(txtFields(11).Text) >
Val(txtFields(12).Text) Then
        MsgBox "P tersedia Minimal Harus
Lebih Kecil dibandingkan Maksimal !",
vbCritical, "EzFeed V 1.0"
        txtFields(12).Text =
Val(txtFields(11).Text) + 0.01
    ElseIf Val(txtFields(13).Text) >
Val(txtFields(14).Text) Then
        MsgBox "Lisin Minimal Harus
Lebih Kecil dibandingkan Maksimal !",
vbCritical, "EzFeed V 1.0"
        txtFields(14).Text =
Val(txtFields(13).Text) + 0.01
    ElseIf Val(txtFields(15).Text) >
Val(txtFields(16).Text) Then

```

```

        MsgBox "Metionin Minimal Harus
Lebih Kecil dibandingkan Maksimal !",
vbCritical, "EzFeed V 1.0"
        txtFields(16).Text = Val
(txtFields(15).Text) + 0.01
    Else
        txtFieldsunggas(0).Enabled = False
        cmdAdd.Enabled = True
        cmdDelete.Enabled = True
        cmdRefresh.Enabled = True
        MDIForm1.Adodc2.Refresh
        silahkan.Visible = False
        Adodc1.Recordset.Update
    End If
        Adodc1.Refresh
    Exit Sub
    salah:
        MsgBox "Data Sudah ada dan semua
data harus diisi", vbCritical, "EzFeed V
1.0 (Isi data)"
End Sub

Private Sub cmdClose_Click()
    MousePointer = 11
    MDIForm1.Adodc2.Refresh
    Unload Me
    MousePointer = 0
End Sub

Private Sub adodc1_Reposition()
    Screen.MousePointer = vbDefault
    On Error Resume Next
    Adodc1.Caption = "" &
(Adodc1.Recordset.Fields("unggas"))
End Sub

Private Sub adodc1_Validate(Action As
Integer, Save As Integer)
    Select Case Action
        Case vbDataActionMoveFirst
        Case vbDataActionMovePrevious
        Case vbDataActionMoveNext
        Case vbDataActionMoveLast
        Case vbDataActionAddNew
        Case vbDataActionUpdate
        Case vbDataActionDelete
        Case vbDataActionFind
        Case vbDataActionBookmark
        Case vbDataActionClose
    End Select
    Screen.MousePointer = 0
End Sub

Private Sub Form_Deactivate()
    Unload Me
End Sub

Private Sub tblmaju_Click()
    Adodc1.Recordset.MoveNext
    If Adodc1.Recordset.EOF Then
        Adodc1.Recordset.MovePrevious
    End If
End Sub

Private Sub tblmundur_Click()
    Adodc1.Recordset.MovePrevious
    If Adodc1.Recordset.BOF Then
        Adodc1.Recordset.MoveNext
    End If
End Sub

Private Sub Timer1_Timer()
    If silahkan.ForeColor = &HC0& Then
        silahkan.ForeColor = &HFFFFFF
    ElseIf silahkan.ForeColor = &HFFFFFF
    Then
        silahkan.ForeColor = &HC0&
    End If
End Sub

Private Sub txtFields_KeyPress(Index As
Integer, KeyAscii As Integer)
    If Not (KeyAscii >= Asc("0") And
KeyAscii <= Asc("9") Or KeyAscii =
vbKeyBack Or KeyAscii = Asc(".")) Then
        Beep
        KeyAscii = 0
    End If
End Sub

Private Sub txtFieldsunggas_Change(Index
As Integer)
    urutan.Caption = "Data : " &
(Adodc1.Recordset.AbsolutePosition) &
"/" & (Adodc1.Recordset.RecordCount)
End Sub

Private Sub txtFieldsunggas_KeyPress(Index As
Integer, KeyAscii As Integer)
    If (KeyAscii = vbKeyReturn) Then
        txtFields(1).SetFocus
    End If
End Sub


```

UPDATE BAHAN

```

Private Sub Form_Load()
    cmbinfo.Visible = False
    Text1.Visible = True
    cmbinfo.AddItem ("Sumber Energi")
    cmbinfo.AddItem ("Sumber Protein")
    cmbinfo.AddItem ("Sumber Lemak")
    cmbinfo.AddItem ("Sumber Serat")
    cmbinfo.AddItem ("Sumber Kalsium")
    cmbinfo.AddItem ("Sumber Phosphor")
    cmbinfo.AddItem ("Sumber Asam Amino
Lisin")
    cmbinfo.AddItem ("Sumber Asam Amino
Metionin")
    cmbinfo.AddItem ("Bahan Suplemen")
End Sub

Private Sub cmdAdd_Click()
    cmdDelete.Enabled = False
    cmdRefresh.Enabled = False
    cmdAdd.Enabled = False
    Dim i As Single
    Adodc1.Recordset.AddNew
    silahkan.Visible = True
    txtFields1(1).Visible = True
    cmbinfo.Visible = True
    Text1.Visible = False
    cmbinfo.Text = "Sumber energi"
    For i = 2 To 19
        txtFields(i).Text = 0
    Next i
    min.Text = 0
    max.Text = 100
    txtFields1(0).SetFocus
End Sub

Private Sub cmdDelete_Click()
    Adodc1.Recordset.Delete
    Adodc1.Recordset.MoveNext
    Adodc1.Refresh
End Sub

```

```

Private Sub cmdRefresh_Click()
    Adodc1.Refresh
End Sub

Private Sub cmdUpdate_Click()
    On Error GoTo salah
    If Val(min.Text) > Val(max.Text) Then
        MsgBox "Batasan penggunaan minimal harus lebih kecil !", vbCritical,
        "EzFeed V 1.0 (min vs max)"
        max.Text = 100
    Else
        silahkan.Visible = False
        cmbinfo.Visible = False
        Text1.Visible = True
        MDIForm1.Adodc1.Refresh
    End If
    cmdDelete.Enabled = True
    cmdRefresh.Enabled = True
    cmdAdd.Enabled = True
    Adodc1.Refresh
Exit Sub
salah:
    MsgBox "Data sudah ada dan semua data harus diisi", vbCritical, "EzFeed V 1.0 (Isi data)"
End Sub

Private Sub cmdClose_Click()
    MousePointer = 11
    MDIForm1.Adodc1.Refresh
    Unload Me
    MousePointer = 0
End Sub

Private Sub adodc1_Reposition()
    Screen.MousePointer = vbDefault
    On Error Resume Next
    Adodc1.Caption = "" &
    (Adodc1.Recordset.Fields("Nama_Bahan"))
End Sub

Private Sub adodc1_Validate(Action As Integer, Save As Integer)
    Select Case Action
        Case vbDataActionMoveFirst
        Case vbDataActionMovePrevious
        Case vbDataActionMoveNext
        Case vbDataActionMoveLast
        Case vbDataActionAddNew
        Case vbDataActionUpdate
        Case vbDataActionDelete
        Case vbDataActionFind
        Case vbDataActionBookmark
        Case vbDataActionClose
    End Select
    Screen.MousePointer = 0
End Sub

Private Sub Form_Deactivate()
    Unload Me
End Sub

Private Sub max_KeyPress(KeyAscii As Integer)
    If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii =
    vbKeyBack Or KeyAscii = Asc(".")) Then
        Beep
        KeyAscii = 0
    End If
    If (KeyAscii = vbKeyReturn) Then
        cmdUpdate.SetFocus
    End If
End Sub

Private Sub min_KeyPress(KeyAscii As Integer)
    If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii =
    vbKeyBack Or KeyAscii = Asc(".")) Then
        Beep
        KeyAscii = 0
    End If
    If (KeyAscii = vbKeyReturn) Then
        max.SetFocus
    End If
End Sub

Private Sub tblmaju_Click()
    On Error GoTo 0
    Adodc1.Recordset.MoveNext
    If Adodc1.Recordset.EOF Then
        Adodc1.Recordset.MovePrevious
    End If
End Sub

Private Sub tblmundur_Click()
    On Error GoTo 0
    Adodc1.Recordset.MovePrevious
    If Adodc1.Recordset.BOF Then
        Adodc1.Recordset.MoveNext
    End If
End Sub

Private Sub Timer1_Timer()
    If silahkan.ForeColor = &HC0& Then
        silahkan.ForeColor = &HFFFFFF
    Elseif silahkan.ForeColor = &HFFFFFF
    Then
        silahkan.ForeColor = &HC0&
    End If
End Sub

Private Sub txtFields_KeyPress(Index As Integer, KeyAscii As Integer)
    If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii =
    vbKeyBack Or KeyAscii = Asc(".")) Then
        Beep
        KeyAscii = 0
    End If
End Sub

Private Sub txtFields1_KeyPress(Index As Integer, KeyAscii As Integer)
    If (KeyAscii = vbKeyReturn) Then
        txtFields1(1).SetFocus
    End If
End Sub

Private Sub txtFields11_Change(Index As Integer)
    urutan.Caption = "Data : " &
    (Adodc1.Recordset.AbsolutePosition) &
    "/" & (Adodc1.Recordset.RecordCount)
End Sub

Private Sub txtFields11_KeyPress(Index As Integer, KeyAscii As Integer)
    If (KeyAscii = vbKeyReturn) Then
        txtFields(2).SetFocus
    End If
End Sub

```

FEEDMIX

```

Private Sub Form_Load()
Dim i As Integer
    banpilfeedmix
    berat.Text = MDIForm1.berat.Text
    For i = 0 To
        MDIForm1.Lstterpilih.ListCount - 1
            Bahan(i).Visible = True
            Komposisi(i).Visible = True
            Kg(i).Visible = True
        Next i
        pictFM.ScaleMode = vbPixels
        pictFM.AutoRedraw = True
        pictFM.AutoSize = True
        pictFM.BorderStyle = vbBSNone
        Me.BorderStyle = vbBSNone
        Set pictFM.Picture =
LoadPicture(App.Path &
"\gambar\feedmix.bmp")
        Me.Width = pictFM.Width
        Me.Height = pictFM.Height

        WindowRegion = MakeRegion(pictFM)
        SetWindowRgn Me.hWnd, WindowRegion,
True
End Sub

Private Sub berat_Change()
Dim WindowRegion As Long
Dim mkg0 As Single
Dim mkg1 As Single
Dim mkg2 As Single
Dim mkg3 As Single
Dim mkg4 As Single
Dim mkg5 As Single
Dim mkg6 As Single
Dim mkg7 As Single
Dim mkg8 As Single
Dim mkg9 As Single
Dim mkg10 As Single
Dim mkg11 As Single
Dim mkg12 As Single
Dim mkg13 As Single
Dim mkg14 As Single
    mkg0 = Val(Komposisi(0).Text) / 100
* Val(berat.Text)
    On Error GoTo salah
    Kg(0).Text = Format(mkg0,
"0.000")
    On Error GoTo 0
    mkg1 = Val(Komposisi(1).Text) / 100
* Val(berat.Text)
    On Error GoTo salah
    Kg(1).Text = Format(mkg1,
"0.000")
    On Error GoTo 0
    mkg2 = Val(Komposisi(2).Text) / 100
* Val(berat.Text)
    On Error GoTo salah
    Kg(2).Text = Format(mkg2,
"0.000")
    On Error GoTo 0
    mkg3 = Val(Komposisi(3).Text) / 100
* Val(berat.Text)
    On Error GoTo salah
    Kg(3).Text = Format(mkg3,
"0.000")
    On Error GoTo 0
    mkg4 = Val(Komposisi(4).Text) / 100
* Val(berat.Text)
    On Error GoTo salah
    Kg(4).Text = Format(mkg4,
"0.000")
    On Error GoTo 0
    mkg5 = Val(Komposisi(5).Text) / 100
* Val(berat.Text)
    On Error GoTo salah
    Kg(5).Text = Format(mkg5,
"0.000")
    On Error GoTo 0
    mkg6 = Val(Komposisi(6).Text) / 100
* Val(berat.Text)
    On Error GoTo salah
    Kg(6).Text = Format(mkg6,
"0.000")
    On Error GoTo 0
    mkg7 = Val(Komposisi(7).Text) / 100
* Val(berat.Text)
    On Error GoTo salah
    Kg(7).Text = Format(mkg7,
"0.000")
    On Error GoTo 0
    mkg8 = Val(Komposisi(8).Text) / 100
* Val(berat.Text)
    On Error GoTo salah
    Kg(8).Text = Format(mkg8,
"0.000")
    On Error GoTo 0
    mkg9 = Val(Komposisi(9).Text) / 100
* Val(berat.Text)
    On Error GoTo salah
    Kg(9).Text = Format(mkg9,
"0.000")
    On Error GoTo 0
    mkg10 = Val(Komposisi(10).Text) /
100 * Val(berat.Text)
    On Error GoTo salah
    Kg(10).Text = Format(mkg10,
"0.000")
    On Error GoTo 0
    mkg11 = Val(Komposisi(11).Text) /
100 * Val(berat.Text)
    On Error GoTo salah
    Kg(11).Text = Format(mkg11,
"0.000")
    On Error GoTo 0
    mkg12 = Val(Komposisi(12).Text) /
100 * Val(berat.Text)
    On Error GoTo salah
    Kg(12).Text = Format(mkg12,
"0.000")
    On Error GoTo 0
    mkg13 = Val(Komposisi(13).Text) /
100 * Val(berat.Text)
    On Error GoTo salah
    Kg(13).Text = Format(mkg13,
"0.000")
    On Error GoTo 0
    mkg14 = Val(Komposisi(14).Text) /
100 * Val(berat.Text)
    On Error GoTo salah
    Kg(14).Text = Format(mkg14,
"0.000")
    On Error GoTo 0
    Exit Sub
salah:
End Sub

Private Sub berat_KeyPress(KeyAscii As
Integer)
    If Not (KeyAscii >= Asc("0") And
KeyAscii <= Asc("9") Or KeyAscii =
vbKeyBack Or KeyAscii = Asc(".")) Then
        Beep
        KeyAscii = 0
    End If
End Sub

```

```

Private Sub keluar_Click()
    Unload Me
End Sub

Private Sub Kg_Change(Index As Integer)
    On Error GoTo salah
    Dim i As Integer
    Dim totkg As Double
    Dim harga As Double
    Dim totharga As Currency
    harga = 0
    totkg = Val(Kg(0).Text) +
    Val(Kg(1).Text) + Val(Kg(2).Text) +
    Val(Kg(3).Text) + Val(Kg(4).Text) +
    Val(Kg(5).Text) + Val(Kg(6).Text) +
    Val(Kg(7).Text) + Val(Kg(8).Text) +
    Val(Kg(9).Text) + Val(Kg(10).Text) +
    Val(Kg(11).Text) + Val(Kg(12).Text) +
    Val(Kg(13).Text) + Val(Kg(14).Text)
    totberat.Caption = Format(totkg,
    "0.000")
    For i = 0 To Komposisi.Count - 1
        If Komposisi(i) <> "" Then
            If Not
MDIForm1.Adodc1.Recordset.BOF Then
MDIForm1.Adodc1.Recordset.MoveFirst

MDIForm1.Adodc1.Recordset.Find
("nama_bahan=''" & Bahan(i) & "'')
            harga = harga +
Val(Kg(i).Text) *
Val(MDIForm1.Adodc1.Recordset.Fields("ha
rga"))
        End If
        Next i
        totharga = harga /
Val(totberat.Caption)
        Harga_ransum = Format(totharga,
        "0.00")
    salah:
End Sub

Private Sub komposisi_Change(Index As
Integer)
On Error GoTo 0
    Dim mtot As Single
    Dim mkg0 As Single
    Dim mkg1 As Single
    Dim mkg2 As Single
    Dim mkg3 As Single
    Dim mkg4 As Single
    Dim mkg5 As Single
    Dim mkg6 As Single
    Dim mkg7 As Single
    Dim mkg8 As Single
    Dim mkg9 As Single
    Dim mkg10 As Single
    Dim mkg11 As Single
    Dim mkg12 As Single
    Dim mkg13 As Single
    Dim mkg14 As Single
    Dim mselisih As Single

    mtot = Val(Komposisi(0).Text) +
Val(Komposisi(1).Text) +
Val(Komposisi(2).Text) +
Val(Komposisi(3).Text) +
Val(Komposisi(4).Text) +
Val(Komposisi(5).Text) +
Val(Komposisi(6).Text) +
Val(Komposisi(7).Text) +
Val(Komposisi(8).Text) +
Val(Komposisi(9).Text) +
Val(Komposisi(10).Text) +

```

Val(Komposisi(11).Text) +
Val(Komposisi(12).Text) +
Val(Komposisi(13).Text) +
Val(Komposisi(14).Text)

On Error GoTo salah

Totalkomposisi.Text =
Format(mtot, "0.000")

On Error GoTo 0

mkg0 = Val(Komposisi(0).Text) / 100

* Val(berat.Text)

On Error GoTo salah

Kg(0).Text = Format(mkg0,
"0.000")

On Error GoTo 0

mkg1 = Val(Komposisi(1).Text) / 100

* Val(berat.Text)

On Error GoTo salah

Kg(1).Text = Format(mkg1,
"0.000")

On Error GoTo 0

mkg2 = Val(Komposisi(2).Text) / 100

* Val(berat.Text)

On Error GoTo salah

Kg(2).Text = Format(mkg2,
"0.000")

On Error GoTo 0

mkg3 = Val(Komposisi(3).Text) / 100

* Val(berat.Text)

On Error GoTo salah

Kg(3).Text = Format(mkg3,
"0.000")

On Error GoTo 0

mkg4 = Val(Komposisi(4).Text) / 100

* Val(berat.Text)

On Error GoTo salah

Kg(4).Text = Format(mkg4,
"0.000")

On Error GoTo 0

mkg5 = Val(Komposisi(5).Text) / 100

* Val(berat.Text)

On Error GoTo salah

Kg(5).Text = Format(mkg5,
"0.000")

On Error GoTo 0

mkg6 = Val(Komposisi(6).Text) / 100

* Val(berat.Text)

On Error GoTo salah

Kg(6).Text = Format(mkg6,
"0.000")

On Error GoTo 0

mkg7 = Val(Komposisi(7).Text) / 100

* Val(berat.Text)

On Error GoTo salah

Kg(7).Text = Format(mkg7,
"0.000")

On Error GoTo 0

mkg8 = Val(Komposisi(8).Text) / 100

* Val(berat.Text)

On Error GoTo salah

Kg(8).Text = Format(mkg8,
"0.000")

On Error GoTo 0

mkg9 = Val(Komposisi(9).Text) / 100

* Val(berat.Text)

On Error GoTo salah

Kg(9).Text = Format(mkg9,
"0.000")

On Error GoTo 0

mkg10 = Val(Komposisi(10).Text) /
100 * Val(berat.Text)

On Error GoTo salah

Kg(10).Text = Format(mkg10,
"0.000")

On Error GoTo 0

```

mkg11 = Val(Komposisi(11).Text) /
100 * Val(berat.Text)
    On Error GoTo salah
        Kg(11).Text = Format(mkg11,
"0.000")
    On Error GoTo 0
mkg12 = Val(Komposisi(12).Text) /
100 * Val(berat.Text)
    On Error GoTo salah
        Kg(12).Text = Format(mkg12,
"0.000")
    On Error GoTo 0
mkg13 = Val(Komposisi(13).Text) /
100 * Val(berat.Text)
    On Error GoTo salah
        Kg(13).Text = Format(mkg13,
"0.000")
    On Error GoTo 0
mkg14 = Val(Komposisi(14).Text) /
100 * Val(berat.Text)
    On Error GoTo salah
        Kg(14).Text = Format(mkg14,
"0.000")
    On Error GoTo 0
mselisih = Val(Totalkomposisi.Text) -
100)
    On Error GoTo salah
        selisih.Text = Format(mselisih,
"0.000")
    On Error GoTo 0
'menghitung secara trial and error
method
    updateNut
    Exit Sub
salah:
End Sub

Private Sub Label35_MouseMove(Button As
Integer, Shift As Integer, x As Single,
Y As Single)
    Label35.BackColor = &H8000&
End Sub

Private Sub Label36_Click()
    keluar_Click
End Sub

Private Sub Label36_MouseMove(Button As
Integer, Shift As Integer, x As Single,
Y As Single)
    Label36.BackColor = &H8000&
End Sub

Private Sub updateNut()
    Dim i As Integer
    Dim eM As Single
    Dim sum0 As Double
    Dim sum1 As Double
    Dim sum2 As Double
    Dim sum3 As Double
    Dim sum4 As Double
    Dim sum5 As Double
    Dim sum6 As Double
    Dim sum7 As Double
    Dim sum8 As Double
    Dim sum9 As Double
    Dim sum10 As Double
    Dim sum11 As Double
    Dim sum12 As Double
    Dim sum13 As Double
    Dim sum14 As Double
    Dim sum15 As Double
    sum0 = 0
    sum1 = 0
    sum2 = 0
    sum3 = 0
    sum4 = 0
    sum5 = 0
    sum6 = 0
    sum7 = 0
    sum8 = 0
    sum9 = 0
    sum10 = 0
    sum11 = 0
    sum12 = 0
    sum13 = 0
    sum14 = 0
    sum15 = 0
    harga = 0
    For i = 0 To Komposisi.Count - 1
        If Komposisi(i) <> "" Then
            If Not
MDIForm1.Adodc1.Recordset.BOF Then
MDIForm1.Adodc1.Recordset.MoveFirst

MDIForm1.Adodc1.Recordset.Find
("nama_bahan=" & Bahan(i) & ""))
        sum0 = sum0 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("en
ergi_metabolis"))
        sum1 = sum1 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("pr
otein_kasar"))
        sum2 = sum2 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("le
mak_kasar"))
        sum3 = sum3 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("se
rat_kasar"))
        sum4 = sum4 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("ka
lsium"))
        sum5 = sum5 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("ph
ospor_tersedia"))
        sum6 = sum6 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("na
trium"))
        sum7 = sum7 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("ch
lor"))
        sum8 = sum8 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("ma
gnesium"))
        sum9 = sum9 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("za
t_besi"))
        sum10 = sum10 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("vi
tamin_b12"))
        sum11 = sum11 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("vi
tamin_e"))
        sum12 = sum12 +
(Val(Komposisi(i)) / 100) *

```

```

Val(MDIForm1.Adodc1.Recordset.Fields("mention"))
    sum13 = sum13 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("lisin"))
    sum14 = sum14 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("sistin"))
    sum15 = sum15 +
(Val(Komposisi(i)) / 100) *
Val(MDIForm1.Adodc1.Recordset.Fields("triptophan"))
End If
Next i
Nut(0) = sum0
Nut(1) = sum1
Nut(2) = sum2
Nut(3) = sum3
Nut(4) = sum4
Nut(5) = sum5
Nut(6) = sum6
Nut(7) = sum7
Nut(8) = sum8
Nut(9) = sum9
Nut(10) = sum10
Nut(11) = sum11
Nut(12) = sum12
Nut(13) = sum13
Nut(14) = sum14
Nut(15) = sum15
End Sub
Private Sub editbahan_Click()
    Unload Me
    frmedit.Show
End Sub

Private Sub editnutrisi_Click()
    Unload Me
    frmbatnutupdate.Show
End Sub

Private Sub ingredient_Click()
    Unload Me
    frmbatbah.Show
End Sub

Private Sub Komposisi_KeyPress(Index As Integer, KeyAscii As Integer)
    If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii = vbKeyBack Or KeyAscii = Asc(".")) Then
        Beep
        KeyAscii = 0
    End If
End Sub

Private Sub Nutrient_Click()
    Unload Me
    frmbatnut.Show
End Sub

Private Sub pilbah_Click()
    Unload Me
End Sub

Private Sub selisih_Change()
    If Val(Totalkomposisi.Text) > 100
Then
    Beep
    MsgBox "Total komposisi maksimal 100 !", vbCritical, "EzFeed V 1.0 (Max 100)"
End If
Label39.FontBold = True
Label39.ForeColor = &HFF&
Else
Label39.FontBold = False
Label39.ForeColor = &H404040
End If
End Sub

Private Sub tblcetak_Click()
    Dim i As Integer
    Dim jumpil As String
    Dim j As Integer
    Dim k As Integer
    cdlg.CancelError = True
    jumpil =
MDIForm1.Lstterpilih.ListCount - 1
    cdlg.Flags = PD_ALLPAGES
    On Error GoTo salah
    cdlg.Action = 5
    For i = 1 To cdlg.Copies
        Printer.FontItalic = False
        Printer.FontBold = False
        Printer.FontSize = 10
        Printer.CurrentX = 7000
        Printer.CurrentY = 500
        Printer.Print Format(Now, "DDDD, DD MMMM YYYY")
        Printer.CurrentX = 1000
        Printer.CurrentY = 500
        Printer.FontBold = True
        Printer.Print "FORMULA RANSUM TERNAK UNGGAS"
        Printer.CurrentX = 1000
        Printer.CurrentY = 800
        Printer.FontBold = True
        Printer.Print "EzFeed Ver 1.0.2003 (C) 2003"
        Printer.CurrentX = 5000
        Printer.CurrentY = 1100
        Printer.FontBold = True
        Printer.FontSize = 14
        Printer.Print "FeedMix"
        Printer.CurrentX = 7000
        Printer.CurrentY = 800
        Printer.FontBold = False
        Printer.FontSize = 10
        Printer.Print Format(Now, "H:mm:ss")
        Printer.CurrentX = 7000
        Printer.CurrentY = 1100
        Printer.FontBold = False
        Printer.Print
MDIForm1.StatusBar1.Panels(4).Text
        Printer.CurrentX = 1000
        Printer.CurrentY = 1400
        Printer.FontSize = 12
        Printer.Print
MDIForm1.cmbounggas.Text
        Printer.Line (1000, 1800)-
(10500, 1800)
        Printer.Line (1000, 1850)-(7000,
1850)
        Printer.CurrentX = 1000
        Printer.CurrentY = 1900
        Printer.FontSize = 10
        Printer.FontBold = True
End If

```

```

Printer.Print "Bahan Pakan"
Printer.CurrentX = 3930
Printer.CurrentY = 1900
Printer.Print "% dlm ransum"

Printer.CurrentX = 5500
Printer.CurrentY = 1900
Printer.Print "kg dlm ransum"

Printer.Line (1000, 2250)-(7000, 2250)

For j = 0 To jumpil
Printer.FontBold = False
Printer.FontSize = 10
Printer.CurrentX = 1000
Printer.CurrentY = 2400 + (300 * j)
Printer.Print Bahan(j).Text

Printer.FontBold = False
Printer.FontSize = 10
Printer.CurrentX = 4430
Printer.CurrentY = 2400 + (300 * j)
Printer.Print Komposisi(j).Text

Printer.FontBold = False
Printer.FontSize = 10
Printer.CurrentX = 6030
Printer.CurrentY = 2400 + (300 * j)
Printer.Print Kg(j).Text
Next j

Printer.FontBold = False
Printer.Line (1000, (2800 + (300 *
jumpil)))-(7000, (2800 + (300*jumpil)))

Printer.CurrentX = 1000
Printer.CurrentY =(3000+(300* jumpil))
Printer.FontBold = True
Printer.FontSize = 12
Printer.Print "Total"

Printer.FontBold = False
Printer.FontSize = 10
Printer.CurrentX = 1000
Printer.CurrentY = 3300+(300 * jumpil)
Printer.Print "Komposisi (%)"

Printer.CurrentX = 4000
Printer.CurrentY = 3300 + (300 * jumpil)
Printer.Print Totalkomposisi.Text

Printer.CurrentX = 1000
Printer.CurrentY = 3600 + (300 * jumpil)
Printer.Print "Berat (kg)"

Printer.CurrentX = 4000
Printer.CurrentY = 3600 + (300 * jumpil)
Printer.Print berat.Text

Printer.CurrentX = 1000
Printer.CurrentY = 3900 + (300 * jumpil)
Printer.Print "Harga (Rp/kg)"

Printer.CurrentX = 4000
Printer.CurrentY = 3900 + (300 * jumpil)
Printer.Print Harga_ransum.Text

Printer.CurrentX = 1000
Printer.CurrentY = 5000 + (300 * jumpil)
Printer.FontBold = True
Printer.Print "Poultry Feed Formulation
System Version 1.0.2003"

Printer.CurrentX = 1000
Printer.CurrentY = 5250 + (300 * jumpil)
Printer.Print "(C) 2003 by Budi
Aprilianto Kusnandar"

Printer.CurrentX = 1000
Printer.CurrentY = 5500 + (300 * jumpil)
Printer.Print "Dep. INMT, IPB"

Printer.Line (7100, 1800)-(7100, 7400)

Printer.FontBold = True
Printer.CurrentX = 7200
Printer.CurrentY = 2000
Printer.Print "HASIL PERHITUNGAN"

For k = 0 To 15
Printer.Currentx = 7200
Printer.Currenty = 2300 + (300 * k)
Printer.FontBold = False
Printer.Print Label7(k).Caption

Printer.CurrentX = 9250
Printer.CurrentY = 2300 + (300 * k)
Printer.Print Nut(k).Text

Printer.CurrentX = 10300
Printer.CurrentY = 2300 + (300 * k)
Printer.Print Label8(k).Caption
Next k
Printer.EndDoc
Next i
salah:
End Sub

Private Sub Simpan_Click()
On Error GoTo salah
On Error GoTo err
cdlg.DialogTitle = "Simpan Formulasi
(FeedMix)"
cdlg.FileName = "EzFeed.Aan"
cdlg.Filter = "File EzFeed
(*.Aan)|*.Aan"
If cdlg.FileName <> "" Then
cdlg.ShowSave
Open cdlg.FileName For Output As #1,
For i = 0 To 15
Write #1, Nut(i).Text
Next i
Write #1,
MDIForm1.Lstterpilih.ListCount
Write #1, Label1(i).Caption
Write #1, berat.Text
Write #1, Harga_ransum.Text
Write #1,
MDIForm1.StatusBar1.Panels(4).Text
Write #1, Format(Now, "dddd, dd mmmm
yyyy")
Write #1, Format(Now, "H:mm")
Write #1, MDIForm1.cmbounggas.Text
For i = 0 To 14
Write #1, Bahan(i).Text
Write #1, Komposisi(i).Text
Write #1, Kg(i).Text
Next i
err:
Close #1
End If
salah:
End Sub

```

LEASTCOST

```
Option Explicit
Dim posPivotX As Long
Dim posPivotY As Long
Dim elpivot As Double
Dim elpivot2 As Double
Dim temp(3400) As Double
Dim newcoll As Integer
Dim header As Integer
Dim rows As Integer
Dim s As ListItem
Dim T As ListItem
Dim jumlahs As Integer
Dim jumlaha As Integer
Dim Nilai_harga As Integer
Dim nosolution As Integer
Dim nohasil As Integer
Dim cn As New ADODB.Connection
Dim slack As String
Dim artifisial As String

Private Sub Form_Deactivate()
    Tutup
End Sub

Private Sub Form_Load()
On Error GoTo err
    MousePointer = 11
    Data1.DatabaseName = App.Path &
    "\Feed.mdb"
    Data1.RecordSource = "Bahan"
    masuk_hitung
    MousePointer = 0
    updateNut
    periksa
    Data1.Refresh
err:
End Sub

Private Sub masuk_hitung()
Dim i As Integer
    berat.Text = MDIForm1.berat.Text
    txtunggas.Text =
    MDIForm1.cboUnggas.Text
    On Error GoTo salah
        cn.ConnectionString =
        "Provider=Microsoft.Jet.OLEDB.4.0;Data
        Source=" & App.Path & "\Feed.mdb;Persist
        Security Info=False"
        cn.Open

        fillCboUnggas
        fillLstBahan

        Pilih

        banpilleastcost
        txtunggas.Text =
        MDIForm1.cboUnggas.Text

MDIForm1.Adodc2.Recordset.MoveFirst

MDIForm1.Adodc2.Recordset.Find
("unggas=''' & txtunggas & '''")
nutmin(0) =
MDIForm1.Adodc2.Recordset.Fields("EM")
nutmin(1) =
MDIForm1.Adodc2.Recordset.Fields("PK")
nutmin(2) =
MDIForm1.Adodc2.Recordset.Fields("LK")
nutmin(3) =
MDIForm1.Adodc2.Recordset.Fields("SK")

nutmin(4) =
MDIForm1.Adodc2.Recordset.Fields("Ca")
nutmin(5) =
MDIForm1.Adodc2.Recordset.Fields("P_ters
edia")
nutmin(6) =
MDIForm1.Adodc2.Recordset.Fields("met")
nutmin(7) =
MDIForm1.Adodc2.Recordset.Fields("lys")
nutmax(0) =
MDIForm1.Adodc2.Recordset.Fields("enumax"
)
nutmax(1) =
MDIForm1.Adodc2.Recordset.Fields("pkmax"
)
nutmax(2) =
MDIForm1.Adodc2.Recordset.Fields("lkmax"
)
nutmax(3) =
MDIForm1.Adodc2.Recordset.Fields("skmax"
)
nutmax(4) = MDIForm1
    .Adodc2.Recordset.Fields("camax")
nutmax(5) =
MDIForm1.Adodc2.Recordset.Fields("p_max"
)
nutmax(6) =
MDIForm1.Adodc2.Recordset.Fields("metmax
")
nutmax(7) =
MDIForm1.Adodc2.Recordset.Fields("lysmax
")
    For i = 0 To
    1stSelectedBahan.ListCount - 1
        lblmin(i).Visible = True
        lblmax(i).Visible = True
        Bahan(i).Visible = True
        Komposisi(i).Visible = True
        Kg(i).Visible = True
        If Bahan(i).Text <> "" Then
            MDIForm1.Adodc1.Recordset.MoveFirst
            MDIForm1.Adodc1.Recordset.Find
            ("nama_bahan=''' & Bahan(i) & '''")
            lblmin(i).Caption =
            MDIForm1.Adodc1.Recordset.Fields("bm_min
            ")
            lblmax(i).Caption =
            MDIForm1.Adodc1.Recordset.Fields("bm_max
            ")
        End If
    Next i
salah:
End Sub

Function fillCboUnggas()
    cboUnggas.Text =
    MDIForm1.cboUnggas.Text
End Function

Function fillLstBahan()
Dim i As Integer
    For i = 0 To
    MDIForm1.Lstterpilih.ListCount - 1
        1stSelectedBahan.AddItem
        MDIForm1.Lstterpilih.List(i)
    Next i
End Function
```

```

Private Sub berat_KeyPress(KeyAscii As
Integer)
    If Not (KeyAscii >= Asc("0") And
KeyAscii <= Asc("9") Or KeyAscii =
vbKeyBack Or KeyAscii = Asc(".")) Then
        Beep
        KeyAscii = 0
    End If
End Sub

Private Sub editbahan_Click()
    Tutup
    frmedit.Show
End Sub

Private Sub editnutrisi_Click()
    Tutup
    frmbarutupdate.Show
End Sub

Private Sub ingredient_Click()
    Tutup
    frmbarbah.Show
End Sub

Private Sub keluar_Click()
    Tutup
End Sub

Private Sub komposisi_Change(Index As
Integer)
    Dim i As Integer
    Totalkomposisi.Text =
    Val(Komposisi(0).Text) +
    Val(Komposisi(1).Text) +
    Val(Komposisi(2).Text) +
    Val(Komposisi(3).Text) +
    Val(Komposisi(4).Text) +
    Val(Komposisi(5).Text) +
    Val(Komposisi(6).Text) +
    Val(Komposisi(7).Text) +
    Val(Komposisi(8).Text) +
    Val(Komposisi(9).Text) +
    Val(Komposisi(10).Text) +
    Val(Komposisi(11).Text) +
    Val(Komposisi(12).Text) +
    Val(Komposisi(13).Text) +
    Val(Komposisi(14).Text)
    Dim eM As Single
    Dim sum0 As Double
    Dim sum1 As Double
    Dim sum2 As Double
    Dim sum3 As Double
    Dim sum4 As Double
    Dim sum5 As Double
    Dim sum6 As Double
    Dim sum7 As Double
    Dim sum8 As Double
    Dim sum9 As Double
    Dim sum10 As Double
    Dim sum11 As Double
    Dim sum12 As Double
    Dim sum13 As Double
    Dim sum14 As Double
    Dim sum15 As Double
    Dim harga As Double
    sum0 = 0
    sum1 = 0
    sum2 = 0
    sum3 = 0
    sum4 = 0
    sum5 = 0
    sum6 = 0
    sum7 = 0
    sum8 = 0
    sum9 = 0
    sum10 = 0
    sum11 = 0
    sum12 = 0
    sum13 = 0
    sum14 = 0
    sum15 = 0
    harga = 0
    For i = 0 To
        lstSelectedBahan.ListCount - 1
        Dim hit As String
        Data1.RecordSource = "Bahan"
        Data1.Refresh
        hit = "select* from bahan where
        nama_bahan= '" & Bahan(i).Text & "'"
        Data1.RecordSource = hit
        Data1.Refresh
        sum0 = sum0 +
        (Val(Komposisi(i).Text) / 100) *
        Val(Data1.Recordset.Fields("energi_metab
        olis").Value)
        sum1 = sum1 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("protein_kasa
        r").Value)
        sum2 = sum2 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("lemak_kasar"
        ).Value)
        sum3 = sum3 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("serat_kasar"
        ).Value)
        sum4 = sum4 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("kalsium").Va
        lue)
        sum5 = sum5 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("phospor_ters
        edia").Value)
        sum6 = sum6 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("metionin").V
        alue)
        sum7 = sum7 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("lisin").Valu
        e)
        sum8 = sum8 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("natrium").Va
        lue)
        sum9 = sum9 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("chlor").Valu
        e)
        sum10 = sum10 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("magnesium").
        Value)
        sum11 = sum11 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("zat_besi").V
        alue)
        sum12 = sum12 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("vitamin_b12"
        ).Value)
        sum13 = sum13 +
        (Val(Komposisi(i)) / 100) *
        Val(Data1.Recordset.Fields("vitamin_e").
        Value)

```

```

        sum14 = sum14 +
(Val(Komposisi(i)) / 100) *
Val(Data1.Recordset.Fields("sistin").Value)
        sum15 = sum15 +
(Val(Komposisi(i)) / 100) *
Val(Data1.Recordset.Fields("triptophan") .Value)
        harga = harga +
Val(Kg(i).Text) *
Val(Data1.Recordset.Fields("harga").Value)
        Nut(0) = Round(sum0, 3)
        Nut(1) = Round(sum1, 3)
        Nut(2) = Round(sum2, 3)
        Nut(3) = Round(sum3, 3)
        Nut(4) = Round(sum4, 3)
        Nut(5) = Round(sum5, 3)
        Nut(6) = Round(sum6, 3)
        Nut(7) = Round(sum7, 3)
        Nut(8) = Round(sum8, 3)
        Nut(9) = Round(sum9, 3)
        Nut(10) = Round(sum10, 3)
        Nut(11) = Round(sum11, 3)
        Nut(12) = Round(sum12, 3)
        Nut(13) = Round(sum13, 3)
        Nut(14) = Round(sum14, 3)
        Nut(15) = Round(sum15, 3)
        Harga_ransum = Round(harga /
Val(berat.Text), 3)
        Dim k As Integer
        For k = 0 To Kg.Count - 1
            Kg(k).Text = Val(Komposisi(k).Text)
            * berat.Text / 100
        Next
        totalberat.Text = Val(Kg(0).Text) +
Val(Kg(1).Text) + Val(Kg(2).Text) +
Val(Kg(3).Text) + Val(Kg(4).Text) +
Val(Kg(5).Text) + Val(Kg(6).Text) +
Val(Kg(7).Text) + Val(Kg(8).Text) +
Val(Kg(9).Text) + Val(Kg(10).Text) +
Val(Kg(11).Text) + Val(Kg(12).Text)
    End Sub

    Private Sub Nutrient_Click()
        Tutup
        frmbatnut.Show
    End Sub

    Private Sub pilbah_Click()
        Tutup
    End Sub

    Private Sub Simpan_Click()
        On Error GoTo salah
        On Error GoTo err
        cdlg.DialogTitle = "Simpan Formulasasi (LeastCost)"
        cdlg.FileName = "EzFeed.Aan"
        cdlg.Filter = "File EzFeed (*.Aan)|*.Aan"
        If cdlg.FileName <> "" Then
            cdlg.ShowSave
            Open cdlg.FileName For Output As #1
            Dim i As Integer
            For i = 0 To 15
                Write #1, Nut(i).Text
            Next i
            Write #1,
        MDIForm1.Lstterpilih.ListCount
        Write #1, Label1(1).Caption
        Write #1, berat.Text
        Write #1, Harga_ransum.Text
        Write #1,
        MDIForm1.StatusBar1.Panels(4).Text
        Write #1, Format(Now, "dddd, dd mmmm yyyy")
        Write #1, Format(Now, "H:mm")
        Write #1, MDIForm1.cmbounggas.Text
        For i = 0 To 14
            Write #1, Bahan(i).Text
            Write #1, Komposisi(i).Text
            Write #1, Kg(i).Text
        Next i
        err:
        Close #1
    End If
    salah:
    End Sub

    Private Sub tblicetak_Click()
        On Error GoTo salah
        Dim i As Integer
        Dim jumpil As String
        Dim j As Integer
        Dim k As Integer
        Dim l As Integer
        jumpil =
        MDIForm1.Lstterpilih.ListCount - 1
        cdlg.Action = 5
        For i = 1 To cdlg.Copies
        Printer.FontItalic = False
        Printer.FontBold = False
        Printer.FontSize = 10
        Printer.CurrentX = 7000
        Printer.CurrentY = 500
        Printer.Print Format(Now, "DDDD, DD MMMM YYYY")

        Printer.CurrentX = 1000
        Printer.CurrentY = 500
        Printer.FontBold = True
        Printer.Print "FORMULA RANSUM TERNAK UNGGAS"

        Printer.CurrentX = 1000
        Printer.CurrentY = 800
        Printer.FontBold = True
        Printer.Print "EzFeed Ver 1.0.2003 (C) 2003"

        Printer.CurrentX = 5000
        Printer.CurrentY = 1100
        Printer.FontBold = True
        Printer.FontSize = 14
        Printer.Print "Least Cost"

        Printer.CurrentX = 7000
        Printer.CurrentY = 800
        Printer.FontBold = False
        Printer.FontSize = 10
        Printer.Print Format(Now, "H:mm:ss")

        Printer.CurrentX = 7000
        Printer.CurrentY = 1100
        Printer.FontBold = False
        Printer.Print
        MDIForm1.StatusBar1.Panels(4).Text

        Printer.CurrentX = 1000
        Printer.CurrentY = 1400
        Printer.FontSize = 10
        Printer.Print MDIForm1.cmbounggas.Text
        Printer.Line (1000, 1800)-(10500, 1800)
    End Sub

```

```

Printer.Line (1000, 1850)-(10500, 1850)

Printer.CurrentX = 1000
Printer.CurrentY = 1900
Printer.FontSize = 10
Printer.FontBold = True
Printer.Print "Bahan Pakan"

Printer.CurrentX = 3500
Printer.CurrentY = 1900
Printer.FontBold = False
Printer.Print "C.B. Min (%)"

Printer.CurrentX = 5000
Printer.CurrentY = 1900
Printer.FontBold = True
Printer.Print "% dlm ransum"

Printer.FontBold = False
Printer.CurrentX = 6500
Printer.CurrentY = 1900
Printer.Print "C.B. Max (%)"

Printer.FontBold = True
Printer.CurrentX = 8500
Printer.CurrentY = 1900
Printer.Print "kg dlm ransum"

Printer.Line (1000, 2250)-(10500, 2250)
Next i

For j = 0 To jumpil
Printer.FontBold = False
Printer.FontSize = 10
Printer.CurrentX = 1000
Printer.CurrentY = 2400 + (300 * j)
Printer.Print Bahan(j).Text

Printer.FontBold = False
Printer.FontSize = 10
Printer.CurrentX = 3750
Printer.CurrentY = 2400 + (300 * j)
Printer.Print lblmin(j).Caption

Printer.FontBold = True
Printer.FontSize = 10
Printer.CurrentX = 5250
Printer.CurrentY = 2400 + (300 * j)
Printer.Print Komposisi(j).Text

Printer.FontBold = False
Printer.FontSize = 10
Printer.CurrentX = 6750
Printer.CurrentY = 2400 + (300 * j)
Printer.Print lblmax(j).Caption

Printer.FontBold = True
Printer.FontSize = 10
Printer.CurrentX = 8700
Printer.CurrentY = 2400 + (300 * j)
Printer.Print Kg(j).Text
Next j

Printer.FontBold = False
Printer.Line (1000, (2800 + (300 *
jumpil)))-(10500, (2800 + (300 *
jumpil)))

Printer.CurrentX = 1000
Printer.CurrentY = (3000 + (300*jumpil))
Printer.FontBold = True
Printer.FontSize = 12
Printer.Print "Total"

Printer.FontBold = False
Printer.FontSize = 10
Printer.CurrentX = 1000
Printer.CurrentY = 3300 + (300 * jumpil)
Printer.Print "Komposisi (%)" 

Printer.CurrentX = 4000
Printer.CurrentY = 3300 + (300 * jumpil)
Printer.Print Totalkomposisi.Text

Printer.CurrentX = 1000
Printer.CurrentY = 3600 + (300 * jumpil)
Printer.Print "Berat (kg)"

Printer.CurrentX = 4000
Printer.CurrentY = 3600 + (300 * jumpil)
Printer.Print totalberat.Text

Printer.CurrentX = 1000
Printer.CurrentY = 3900 + (300 * jumpil)
Printer.Print "Harga (Rp/kg)"

Printer.CurrentX = 4000
Printer.CurrentY = 3900 + (300 * jumpil)
Printer.Print Harga_ransum.Text

Printer.Line (1000, (4300 + (300 *
jumpil)))-(10500, (4300 + (300 *
jumpil)))
Printer.Line (1000, 8500)-(10500, 8500)
Printer.Line (1000, 8850)-(10500, 8850)

For k = 0 To 15
Printer.CurrentX = 1000
Printer.CurrentY = 8550
Printer.FontBold = True
Printer.Print "NUTRISI"

Printer.CurrentX = 1000
Printer.CurrentY = 9000 + (300 * k)
Printer.FontBold = False
Printer.Print Label1(k).Caption

Printer.CurrentX = 3000
Printer.CurrentY = 8550
Printer.Print "C.N. Min"

Printer.CurrentX = 4500
Printer.CurrentY = 8550
Printer.FontBold = True
Printer.Print "Kand. Nutrisi"

Printer.CurrentX = 6300
Printer.CurrentY = 8550
Printer.FontBold = False
Printer.Print "C.N. Max"

Printer.CurrentX = 4700
Printer.CurrentY = 9000 + (300 * k)
Printer.FontBold = True
Printer.Print Nut(k).Text

Printer.CurrentX = 7500
Printer.CurrentY = 9000 + (300 * k)
Printer.FontBold = False
Printer.Print Label8(k).Caption
Next k

For l = 0 To 7
Printer.CurrentX = 3200
Printer.CurrentY = 9000 + (300 * l)
Printer.Print nutmin(l).Caption

Printer.CurrentX = 6200

```



```

        For j = 1 To
    lvBahan.ColumnHeaders.Count
        1.ListSubItems.Add , , 0
    Next
Next

End If
Next n
For i = 1 To
lstSelectedBahan.ListCount
    harga.AddItem
    (lvBahan.ListItems(2 +
    i).ListSubItems(1).Text)
    lvBahan.ListItems(2 +
    i).ListSubItems(1).Text = 0
Next
Call transpose
Call fase2
End Sub

Private Sub transpose()
Dim Total As ListItem
Dim k As Integer
If lvBahan.ListItems.Count < 1 Then Exit Sub

lvHitung.ColumnHeaders.Clear
lvHitung.ListItems.Clear
ransum.ColumnHeaders.Clear
lvHitung.ListItems.Clear
transposeLV lvBahan, lvHitung

'Supaya 100 %
Set Total = lvHitung.ListItems.Add(, ,
"Total 100")
Total.ListSubItems.Add , , "-1"
For k = 3 To lstSelectedBahan.ListCount
+ 3
    Total.ListSubItems.Add , , "1"
Next
For k = lstSelectedBahan.ListCount + 3
To lvHitung.ColumnHeaders.Count - 1
    Total.ListSubItems.Add , , "0"
Next

lvHitung.ListItems.Add , , "Zj"
lvHitung.ListItems.Add , , "Zj-Cj"
lvHitung.ListItems(lvHitung.ListItems.Count - 1).ListSubItems.Add , , "0"
lvHitung.ListItems(lvHitung.ListItems.Count).ListSubItems.Add , , "0"

Dim i As Long
Dim j As Long
Dim Zj As Double
Dim ZjMinCj As Double

For i = 2 To
    lvHitung.ColumnHeaders.Count - 1
        Zj = 0
        For j = 1 To
            lvHitung.ListItems.Count - 2
                Zj = Zj +
                    Round(CDbl(lvHitung.ListItems(j).ListSubItems(i).Text), 4) *
                    Round(CDbl(lvHitung.ListItems(j).ListSubItems(1).Text), 4)
        Next

    lvHitung.ListItems(lvHitung.ListItems.Count - 1).ListSubItems.Add , , Zj
Next

For i = 2 To
    lvHitung.ColumnHeaders.Count - 1
        ZjMinCj =
            Round(CDbl(lvHitung.ListItems(lvHitung.ListItems.Count - 1).ListSubItems(i).Text), 4) -
            Round(CDbl(lvHitung.ListItems(1).ListSubItems(i).Text), 4)

    lvHitung.ListItems(lvHitung.ListItems.Count).ListSubItems.Add , , ZjMinCj
Next

Dim tmpZjMinCj As Double
Dim maxZjMinCj As Double

tmpZjMinCj = 0
posPivotX = 3
For i = 3 To
    lvHitung.ColumnHeaders.Count - 15
        maxZjMinCj =
            Round(CDbl(lvHitung.ListItems(lvHitung.ListItems.Count).ListSubItems(i)), 4)

        If maxZjMinCj < tmpZjMinCj Then
            posPivotX = i
            tmpZjMinCj = maxZjMinCj
        End If
Next

posPivotX = posPivotX
For i = 1 To lvHitung.ListItems.Count
    lvHitung.ListItems(i).ListSubItems(posPivotX).ForeColor = vbRed 'kolom terpilih
    lvHitung.ListItems(i).ListSubItems(posPivotX).Bold = True
Next
lvHitung.Refresh

'//cari rasio kolom terpilih
Dim rasio As Double
lvHitung.ColumnHeaders.Add , , "Rasio Terpilih"
For i = 1 To lvHitung.ListItems.Count - 2
    rasio = 0
    If
        Round(CDbl(lvHitung.ListItems(i).ListSubItems(posPivotX).Text), 4) <> 0 Then
            rasio =
                Round(CDbl(lvHitung.ListItems(i).ListSubItems(2).Text), 4) /
                Round(CDbl(lvHitung.ListItems(i).ListSubItems(posPivotX).Text), 4)

        lvHitung.ListItems(i).ListSubItems.Add ,
            Round(CDbl(rasio), 4)
        Else
            lvHitung.ListItems(i).ListSubItems.Add ,
            "DIV/0!"
        End If
    Next

    lvHitung.ListItems(lvHitung.ListItems.Count - 2).ListSubItems.Add , , ""

```

```

lvHitung.ListItems(lvHitung.ListItems.Count - 1).ListSubItems.Add , , ""
lvHitung.ListItems(lvHitung.ListItems.Count).ListSubItems.Add , , ""

'//cari rasio terkecil
Dim tmpRasio As Double
Dim minRasio As Double

tmpRasio = 0

posPivotY = 2
tmpRasio =
(lvHitung.ListItems(2).ListSubItems(lvHitung.ColumnHeaders.Count - 1))

For i = 2 To lvHitung.ListItems.Count - 2
    If IsNumeric(lvHitung.ListItems(i).ListSubItems(lvHitung.ColumnHeaders.Count - 1)) Then
        minRasio =
Round(CDbl(lvHitung.ListItems(i).ListSubItems(lvHitung.ColumnHeaders.Count - 1)), 2)
        If Round(CDbl(minRasio), 2) <= Round(CDbl(tmpRasio), 2) Then
            If minRasio > 0 Then
                posPivotY = i
                tmpRasio = minRasio
            End If
        End If
    End If
Next

For i = 1 To
lvHitung.ListItems(posPivotY).ListSubItems.Count
lvHitung.ListItems(posPivotY).ListSubItems(i).ForeColor = vbRed 'baris terpilih
lvHitung.ListItems(posPivotY).ListSubItems(i).Bold = True
Next
lvHitung.ListItems(posPivotY).ListSubItems(posPivotX).ForeColor = vbBlue 'element pivot
elpivot =
CDbl(lvHitung.ListItems(posPivotY).Lists.SubItems(posPivotX).Text)
Call faktor_pengali
lvHitung.Refresh
ransum.Refresh
salah:
End Sub

Sub transposeLV(lvSrc As ListView,
lvdest As ListView)
    Dim i As Long
    Dim j As Long

    lvdest.ColumnHeaders.Clear
    lvdest.ListItems.Clear
    lvdest.ColumnHeaders.Add , , ""

    For i = 1 To lvSrc.ListItems.Count
        lvdest.ColumnHeaders.Add , ,
        lvSrc.ListItems(i).Text
    Next

    lvSrc.Refresh
    lvHitung.Refresh
    Dim k As Long
    Dim l As ListItem

```

```

        Dim m As Long

        For i = 2 To
lvSrc.ColumnHeaders.Count - 1
            If lvSrc.ColumnHeaders(i).Text <> "bmMin" And
lvSrc.ColumnHeaders(i).Text <> "bmMax" Then
                Set l =
                lvdest.ListItems.Add(, ,
                lvSrc.ColumnHeaders(i).Text)
                For j = 1 To
                lvSrc.ListItems.Count 'label
                    l.ListSubItems.Add , ,
                    lvSrc.ListItems(j).ListSubItems(i - 1).Text
                Next
            Else
                For k = 3 To lvSrc.ListItems.Count
                    If lvSrc.ListItems(k).Text = "S1" Then
                        Exit For
                    Set l = lvdest.ListItems.Add(, ,
                    lvSrc.ColumnHeaders(i).Text)
                    l.ListSubItems.Add , , -1
                    l.ListSubItems.Add , ,
                    lvSrc.ListItems(k).ListSubItems(i - 1).Text
                Next
                For m = 3 To lvSrc.ListItems.Count
                    l.ListSubItems.Add , , 0
                Next
                'bmmax
                Set l = lvdest.ListItems.Add(, ,
                lvSrc.ColumnHeaders(i + 1).Text)

                l.ListSubItems.Add , , 0
                l.ListSubItems.Add , ,
                lvSrc.ListItems(k).ListSubItems(i).Text
                For m = 3 To
                lvSrc.ListItems.Count
                    l.ListSubItems.Add , , 0
                Next
                Next
                Dim bStart As Boolean
                j = 0
                For i = 1 To
                lvHitung.ListItems.Count
                    If lvHitung.ListItems(i).Text =
                    "bmMin" Then
                        lvHitung.ListItems(i).ListSubItems(3 +
                    j).Text = "1"
                    End If
                    If lvHitung.ListItems(i).Text =
                    "bmMax" Then
                        lvHitung.ListItems(i).ListSubItems(3 +
                    j).Text = "1"
                        j = j + 1
                    End If
                Next
                bStart = False

```

```

        For i = 2 To
    lvHitung.ColumnHeaders.Count
        If
    lvHitung.ColumnHeaders(i).Text = "S1"
Then
        bStart = True
        Exit For
    End If
Next

j = i
k = 2
For i = j To
lvHitung.ColumnHeaders.Count - 1
    If k > lvHitung.ListItems.Count
Then Exit For
    If bStart = True Then
        If k Mod 2 = 0 Then

lvHitung.ListItems(k).ListSubItems(i -
1).Text = "-1"
        Else

lvHitung.ListItems(k).ListSubItems(i -
1).Text = "1"
        End If
        k = k + 1
    End If
Next

bStart = False
For i = 2 To
lvHitung.ColumnHeaders.Count
    If
lvHitung.ColumnHeaders(i).Text = "A1"
Then
    bStart = True
    Exit For
End If
Next

j = i
k = 2
For i = j To
lvHitung.ColumnHeaders.Count
    If bStart = True Then

lvHitung.ListItems(k).ListSubItems(i -
1).Text = "1"

k = k + 2
End If
Next

End Sub

Function appendRsLvw(lv As ListView, rs
As ADODB.Recordset)
    Dim lvItem As ListItem
    Dim i As Integer
    If rs.EOF Or rs.BOF Then Exit
Function

    Set lvItem = lv.ListItems.Add(, ,
rs.Fields(0).Value)
    For i = 1 To rs.Fields.Count - 1
        lvItem.ListSubItems.Add(, ,
rs.Fields(i).Value)
    Next
End Function

Function getBahan(nama_bahan As String)
As ADODB.Recordset
    Dim rs As New ADODB.Recordset
    Dim sql As String
    sql = "select "
    sql = sql & " nama_bahan as nama "
    sql = sql & ", harga "
    sql = sql & ", energi_metabolis as
em"
    sql = sql & ", energi_metabolis as
emMax"
    sql = sql & ", protein_kasar as pk"
    sql = sql & ", protein_kasar as
pkMax"
    sql = sql & ", lemak_kasar as lk"
    sql = sql & ", lemak_kasar as lkMax"
    sql = sql & ", serat_kasar as sk"
    sql = sql & ", serat_kasar as skMax"
    sql = sql & ", kalsium as ca"
    sql = sql & ", kalsium as caMax"
    sql = sql & ", phospor_tersedia as
p_tersedia"
    sql = sql & ", phospor_tersedia as
p_Max"
    sql = sql & ", lisin as lys"
    sql = sql & ", lisin as lysMax"
    sql = sql & ", metionin as met"
    sql = sql & ", metionin as metMax"
    sql = sql & ", bm_min/100 as bmMin"
    sql = sql & ", bm_max/100 as bmMax"
    sql = sql & " from bahan "
    sql = sql & " where nama_bahan = '" &
nama_bahan & "'"

    rs.Open sql, cn, adOpenKeyset
    Set getBahan = rs
    Set rs = Nothing
End Function

Function getUnggas(unggas As String) As
ADODB.Recordset
    Dim rs As New ADODB.Recordset
    Dim sql As String
    sql = "select "
    sql = sql & " unggas as nama "
    sql = sql & ", -1* harga as harga "
    sql = sql & ", em"
    sql = sql & ", emMax"
    sql = sql & ", pk"
    sql = sql & ", pkMax"
    sql = sql & ", lk"
    sql = sql & ", lkMax"
    sql = sql & ", sk"
    sql = sql & ", skMax"
    sql = sql & ", ca"
    sql = sql & ", caMax"
    sql = sql & ", p_tersedia"
    sql = sql & ", p_Max"
    sql = sql & ", lys"
    sql = sql & ", lysMax"
    sql = sql & ", met"
    sql = sql & ", metMax"
    sql = sql & ", 0 as bmMin"
    sql = sql & ", 0 as bmMax"
    sql = sql & " from batnut "
    sql = sql & " where unggas = '" &
unggas & "'"

    rs.Open sql, cn, adOpenKeyset
    Set getUnggas = rs
    Set rs = Nothing
End Function

Function getBlankRs() As ADODB.Recordset
    Dim rs As New ADODB.Recordset
    Dim sql As String
    sql = "select "

```

```

sql = sql & " 'C' as nama "
sql = sql & ", 0 as harga "
sql = sql & ", 1 as em"
sql = sql & ", 0 as emMax"
sql = sql & ", 1 as pk"
sql = sql & ", 0 as pkMax"
sql = sql & ", 1 as lk"
sql = sql & ", 0 as lkMax"
sql = sql & ", 1 as sk"
sql = sql & ", 0 as skMax"
sql = sql & ", 1 as ca "
sql = sql & ", 0 as caMax "
sql = sql & ", 1 as p_tersedia"
sql = sql & ", 0 as p_Max"
sql = sql & ", 1 as lys "
sql = sql & ", 0 as lysMax "
sql = sql & ", 1 as met "
sql = sql & ", 0 as metMax "
sql = sql & " from batnut "

rs.Open sql, cn, adOpenKeyset
Set getBlankRs = rs
Set rs = Nothing
End Function

Function TransposeDim(v As Variant) As Variant
    Dim x As Long, Y As Long, Xupper As Long, Yupper As Long
    Dim tempArray As Variant
    Xupper = UBound(v, 2)
    Yupper = UBound(v, 1)

    ReDim tempArray(Xupper, Yupper)
    For x = 0 To Xupper
        For Y = 0 To Yupper
            tempArray(x, Y) = v(Y, x)
        Next Y
    Next x
    TransposeDim = tempArray
End Function

Private Sub faktor_pengali()
    Dim i As Integer
    For i = 2 To lvHitung.ListItems.Count - 2
        temp(i) = Round(-
            (Val(lvHitung.ListItems(i).ListSubItems(
            posPivotX).Text) / elpivot), 4)
        If i = (posPivotY) Then temp(i) =
            Round((1 / elpivot) - 1, 4)
    Next
    Call Nilai_harga_baru
End Sub

Private Sub Nilai_harga_baru()
    Dim elbaris(3500) As Double
    Dim i As Integer
    Dim j As Integer
    Dim q As ListItem
    Dim r As ListItem
    Dim m As Integer
    Dim zero As String
    Set q = ransum.ListItems.Add(, , "C")
    q.ListSubItems.Add(, , "0")
    Nilai_harga = Nilai_harga + 1
    For i = 1 To lvHitung.ListItems.Count - 1
        If Nilai_harga = 1 Then
            If i = posPivotY - 1 Then
                q.ListSubItems.Add(, , "0")
            Else
                If Val(q.ListSubItems(i).Text) <= -1
                    q.ListSubItems.Add(, , "0")
                Else
                    If i = lvHitung.ListItems.Count - 3
                        q.ListSubItems.Add(, , "0")
                    Else
                        q.ListSubItems.Add(, , "-1")
                    End If
                End If
            End If
        Else
            If i = posPivotY - 1 Then
                q.ListSubItems.Add(, ,
                    (lvHitung.ListItems(1).ListSubItems(posPivotX).Text))
            Else
                q.ListSubItems.Add(, ,
                    (List1.List(i)))
            End If
        End If
    Next
    ransum.ColumnHeaders.Add(, , "nama")
    For i = 1 To lvHitung.ListItems.Count - 2
        If i = posPivotY Then
            ransum.ColumnHeaders.Add(, ,
                (lvHitung.ColumnHeaders(posPivotX +
                1).Text))
        Else
            ransum.ColumnHeaders.Add(, ,
                lvHitung.ListItems(i).Text)
        End If
    Next
    For i = 2 To newcoll +
        lstSelectedBahan.ListCount + 2
        elbaris(i) =
            CDbl(lvHitung.ListItems(posPivotY).ListSubItems(i).Text)
        Set s = ransum.ListItems.Add(, ,
            lvHitung.ColumnHeaders(i + 1).Text)

        For j = 1 To lvHitung.ListItems.Count - 1
            s.ListSubItems.Add(, ,
                Round((lvHitung.ListItems(j).ListSubItems(i).Text) +
                    elbaris(i) * temp(j)), 4))
        Next
    Next
    For i = 1 To lvHitung.ListItems.Count
        If ransum.ListItems(i).Text = "S1" Then
            Exit For
        Next
        For j = 1 To jumlahs
            zero = ""
            zero = "S" & j
            If ransum.ListItems(j + i - 1).Text =
                zero Then
                ransum.ListItems(j + i - 1).ListSubItems(1).Text = "0"
            End If
        Next
        Dim k As Integer
        Dim l As Integer
        For k = 1 To newcoll +
            lstSelectedBahan.ListCount + 2

```

```

If ransum.ListItems(k).Text = "A1" Then
Exit For
Next
For l = 1 To (16 +
(lstSelectedBahan.ListCount * 2)) / 2
zero = ""
zero = "A" & l
If ransum.ListItems(l + k - 1).Text =
zero Then
ransum.ListItems(l + k -
1).ListSubItems(l).Text = "-1"
End If
Next
Call balikan
End Sub

Private Sub balikan()
Dim Zj As Double
Dim ZjMinCj As Double
Dim i As Long
Dim j As Long
Dim tmpZjMinCj As Double
Dim maxZjMinCj As Double
List1.Clear
For i = 1 To lvHitung.ListItems.Count -
2
List1.AddItem
(ransum.ListItems(1).ListSubItems(i).Tex
t)
Next
lvHitung.ColumnHeaders.Clear
lvHitung.ListItems.Clear
lvHitung.ColumnHeaders.Add , , " "
For i = 1 To ransum.ListItems.Count
'header baru
lvHitung.ColumnHeaders.Add , ,
ransum.ListItems(i).Text
Next
Dim k As Long
Dim l As ListItem
Dim m As Long
For i = 2 To ransum.ColumnHeaders.Count
Set l = lvHitung.ListItems.Add(, ,
ransum.ColumnHeaders(i).Text)
For j = 1 To ransum.ListItems.Count
l.ListSubItems.Add , ,
ransum.ListItems(j).ListSubItems(i -
1).Text
Next
Next
lvHitung.ListItems.Add , , "Zj"
lvHitung.ListItems.Add , , "Zj-Cj"
lvHitung.ListItems(lvHitung.ListItems.Co
unt - 1).ListSubItems.Add , , ""
For i = 2 To
lvHitung.ColumnHeaders.Count - 1
Zj = 0
For j = 1 To
lvHitung.ListItems.Count - 2
Zj = Zj +
Round(CDbl(lvHitung.ListItems(j).ListSub
Items(i).Text), 4) *
Round(CDbl(lvHitung.ListItems(j).ListSub
Items(1).Text), 4)
Next

lvHitung.ListItems(lvHitung.ListItems.Co
unt - 1).ListSubItems.Add , , Zj
Next
lvHitung.ListItems(lvHitung.ListItems.Co
unt).ListSubItems.Add , , ""
For i = 2 To
lvHitung.ColumnHeaders.Count - 1
ZjMinCj =
Round(CDbl(lvHitung.ListItems(lvHitung.L
istItems.Count -
1).ListSubItems(i).Text), 4) -
Round(CDbl(lvHitung.ListItems(1).ListSub
Items(i).Text), 4)

lvHitung.ListItems(lvHitung.ListItems.Co
unt).ListSubItems.Add , , ZjMinCj
Next
tmpZjMinCj = 0
posPivotX = 3
For i = 3 To
lvHitung.ColumnHeaders.Count - (9 +
lstSelectedBahan.ListCount)
maxZjMinCj =
Round(CDbl(lvHitung.ListItems(lvHitung.L
istItems.Count).ListSubItems(i)), 4)

If maxZjMinCj < tmpZjMinCj Then
    posPivotX = i
    tmpZjMinCj = maxZjMinCj
End If
Next
posPivotX = posPivotX

For i = 1 To lvHitung.ListItems.Count
lvHitung.ListItems(i).ListSubItems(posPi
votX).ForeColor = vbRed 'kolom terpilih
lvHitung.ListItems(i).ListSubItems(posPi
votX).Bold = True
Next
'//cari rasio kolom terpilih
Dim rasio As Double
lvHitung.ColumnHeaders.Add , , "Rasio
Terpilih"

For i = 1 To lvHitung.ListItems.Count -
2
    rasio = 0
    If
Round(CDbl(lvHitung.ListItems(i).ListSub
Items(posPivotX).Text), 4) <> 0 Then
        rasio =
Round(CDbl(lvHitung.ListItems(i).ListSub
Items(2).Text), 4) /
Round(CDbl(lvHitung.ListItems(i).ListSub
Items(posPivotX).Text), 4)

lvHitung.ListItems(i).ListSubItems.Add ,
, Round(CDbl(rasio), 4)
    Else
        lvHitung.ListItems(i).ListSubItems.Add ,
, "DIV/0!"
    End If
Next

lvHitung.ListItems(lvHitung.ListItems.Co
unt - 2).ListSubItems.Add , , ""
lvHitung.ListItems(lvHitung.ListItems.Co
unt - 1).ListSubItems.Add , , ""
lvHitung.ListItems(lvHitung.ListItems.Co
unt).ListSubItems.Add , , ""

'//cari rasio terkecil
Dim tmpRasio As Double
Dim minRasio As Double
tmpRasio = 0
For i = 2 To lvHitung.ListItems.Count - 2

```

```

If IsNumeric
(lvHitung.ListItems(i).ListSubItems
(lvHitung.ColumnHeaders.Count - 1)) Then
If
Val(lvHitung.ListItems(i).ListSubItems(1
vHitung.ColumnHeaders.Count - 1)) >= 0
Then
tmpRatio = Val
((lvHitung.ListItems(i).ListSubItems
(lvHitung.ColumnHeaders.Count - 1)))
Else

End If
Next
For i = 2 To lvHitung.ListItems.Count - 2
If
IsNumeric(lvHitung.ListItems(i).ListSubI
tems(lvHitung.ColumnHeaders.Count - 1))
Then
    minRatio =
(CDbl(lvHitung.ListItems(i).ListSubItems
(lvHitung.ColumnHeaders.Count - 1)))
    If Round(CDbl(minRatio), 4) <=
Round(CDbl(tmpRatio), 4) Then
        If minRatio > 0 Then
            posPivotY = i
            tmpRatio = minRatio
        End If
    End If
End If
Next

For i = 1 To
lvHitung.ListItems(posPivotY).ListSubIte
ms.Count
lvHitung.ListItems(posPivotY).ListSubIte
ms(i).ForeColor = vbRed 'baris terpilih
lvHitung.ListItems(posPivotY).ListSubIte
ms(i).Bold = True
Next
lvHitung.ListItems(posPivotY).ListSubIte
ms(posPivotX).ForeColor = vbBlue 'elemen
pivot
elpivot =
CDbl(lvHitung.ListItems(posPivotY).Lists
ubItems(posPivotX).Text)
ransum.ColumnHeaders.Clear
ransum.ListItems.Clear
ransum.Refresh
lvHitung.Refresh
Call looping
End Sub

Private Sub looping()
Dim cocok As Integer
Dim lebih As Double
Dim j As Integer
Dim i As Integer
cocok = lvHitung.ListItems.Count
lebih = 0
For j = 3 To 19 + (2 *
lstSelectedBahan.ListCount)
    If
Val(lvHitung.ListItems(cocok).ListSubIte
ms(j).Text) < 0 Then
        lebih = lebih +
Val(lvHitung.ListItems(cocok).ListSubIte
ms(j).Text)
    End If
Next
If lebih < 0 Then
nosolution = nosolution + 1
If nosolution = lvHitung.ListItems.Count
- 3 Then 'lvHitung.ListItems.Count - 3
    Exit Sub
End If
Call faktor_pengali
End If
End Sub

Private Sub fase2()
Dim i As Integer
Dim j As Integer
lvBahan.ColumnHeaders.Clear
lvBahan.ListItems.Clear
For i = 1 To (16 +
(lstSelectedBahan.ListCount * 2)) +
(lstSelectedBahan.ListCount) + 3
lvBahan.ColumnHeaders.Add , ,
lvHitung.ColumnHeaders(i).Text
Next
For i = 1 To lvHitung.ListItems.Count -
2
    lvBahan.ListItems.Add , ,
(lvHitung.ListItems(i).Text)
    For j = 1 To
lvHitung.ListItems.Count +
lstSelectedBahan.ListCount
        lvBahan.ListItems(i).ListSubItems.Add ,
,
lvHitung.ListItems(i).ListSubItems(j).Te
xt
    Next
Next
lvHitung.ColumnHeaders.Clear
lvHitung.ListItems.Clear
For i = 1 To lvBahan.ColumnHeaders.Count
lvHitung.ColumnHeaders.Add , ,
lvBahan.ColumnHeaders(i).Text
Next
For i = 1 To lvBahan.ListItems.Count
    lvHitung.ListItems.Add , ,
(lvBahan.ListItems(i).Text)
    For j = 1 To lvBahan.ListItems.Count
+ lstSelectedBahan.ListCount
        lvHitung.ListItems(i).ListSubItems.Add ,
,
lvBahan.ListItems(i).ListSubItems(j).Tex
t
    Next
Next
For i = 1 To lstSelectedBahan.ListCount
    lvHitung.ListItems(1).ListSubItems(2
+ i).Text = harga.List(i - 1)
Next
For i = 1 To lvBahan.ListItems.Count
    For j = 1 To
lvHitung.ColumnHeaders.Count
        If lvHitung.ListItems(i).Text =
lvHitung.ColumnHeaders(j).Text Then
            lvHitung.ListItems(i).ListSubItems(1).Te
xt =
            lvHitung.ListItems(1).ListSubItems(j -
1).Text
        End If
    Next
Next
'cari zj
Dim Zj As Double
Dim ZjMinCj As Double
Dim tmpZjMinCj As Double
Dim maxZjMinCj As Double
List1.Clear

```

```

lvHitung.ListItems.Add , , "Zj"
lvHitung.ListItems.Add , , "Zj-Cj"
lvHitung.ListItems(lvHitung.ListItems.Count - 1).ListSubItems.Add , , ""
For i = 2 To
lvHitung.ColumnHeaders.Count - 1
    Zj = 0
    For j = 1 To
lvHitung.ListItems.Count - 2
        Zj = Zj +
Round(CDbl(lvHitung.ListItems(j).ListSubItems(i).Text), 4) *
Round(CDbl(lvHitung.ListItems(j).ListSubItems(1).Text), 4)
    Next

lvHitung.ListItems(lvHitung.ListItems.Count - 1).ListSubItems.Add , , Zj
Next
lvHitung.ListItems(lvHitung.ListItems.Count).ListSubItems.Add , , ""
For i = 2 To
lvHitung.ColumnHeaders.Count - 1
    ZjMinCj =
Round(CDbl(lvHitung.ListItems(lvHitung.ListItems.Count -
1).ListSubItems(i).Text), 4) -
Round(CDbl(lvHitung.ListItems(1).ListSubItems(i).Text), 4)

lvHitung.ListItems(lvHitung.ListItems.Count).ListSubItems.Add , , ZjMinCj
Next
tmpZjMinCj = 0
posPivotX = 3
For i = 3 To
lvHitung.ColumnHeaders.Count - 1
    maxZjMinCj =
Round(CDbl(lvHitung.ListItems(lvHitung.ListItems.Count).ListSubItems(i)), 4)

    If maxZjMinCj > tmpZjMinCj Then
        posPivotX = i
        tmpZjMinCj = maxZjMinCj
    End If
Next
posPivotX = posPivotX

For i = 1 To lvHitung.ListItems.Count
    lvHitung.ListItems(i).ListSubItems(posPivotX).ForeColor = vbRed 'kolom terpilih
    lvHitung.ListItems(i).ListSubItems(posPivotX).Bold = True
Next

'//cari rasio kolom terpilih
Dim rasio As Double
lvHitung.ColumnHeaders.Add , , "Rasio Terpilih"

For i = 1 To lvHitung.ListItems.Count - 2
    rasio = 0
    If
Round(CDbl(lvHitung.ListItems(i).ListSubItems(posPivotX).Text), 4) <> 0 Then
        rasio =
Round(CDbl(lvHitung.ListItems(i).ListSubItems(2).Text), 4) /
Round(CDbl(lvHitung.ListItems(i).ListSubItems(posPivotX).Text), 4)
    Else
        lvHitung.ListItems(i).ListSubItems.Add ,
        Round(CDbl(rasio), 4)
    End If
Next

lvHitung.ListItems(lvHitung.ListItems.Count - 2).ListSubItems.Add , , ""
lvHitung.ListItems(lvHitung.ListItems.Count - 1).ListSubItems.Add , , ""
lvHitung.ListItems(lvHitung.ListItems.Count).ListSubItems.Add , , ""

//cari rasio terkecil
Dim tmpRasio As Double .
Dim minRasio As Double

tmpRasio = 0
For i = 2 To lvHitung.ListItems.Count - 2
If
IsNumeric(lvHitung.ListItems(i).ListSubItems(lvHitung.ColumnHeaders.Count - 1))
Then
If
Val(lvHitung.ListItems(i).ListSubItems(lvHitung.ColumnHeaders.Count - 1)) >= 0
Then
    tmpRasio =
Val((lvHitung.ListItems(i).ListSubItems(lvHitung.ColumnHeaders.Count - 1)))
Else
End If
Next
For i = 2 To lvHitung.ListItems.Count - 2
    If
IsNumeric(lvHitung.ListItems(i).ListSubItems(lvHitung.ColumnHeaders.Count - 1))
Then
        minRasio =
(CDbl(lvHitung.ListItems(i).ListSubItems(lvHitung.ColumnHeaders.Count - 1)))
        If Round(CDbl(minRasio), 4) <=
Round(CDbl(tmpRasio), 4) Then
            If minRasio > 0 Then
                posPivotY = i
                tmpRasio = minRasio
            End If
        End If
    End If
Next

For i = 1 To
lvHitung.ListItems(posPivotY).ListSubItems.Count
    lvHitung.ListItems(posPivotY).ListSubItems(i).ForeColor = vbRed 'baris terpilih
    lvHitung.ListItems(posPivotY).ListSubItems(i).Bold = True
Next
lvHitung.ListItems(posPivotY).ListSubItems(posPivotX).ForeColor = vbBlue 'elemen pivot

```

```

elpivot2 =
CDbl(lvHitung.ListItems(posPivotY).ListSubItems(posPivotX).Text)
ransum.ColumnHeaders.Clear
ransum.ListItems.Clear
ransum.Refresh
lvHitung.Refresh
For i = 1 To lvHitung.ListItems.Count - 2
List1.AddItem
(lvHitung.ListItems(i).ListSubItems(1).Text)
Next
Call fp_fase2
End Sub

Private Sub fp_fase2()
Dim i As Integer
For i = 2 To lvHitung.ListItems.Count - 2
temp(i) = Round(-
(Val(lvHitung.ListItems(i).ListSubItems(posPivotX).Text) / elpivot2), 4)
If i = (posPivotY) Then temp(i) =
Round((1 / elpivot2) - 1, 4)
Next
Call nb_fase2
End Sub

Private Sub nb_fase2()
Dim elbaris(3500) As Double
Dim i As Integer
Dim j As Integer
Dim q As ListItem
Dim r As ListItem
Dim m As Integer
Dim zero As String
Set q = ransum.ListItems.Add(, , "C")
q.ListSubItems.Add(, , "0")
Nilai_harga = Nilai_harga + 1
For i = 1 To lvHitung.ListItems.Count - 1
If Nilai_harga = 1 Then
    If i = posPivotY - 1 Then
        q.ListSubItems.Add(, ,
(lvHitung.ListItems(1).ListSubItems(posPivotX).Text)
    Else
        If Val(q.ListSubItems(i).Text) <= -1
Then
        q.ListSubItems.Add(, , ("0"))
        Else
        If i = lvHitung.ListItems.Count - 3
Then
        q.ListSubItems.Add(, , ("0"))
        Else
        q.ListSubItems.Add(, , ("~1"))
        End If
        End If
        End If
    Else
        If i = posPivotY - 1 Then
        q.ListSubItems.Add(, ,
(lvHitung.ListItems(1).ListSubItems(posPivotX).Text)
        Else
        q.ListSubItems.Add(, ,
(List1.List(i))
        End If
    End If
Next
ransum.ColumnHeaders.Add(, , "nama"
    For i = 1 To lvHitung.ListItems.Count - 2
        If i = posPivotY Then
            ransum.ColumnHeaders.Add(, ,
(lvHitung.ColumnHeaders(posPivotX + 1).Text)
        Else
            ransum.ColumnHeaders.Add(, ,
lvHitung.ListItems(i).Text
        End If
    Next
    For i = 2 To lvBahan.ColumnHeaders.Count - 1
elbaris(i) =
CDbl(lvHitung.ListItems(posPivotY).ListSubItems(i).Text)
Set s = ransum.ListItems.Add(, ,
lvHitung.ColumnHeaders(i + 1).Text)

For j = 1 To lvHitung.ListItems.Count - 1
s.ListSubItems.Add(, ,
Round((lvHitung.ListItems(j).ListSubItems(s(i).Text) + (elbaris(i) * temp(j))), 4)

Next
Next
For i = 1 To lvHitung.ListItems.Count
If ransum.ListItems(i).Text = "S1" Then
Exit For
Next
For j = 1 To jumlahs
zero = ""
zero = "S" & j
If ransum.ListItems(j + i - 1).Text =
zero Then
ransum.ListItems(j + i - 1).ListSubItems(1).Text = "0"
End If
Next
Call balikan_fase2
End Sub

Private Sub balikan_fase2()
Dim Zj As Double
Dim ZjMinCj As Double
Dim i As Long
Dim j As Long
Dim tmpZjMinCj As Double
Dim maxZjMinCj As Double
List1.Clear
For i = 1 To lvHitung.ListItems.Count - 2
List1.AddItem
(ransum.ListItems(1).ListSubItems(i).Text)
Next
lvHitung.ColumnHeaders.Clear
lvHitung.ListItems.Clear
lvHitung.ColumnHeaders.Add(, , " ")
For i = 1 To ransum.ListItems.Count
'header baru
lvHitung.ColumnHeaders.Add(, ,
ransum.ListItems(i).Text
Next
Dim k As Long
Dim l As ListItem
Dim m As Long
For i = 2 To ransum.ColumnHeaders.Count
Set l = lvHitung.ListItems.Add(, ,
ransum.ColumnHeaders(i).Text)
For j = 1 To ransum.ListItems.Count

```

```

1.ListSubItems.Add , ,
ransum.ListItems(j).ListSubItems(i - 1).Text
Next
Next
lvHitung.ListItems.Add , , "Zj"
lvHitung.ListItems.Add , , "Zj-Cj"
lvHitung.ListItems(lvHitung.ListItems.Count - 1).ListSubItems.Add , , ""

For i = 2 To
lvHitung.ColumnHeaders.Count - 1
    Zj = 0
    For j = 1 To
        lvHitung.ListItems.Count - 2
            Zj = Zj +
Round(CDbl(lvHitung.ListItems(j).ListSubItems(i).Text), 4) *
Round(CDbl(lvHitung.ListItems(j).ListSubItems(1).Text), 4)
        Next

    lvHitung.ListItems(lvHitung.ListItems.Count - 1).ListSubItems.Add , , Zj
    Next
    lvHitung.ListItems(lvHitung.ListItems.Count).ListSubItems.Add , , ""
    For i = 2 To
        lvHitung.ColumnHeaders.Count - 1
            ZjMinCj =
Round(CDbl(lvHitung.ListItems(lvHitung.ListItems.Count - 1).ListSubItems(i).Text), 4) -
Round(CDbl(lvHitung.ListItems(1).ListSubItems(i).Text), 4)

        lvHitung.ListItems(lvHitung.ListItems.Count).ListSubItems.Add , , ZjMinCj
        Next
        tmpZjMinCj = 0
        posPivotX = 3
        For i = 3 To
            lvHitung.ColumnHeaders.Count - 1
                maxZjMinCj =
Round(CDbl(lvHitung.ListItems(lvHitung.ListItems.Count).ListSubItems(i)), 4)

                If maxZjMinCj < tmpZjMinCj Then
                    posPivotX = i
                    tmpZjMinCj = maxZjMinCj
                End If
            Next
            posPivotX = posPivotX

            For i = 1 To lvHitung.ListItems.Count
                lvHitung.ListItems(i).ListSubItems(posPivotX).ForeColor = vbRed 'kolom terpilih
                lvHitung.ListItems(i).ListSubItems(posPivotX).Bold = True
            Next
            '/cari rasio kolom terpilih
            Dim rasio As Double
            lvHitung.ColumnHeaders.Add , , "Rasio Terpilih"

            For i = 1 To lvHitung.ListItems.Count - 2
                rasio = 0
                If
Round(CDbl(lvHitung.ListItems(i).ListSubItems(posPivotX).Text), 4) <> 0 Then
                    rasio =
Round(CDbl(lvHitung.ListItems(i).ListSubItems(2).Text), 4) /
Round(CDbl(lvHitung.ListItems(i).ListSubItems(posPivotX).Text), 4)
                    lvHitung.ListItems(i).ListSubItems.Add , , Round(CDbl(rasio), 4)
                Else
                    lvHitung.ListItems(i).ListSubItems.Add , , "DIV/0!"
                End If
            Next
            lvHitung.ListItems(lvHitung.ListItems.Count - 2).ListSubItems.Add , , ""
            lvHitung.ListItems(lvHitung.ListItems.Count - 1).ListSubItems.Add , , ""
            lvHitung.ListItems(lvHitung.ListItems.Count).ListSubItems.Add , , ""

            '//cari rasio terkecil
            Dim tmpRatio As Double
            Dim minRatio As Double

            tmpRatio = 0
            For i = 2 To lvHitung.ListItems.Count - 2
                If
IsNumeric(lvHitung.ListItems(i).ListSubItems(lvHitung.ColumnHeaders.Count - 1))
                Then
                    If
Val(lvHitung.ListItems(i).ListSubItems(lvHitung.ColumnHeaders.Count - 1)) > 0
                    Then
                        tmpRatio =
Val((lvHitung.ListItems(i).ListSubItems(lvHitung.ColumnHeaders.Count - 1)))
                    Else
                End If
            Next
            For i = 2 To lvHitung.ListItems.Count - 2
                If
IsNumeric(lvHitung.ListItems(i).ListSubItems(lvHitung.ColumnHeaders.Count - 1))
                Then
                    minRatio =
(CDbl(lvHitung.ListItems(i).ListSubItems(lvHitung.ColumnHeaders.Count - 1)))
                    If Round(CDbl(minRatio), 4) <=
Round(CDbl(tmpRatio), 4) Then
                        If minRatio > 0 Then
                            posPivotY = i
                            tmpRatio = minRatio
                        End If
                    End If
                End If
            Next
            For i = 1 To
                lvHitung.ListItems(posPivotY).ListSubItems.Count
                lvHitung.ListItems(posPivotY).ListSubItems(i).ForeColor = vbRed 'baris terpilih
                lvHitung.ListItems(posPivotY).ListSubItems(i).Bold = True
            Next

```

```

lvHitung.ListItems(posPivotY).ListSubIte
ms(posPivotX).ForeColor = vbBlue 'elemen
pivot
elpivot2 =
CDbl(lvHitung.ListItems(posPivotY).ListS
ubItems(posPivotX).Text)
ransum.ColumnHeaders.Clear
ransum.ListItems.Clear
ransum.Refresh
lvHitung.Refresh
For i = 1 To lvBahan.ListItems.Count
    For j = 1 To
        lvHitung.ColumnHeaders.Count
            If lvHitung.ListItems(i).Text =
        lvHitung.ColumnHeaders(j).Text Then

            lvHitung.ListItems(i).ListSubItems(1).Te
xt =
            lvHitung.ListItems(1).ListSubItems(j -
1).Text
            End If
        Next
    Next

    If elpivot2 = 0 Then Exit Sub
    Call looping_fase2
End Sub

Private Sub looping_fase2()
Dim cocok As Integer
Dim lebih As Double
Dim j As Integer
Dim i As Integer
cocok = lvHitung.ListItems.Count
lebih = 0
For j = 3 To
    lvHitung.ColumnHeaders.Count - 1
        If
            Val(lvHitung.ListItems(cocok).ListSubIte
ms(j).Text) < 0 Then
                lebih = lebih +
            Val(lvHitung.ListItems(cocok).ListSubIte
ms(j).Text)
            End If
        Next
    If lebih < 0 Then
        nohasil = nohasil + 1
        If nohasil = lvHitung.ListItems.Count - 3 Then 'lvHitung.ListItems.Count - 3
            MsgBox "Tidak Ada Penyelesaian !",
            vbCritical, "EzFeed V 1.0.2003"
        Exit Sub
    End If
    Call fp_fase2
End If
End Sub

Private Sub updateNut()
Dim j As Integer
Dim i As Integer

For j = 0 To lstSelectedBahan.ListCount
    For i = 1 To lvHitung.ListItems.Count
        If lstSelectedBahan.List(j) =
            lvHitung.ListItems(i).Text Then
            If
                Val(lvHitung.ListItems(i).ListSubItems(2
)) < 0 Then
                    hsl(j).Text = "0"
                ElseIf
                    Val(lvHitung.ListItems(i).ListSubItems(2
)) > 1 Then
                        hsl(j).Text = "0"
                    Else
                        hsl(j).Text =
                    Round(Val(lvHitung.ListItems(i).ListSubI
tems(2)) * 100, 3)
                        hsl(j).ForeColor = &H0&
                        hsl(j).FontBold = True
                    End If
                ElseIf Val(hsl(j).Text) <
                    Val(lblmin(j).Caption) Then
                        hsl(j).Text = lblmin(j).Caption
                        hsl(j).ForeColor = &H0&
                        hsl(j).FontBold = True
                    End If
                Next
                Next

                toths1.Text = Val(hsl(0).Text) +
                Val(hsl(1).Text) + Val(hsl(2).Text) +
                Val(hsl(3).Text) + Val(hsl(4).Text) +
                Val(hsl(5).Text) + Val(hsl(6).Text) +
                Val(hsl(7).Text) + Val(hsl(8).Text) +
                Val(hsl(9).Text) + Val(hsl(10).Text) +
                Val(hsl(11).Text) + Val(hsl(12).Text) +
                Val(hsl(13).Text) + Val(hsl(14).Text)
                Dim tot_100 As Single
                tot_100 = 100 / Val(toths1.Text)
                For j = 0 To lstSelectedBahan.ListCount
                    Komposisi(j).Text =
                    Round(Val(hsl(j).Text) * tot_100, 3)
                Next
                For j = 0 To lstSelectedBahan.ListCount
                    If Val(Komposisi(j).Text) >
                    Val(lblmax(j).Caption) Then
                        Komposisi(j).Text =
                    lblmax(j).Caption
                    End If
                Next
                Totalkomposisi.Text =
                Val(Komposisi(0).Text) +
                Val(Komposisi(1).Text) +
                Val(Komposisi(2).Text) +
                Val(Komposisi(3).Text) +
                Val(Komposisi(4).Text) +
                Val(Komposisi(5).Text) +
                Val(Komposisi(6).Text) +
                Val(Komposisi(7).Text) +
                Val(Komposisi(8).Text) +
                Val(Komposisi(9).Text) +
                Val(Komposisi(10).Text) +
                Val(Komposisi(11).Text) +
                Val(Komposisi(12).Text) +
                Val(Komposisi(13).Text) +
                Val(Komposisi(14).Text)
                Dim eM As Single
                Dim sum0 As Double
                Dim sum1 As Double
                Dim sum2 As Double
                Dim sum3 As Double
                Dim sum4 As Double
                Dim sum5 As Double
                Dim sum6 As Double
                Dim sum7 As Double
                Dim sum8 As Double
                Dim sum9 As Double
                Dim sum10 As Double
                Dim sum11 As Double
                Dim sum12 As Double
                Dim sum13 As Double
                Dim sum14 As Double
                Dim sum15 As Double
                Dim harga As Double
                sum0 = 0

```

```

sum1 = 0
sum2 = 0
sum3 = 0
sum4 = 0
sum5 = 0
sum6 = 0
sum7 = 0
sum8 = 0
sum9 = 0
sum10 = 0
sum11 = 0
sum12 = 0
sum13 = 0
sum14 = 0
sum15 = 0
harga = 0
For i = 0 To
lstSelectedBahan.ListCount - 1
Dim hit As String
Data1.RecordSource = "Bahan"
Data1.Refresh
hit = "select* from bahan where
nama_bahan= '" & Bahan(i).Text & "'"
Data1.RecordSource = hit
Data1.Refresh
    sum0 = sum0 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("energi_metabolis").Value)
    sum1 = sum1 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("protein_kasar_r").Value)
    sum2 = sum2 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("lemak_kasar").Value)
    sum3 = sum3 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("serat_kasar").Value)
    sum4 = sum4 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("kalsium").Value)
    sum5 = sum5 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("phospor_tersedia").Value)
    sum6 = sum6 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("metionin").Value)
    sum7 = sum7 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("lisin").Value)
    sum8 = sum8 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("natrium").Value)
    sum9 = sum9 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("chlor").Value)
    sum10 = sum10 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("magnesium").Value)
    sum11 = sum11 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("zat_besi").Value)

sum12 = sum12 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("vitamin_b12").Value)
    sum13 = sum13 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("vitamin_e").Value)
    sum14 = sum14 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("sistin").Value)
    sum15 = sum15 +
(Val(Komposisi(i).Text) / 100) *
Val(Data1.Recordset.Fields("triptophan").Value)
    harga = harga +
Val(Kg(i).Text) *
Val(Data1.Recordset.Fields("harga").Value)

Nut(0) = Round(sum0, 3)
Nut(1) = Round(sum1, 3)
Nut(2) = Round(sum2, 3)
Nut(3) = Round(sum3, 3)
Nut(4) = Round(sum4, 3)
Nut(5) = Round(sum5, 3)
Nut(6) = Round(sum6, 3)
Nut(7) = Round(sum7, 3)
Nut(8) = Round(sum8, 3)
Nut(9) = Round(sum9, 3)
Nut(10) = Round(sum10, 3)
Nut(11) = Round(sum11, 3)
Nut(12) = Round(sum12, 3)
Nut(13) = Round(sum13, 3)
Nut(14) = Round(sum14, 3)
Nut(15) = Round(sum15, 3)
Harga_ransum = Round(harga /
Val(berat.Text), 3)
Dim k As Integer
For k = 0 To Kg.Count - 1
    Kg(k).Text = Val(Komposisi(k).Text) *
berat.Text / 100
Next
Next
totalberat.Text = Val(Kg(0).Text) +
Val(Kg(1).Text) + Val(Kg(2).Text) +
Val(Kg(3).Text) + Val(Kg(4).Text) +
Val(Kg(5).Text) + Val(Kg(6).Text) +
Val(Kg(7).Text) + Val(Kg(8).Text) +
Val(Kg(9).Text) + Val(Kg(10).Text) +
Val(Kg(11).Text) + Val(Kg(12).Text)
End Sub

Private Sub Timer1_Timer()
If SOLUSI.ForeColor = &HC0& Then
    SOLUSI.ForeColor = &HFFFFFF
ElseIf SOLUSI.ForeColor = &HFFFFFF
Then
    SOLUSI.ForeColor = &HC0&
End If
End Sub

Private Sub periksa()
Dim p As Integer
For p = 0 To 7
    If Val(Nut(p).Text) <
Val(nutmin(p).Caption) Or
Val(Nut(p).Text) >
Val(nutmax(p).Caption) Then
        SOLUSI.Visible = True
    Else
        SOLUSI.Visible = False
End If
End Sub

```

```

        End If
    Next
End Sub

Private Sub Tutup()
    harga.Clear
    ransum.ColumnHeaders.Clear
    ransum.ListItems.Clear
    lstSelectedBahan.Clear
    cboUnggas.Clear
    lvBahan.ColumnHeaders.Clear
    lvBahan.ListItems.Clear
    lvHitung.ColumnHeaders.Clear
    lvHitung.ListItems.Clear
    List1.Clear
    Dim i As Double
    cn.Close
    Unload Me
End Sub

MODUL 1

Public RetVal As Long
Public Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA"
    (ByVal hWnd As Long, ByVal lpOperation
    As String, ByVal lpFile As String, ByVal
    lpParameters As String, ByVal
    lpDirectory As String, ByVal nShowCmd As
    Long) As Long
Dim i, j, k As Integer
Dim bahan_terpilih As String

Public Sub banpilfeedmix()
    Dim i As Single
    For i = 0 To 14
        frmfeedmix.Bahan(i).Text =
    MDIForm1.Lstterpilih.List(i)
    Next i
End Sub

Public Sub banpilbatbah()
    Dim i As Single
    For i = 0 To 14
        frmbatbah.Bahan(i).Text =
    MDIForm1.Lstterpilih.List(i)
    Next i
End Sub

Public Sub banpilleastcost()
    Dim i As Single
    For i = 0 To 14
        frmLeastCost.Bahan(i).Text =
    MDIForm1.Lstterpilih.List(i)
    Next i
        frmLeastCost.berat.Text =
MDIForm1.berat.Text
End Sub

Public Sub jumpil()
    Dim jumpil As String
    jumpil =
    MDIForm1.Lstterpilih.ListCount - 1
End Sub

```

MODUL HTML HELP

```

'//Modul Untuk Menampilkan Bantuan (HTML
Help)
Option Explicit

Public Const HH_HELP_CONTEXT = &HF
Private Const HH_DISPLAY_TOPIC = &H0
Private Const HH_DISPLAY_TOC = &H1
Private Const HH_DISPLAY_INDEX = &H2

```

```

Private Const HH_DISPLAY_TEXT_POPUP =
&HE

Private Type tagHH_FTS_QUERY
    cbStruct          As Long
    fUnicodeStrings  As Long
    pszSearchQuery   As String
    iProximity        As Long
    fStemmedSearch   As Long
    fTitleOnly       As Long
    fExecute          As Long
    pszWindow         As String
End Type

Public Declare Function HTMLHelp Lib
"hhctrl.ocx" Alias "HtmlHelpA" (ByVal
hWnd As Long, ByVal lpHelpFile As
String, ByVal wCommand As Long, ByVal
dwData As Long) As Long
Private Declare Function
HTMLHelpCallSearch Lib "hhctrl.ocx"
Alias "HtmlHelpA" (ByVal hWnd As Long,
ByVal lpHelpFile As String, ByVal
wCommand As Long, ByRef dwData As
tagHH_FTS_QUERY) As Long
    .pszSearchQuery = ""
    .iProximity = 0&
    .fStemmedSearch = 0&
    .fTitleOnly = 0&
    .fExecute = 1&
    .pszWindow = ""
End With
End Function

```

LAMPIRAN 2
Petunjuk Penggunaan Program EzFeed V. 1.0

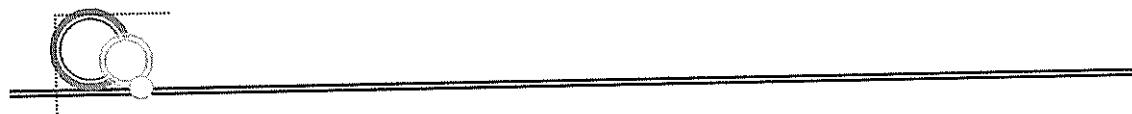
PETUNJUK PENGGUNAAN PROGRAM EzFeed

VERSI 1.0

Oleh :
Budi Aprilianto Kusnandar



**PROGRAM STUDI NUTRISI DAN MAKANAN TERNAK
DEPARTEMEN ILMU NUTRISI DAN MAKANAN TERNAK
FAKULTAS PETERNAKAN
INSTITUT PERTANIAN BOGOR
2004**



EzFeed's How To....

Selamat Datang Di Petunjuk Penggunaan Program EzFeed V 1.0

Petunjuk penggunaan program ini berisi informasi mengenai :

- Informasi singkat mengenai program
- Fasilitas dan kebutuhan minimal sistem pengoperasian
- Cara untuk :
 - Menambah, mengubah atau memperbaharui data
 - Formulasi ransum FeedMix
 - Formulasi Ransum LeastCost
- Keluar dari Program

Program ini Dibuat dalam Rangka Memenuhi Tugas Akhir Akademik Sebagai
Salah Satu Syarat Meraih Gelar Sarjana Peternakan (S.Pt.) di Lingkungan

Departemen Ilmu Nutrisi dan Makanan Ternak

Fakultas Peternakan

Institut Pertanian Bogor

Catatan :

Fasilitas ini hanya berisi tentang Cara pengoperasian dari program Formulasi Ransum EzFeed v 1.0.2003. Untuk petunjuk selengkapnya, gunakan fasilitas bantuan yang tersedia pada program EzFeed.



Informasi Singkat Mengenai Program Formulasi Ransum EzFeed V 1.0.2003

Program Formulasi Ransum ini diberi nama "EzFeed" yang diharapkan nantinya program ini dapat dengan mudah dipergunakan oleh orang yang masih awam sekalipun. Program ini berbasiskan Microsoft Windows™ sehingga pengguna tidak perlu menggunakan kode-kode perintah yang harus diketikkan secara manual seperti halnya pada program yang masih berbasiskan DOS.

Program ini dibuat berdasarkan metode Simplex Linear Programming dengan fungsi tujuan meminimalkan biaya pakan. Metode ini dipilih setain karena dapat meminimalkan biaya pakan, variabel penggunaan bahan pakannya dapat dibuat dinamis sesuai dengan keinginan pengguna.

Program ini menggunakan 8 (delapan) faktor pembatas, antara lain: Energi Metabolis, Protein kasar, Lemak kasar, Serat kasar, Kalsium, Phosphor tersedia, asam amino Metionin dan asam amino Lisin.



Fasilitas dan Kebutuhan Minimal Sistem Operasional

Program ini dilengkapi dengan beberapa fasilitas yang dapat dipergunakan oleh pengguna program ini melakukan formulasi ransum unggas. Fasilitas tersebut terbagi menjadi dua, yaitu fasilitas utama berupa Edit data, formulasi secara LeastCost dan FeedMix. Sedangkan fasilitas kedua yang merupakan fasilitas pendukung terdiri dari menu Tips, Kalkulator, Bantuan, Buka Data Tersimpan dan Status basis data.

Untuk pengoperasiannya, program ini memerlukan komputer dengan spesifikasi minimal :

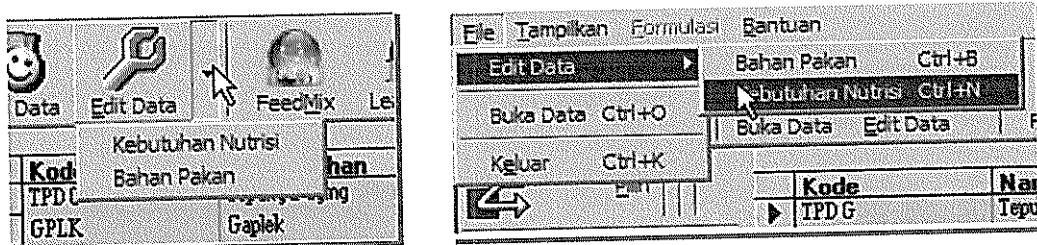
- Pentium 100;
- Memori 32 MB;
- Ruang HardDisk sebesar 50 MB;
- Monitor yang mampu mendukung resolusi 800 x 600 pixels; dan
- Sistem Operasi Microsoft Windows 98 atau yang lebih baru.



Cara Pengoperasian

A. Menambah, Mengubah atau Memperbarui data

- ✓ Pada halaman utama pilih Icon Edit Data | Kebutuhan Nutrisi atau Menu File | Edit Data | Kebutuhan Nutrisi;



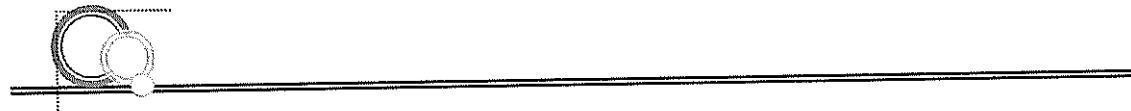
- ✓ Gunakan Tanda Panah maju-mundur untuk melihat data yang sudah ada;
- ✓ Klik Tombol "Tambah Data" untuk menambah data dan setelah semua terisi (semua data harus diisi meskipun nilainya 0) klik tombol "Simpan". Kemudian tekan tombol "Refresh";

The dialog box is titled 'Edit Data (Bantuan Nutrisi)' and has a sub-section title 'Kebutuhan Nutrisi Unggas'. It contains a table with columns for 'Parameter' (Parameter) and 'Nilai (%)' (Value (%)). The table includes rows for Energi Metabolit, Protein Kasar, Lemak Kasar, Serat Kasar, Kalsium, Fosfor Tereduksi, Asam Amino Lissin, and Asam Amino Nekton. At the bottom are buttons for 'Tambah Data' (Add Data), 'Batal' (Cancel), 'Simpan' (Save), 'Hapus' (Delete), and 'Tutup' (Close). A note at the bottom left says '(C) 2003 by Budi Aprilianto Kusnandar'.

Parameter	Nilai (%)
Energi Metabolit	2905
Protein Kasar	16
Lemak Kasar	1
Serat Kasar	4
Kalsium	0.95
Fosfor Tereduksi	0.45
Asam Amino Lissin	1.6
Asam Amino Nekton	2.8

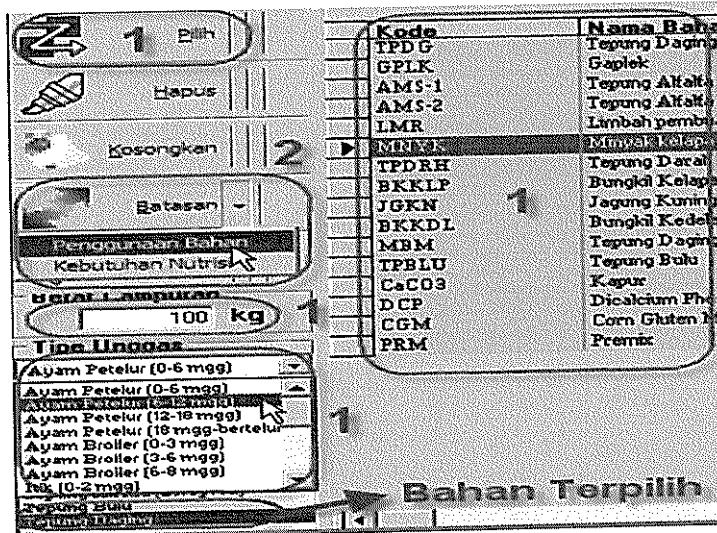
- ✓ Untuk mengubah data yang sudah ada, ketikkan nilai yang akan diperbarui, kemudian tekan tombol simpan;
- ✓ Untuk menghapus data, pada data yang terlihat (gunakan tombol panah maju-mundur untuk memposisikan data) kemudian tekan tombol hapus;
- ✓ Untuk keluar dari bagian ini, cukup tekan tombol "Tutup".

Cara ini juga berlaku untuk data bahan pakan

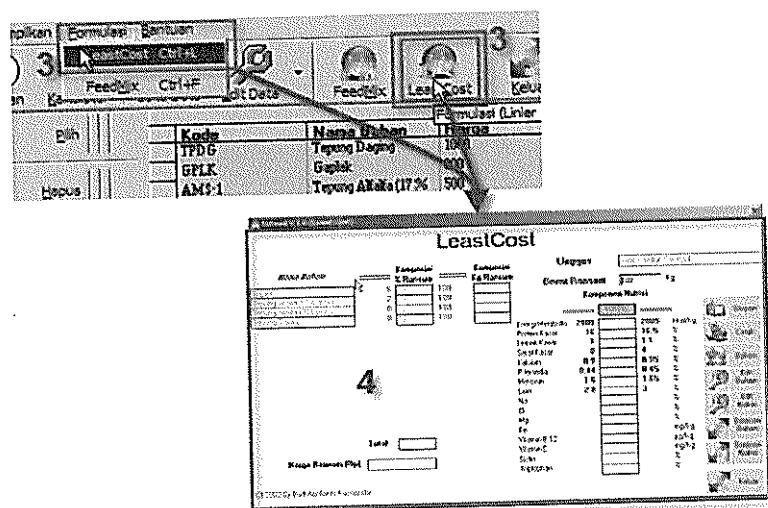


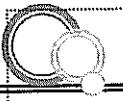
B. Cara Formulasi Ransum secara LeastCost

- Sebelum melakukan formulasi, pilih beberapa bahan yang akan digunakan dalam formulasi dengan cara meng-klik ganda pada tabel bahan atau dengan meng-klik tombol "Pilih" pada toolbar 2 serta memilih jenis unggas yang akan dipergunakan dan menentukan berat ransum;



- Setelah memilih bahan, tentukan batasan penggunaan bahan dan batasan kebutuhan nutrisi unggas;
- Setelah selesai, tekan tombol LeastCost pada toolbar 1 atau tekan menu Formulasi | LeastCost sampai muncul form LeastCost. Ulangi langkah-langkah tersebut untuk mendapatkan hasil yang optimal.





C. Cara Formulasi Ransum secara FeedMix

- ❖ Seperti halnya dengan formulasi secara LeastCost, pengguna diharuskan untuk memilih bahan yang dipergunakan dalam formulasi dengan meng-klik ganda pada tabel bahan atau dengan menekan tombol “pilih” pada toolbar 2;

Bantuan		Kalkulator	Buka Data	Edit Data	Feedmix	Least
	1	Pilih				
		Hapus				
		Kosongkan				
		Batasan	▼			
		Refresh				
Berat Campuran		1	100	kg		
Tipe Bungkus		1				
Pengkil Ketepa						
Dicalcium Phospat						
Gapilek						
Jagung Kuning						
Minyak Kelapa						
Tepung Datarah						
Bungkil Kelapa						
Jagung Kuning						
Bungkil Kedelai						
Tepung Daging tulang						
Tepung Bulu						
Kapur						
Dicalcium Phospat						
Corn Gluten Meal						
Premix						
Bahan Terpilih						

- ❖ Setelah itu, tentukan berat ransum yang akan digunakan;
 - ❖ Tekan tombol FeedMix atau pilih menu Formulasi | FeedMix sampai muncul form FeedMix;

The screenshot shows the 'LeastCost' software interface. The menu bar includes 'File', 'Tampilan', 'Formulas', 'Bantuan', 'LeastCost Child', 'LeastCost Child Data', 'Feedback', 'LeastCost', and 'Keluar'. The toolbar features icons for 'Bantuan', 'Kalkulator', 'TPD G', 'Nama Bahan', 'Technologi', '1000', and '92'. The main window displays a table with columns: Kode Bahan, Nama Bahan, Kuantitas, Satuan, Harga Satuan, Biaya Bahan, and Status Pemasukan. A context menu is open over the last row of the table, listing options: 'Pilih', 'Edit', 'Hapus', 'Ubah Status', 'Edit Bahan', 'Edit Paket', 'Cetak', 'Print Preview', 'Bantuan', 'Feedback', and 'Keluar'.



- ❖ Setelah muncul, masukkan nilai persentase (%) komposisi bahan dalam ransum, maka komputer akan langsung memperhitungkan nilai komposisi kimia ransum tersebut. Ulangi langkah-langkah tersebut untuk mendapatkan ransum yang diinginkan.

FeedMix

Nama Bahan <input type="text"/> <input type="text"/>	Komposisi % Ransum <input type="text"/> <input type="text"/>	Komposisi Kg Ransum <input type="text"/> <input type="text"/>	Berat Ransum [100] Kg <input type="text"/>	Komponen Nutrisi Energi Metabolis: <input type="text"/> kcal/kg Protein Kasar: <input type="text"/> % Lemak Kasar: <input type="text"/> % Serat Kasar: <input type="text"/> % Kalsium: <input type="text"/> % Ptersedia: <input type="text"/> % Na: <input type="text"/> % Cl: <input type="text"/> % Mg: <input type="text"/> % Fe: <input type="text"/> mg/kg Vitamin B-12: <input type="text"/> µg/kg Vitamin E: <input type="text"/> mg/kg Melatonin: <input type="text"/> % Lisin: <input type="text"/> % Sistin: <input type="text"/> % Tryptophan: <input type="text"/> %
		Simpan Cetak Bahan Edit Bahan Edit Nutrisi Balasan Bahan Balasan Nutrisi Keluar		
Selain: <input type="text"/> -73.000 Total: <input type="text"/> 27.000 %		Harga Ransum: <input type="text"/> 974.07 Rp/kg		

© 2003 By Budi Aprilianto Kusnandar

Keluar dari Program

Untuk Keluar dari Program, Cukup tekan tombol "Keluar" pada toolbar 1 atau pilih menu File | Keluar (Ctrl + K).

