

# **LAPORAN PRAKTIKUM KECERDASAN BUATAN**

## **Sistem Pakar Untuk Mendiagnosa Penyakit Pada Penderita Obesitas**



**Disusun Oleh :**

<b>Abdul Kahar Nusi</b>	<b>1515015111</b>
<b>Rulita Safitri</b>	<b>1515015114</b>
<b>Fernando Elda Pati</b>	<b>1515015115</b>
<b>Ahmad Zeka Renaldi</b>	<b>1515015046</b>

**Asisten Praktikum :**

<b>M. Denny Irawan</b>	<b>Asdar Zulkiawan</b>
<b>1415015077</b>	<b>1415015052</b>

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI DAN KOMUNIKASI**

**FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI**

**UNIVERSITAS MULAWARMAN**

**2017**

## **KATA PENGANTAR**

Puji Syukur kami panjatkan Kepada Tuhan Yang Maha Esa yang telah memberikan Rahmat dan karunia-Nya kepada kami sebagai penyusun laporan tugas akhir praktikum sehingga laporan ini dapat kami susun dan terselesaikan. Laporan ini kami buat sebagai tugas akhir praktikum dengan judul “SISTEM PAKAR UNTUK MENDIAGNOSA PENYAKIT PADA PENDERITA OBESITAS” dan sebagai salah satu syarat untuk menyelesaikan praktikum pada mata kuliah kecerdasan buatan dan

Dalam menyusun laporan tugas akhir ini, kami mengalami kesulitan dalam pemilihan materi dan perintah - perintah apa saja yang akan diimplementasikan dalam program.

Pada kesempatan ini, kami selaku penulis ingin mengucapkan terima kasih khususnya kepada Ayah dan Ibu yang telah memberikan dukungan doa yang tiada hentinya kepada kami.

Serta dalam kesempatan ini, kami selaku penulis juga mengucapkan terima kasih kepada yang terhormat:

1. Ibu Joan Angelina MT dan Masnawati MT selaku Dosen mata kuliah kecerdasan batasan yang telah mendidik dan membimbing kami sehingga ilmu pengetahuan kami tentang dasar - dasar teori dari mata kuliah Kecerdasan Buatan dapat bertambah.
2. M. Denny Irawan dan Asdar Zulkiawan selaku Asisten Laboratorium Kelompok C1, yang telah memberikan ilmu, motivasi serta waktu dan kesempatan untuk mengajarkan kami melalui responsi dan praktikum selama kurang lebih 9 pertemuan.
3. Kak Muhammad Deny Irawan selaku konsultan yang telah meluangkan waktu dan pemikiran serta saran dan kritik untuk peningkatan dan perbaikan kami dalam penyusunan konsep dan pembuatan program.
4. Seluruh dukungan teman-teman baik yang satu kelompok ataupun teman- teman yang lain atas nasehat dan saran-saran yang membangun guna perbaikan program

yang kami buat.

Kami menyadari bahwa dalam pembuatan laporan tugas akhir ini masih banyak kekurangan, untuk itu saran dan kritik yang bersifat membangun sangat kami butuhkan demi perbaikan laporan tugas akhir ini. Akhir kata, kami berharap semoga laporan tugas akhir ini dapat bermanfaat bagi kami khususnya dan para pembaca umumnya.

Samarinda, 5 Mei 2017

Penulis

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	<b>i</b>
<b>DAFTAR ISI.....</b>	<b>ii</b>
<b>DAFTAR TABEL .....</b>	<b>iii</b>
<b>DAFTAR GAMBAR.....</b>	<b>iv</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar belakang .....	1
1.2. Rumusan masalah.....	2
1.3. Batasan masalah .....	2
1.4. Tujuan dan Manfaat .....	2
<b>BAB II LANDASAN TEORI .....</b>	<b>3</b>
2.1 Fungsi-Fungsi Visual Prolog yang Digunakan .....	3
2.1.1 Unifikasi .....	3
2.1.2 Predikar Fail .....	3
2.1.3 Data Objek Sederhana .....	4
2.1.4 Section Clauses .....	4
2.1.5 Section Predicates .....	4
2.1.6 Section Domain .....	4
2.1.7 Section Goal .....	5

2.2 Teori Keilmuan yang Diimplementasikan .....	5
2.2.1 Cara Mengetahui Obesitas .....	5
2.2.2 Gejala Obesitas.....	5
<b>BAB III METODOLOGI.....</b>	<b>6</b>
3.1 Alur Pembuatan Sistem .....	6
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>	<b>7</b>
4.1 Tabel Kebenaran .....	7
4.2 Analisis Aplikasi .....	7
<b>BAB V PENUTUP.....</b>	<b>13</b>
5.1 Kesimpulan .....	13
5.2 Saran .....	13
<b>DAFTAR PUSTAKA .....</b>	<b>14</b>
<b>LAMPIRAN.....</b>	<b>16</b>
1 Source Code .....	15
2 Kartu Konsul.....	29

## DAFTAR TABEL

Tabel 4.1 Tabel Kebenaran .....	7
---------------------------------	---

## DAFTAR GAMBAR

Gambar 4.2.1 Deklarasi Domains .....	7
Gambar 4.2.2 Deklarasi predikat program .....	8
Gambar 4.2.3 Deklarasi Clause .....	9
Gambar 4.2.4 Sintaks Unifikasi .....	10
Gambar 4.2.5 Data Objek Sederhana .....	10
Gambar 4.2.6 Goal Program .....	11
Gambar 4.3.1 Tampilan Awal Program .....	11
Gambar 4.3.2 Masukkan Data .....	12
Gambar 4.3.3 Pertanyaan Yang Harus Di Jawab.....	12
Gambar 4.3.4 Hasil Dan Saran .....	12

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Kecerdasan Buatan merupakan salah satu bagian ilmu komputer yang membuat agar mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia. Ilmu komputer tersebut mengembangkan perangkat lunak dan perangkat keras untuk menirukan tindakan manusia. Salah satu ilmu komputer yang penulis gunakan adalah sistem pakar, dimana sistem pakar itu sendiri merupakan sistem yang berusaha mengadopsi pengetahuan manusia yaitu dari seorang pakar ke komputer agar komputer dapat menyelesaikan masalah seperti yang dilakukan oleh para ahli atau pakar. Sistem pakar disini digunakan untuk mengetahui penyakit pada penderita obesitas melalui gejala-gejalanya serta pencegahannya.

Obesitas adalah kelebihan berat badan sebagai akibat dari penimbunan lemak tubuh yang berlebihan. Kegemukan atau obesitas selalu berhubungan dengan berbagai penyakit dan peningkatan kematian. Lambatnya para penderita obesitas mengetahui penyakit serta tidak mengetahui cara penanganannya dapat mengakibatkan penyakit tersebut menjadi lebih berat. Dengan adanya bantuan teknologi komputer sistem pakar ini diharapkan dapat membantu mempermudah dalam menentukan penyakit. Untuk menentukan jenis penyakit, sehingga penderita obesitas dapat dengan mudah dan cepat mengetahui jenis penyakit yang dikarenakan kelebihan berat badan atau obesitas beserta pencegahannya.

Dari kesimpulan di atas kami akan membuat *project* dengan judul “SISTEM PAKAR UNTUK MENDIAGNOSA PENYAKIT PADA PENDERITA OBESITAS”.



## **1.2 Rumusan Masalah**

Berdasarkan dari deskripsi masalah di atas maka permasalahan yang dapat diambil yaitu “ Bagaimana Sistem Pakar Mendiagnosa Penyakit Pada Penderita Obesitas”.

## **1.3 Batasan Masalah**

Dari judul yang kami ambil ada beberapa batasan masalah yaitu :

1. Sistem pakar ini hanya di tujukan untuk penderita obesitas
2. sistem pakar ini hanya menggunakan aplikasi visual prolog.
3. sistem pakar ini hanya untuk mengetahui berbagai macam obesitas

## **1.4. Tujuan dan Manfaat**

### **1.4.1. Tujuan**

1. Agar pasien Mengetahui jenis spesifikasi obesitas yang diderita.
2. Memberikan Solusi kepada pasien dari penyakit Obesitas yang di.derita

### **1.4.2. Manfaat**

1. Dengan program Sistem Pakar dapat menambah ilmu pengetahuan berbasis teknologi
2. Mengetahui alur kerja sebuah program sistem pakar.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Fungsi-Fungsi Visual Prolog yang Digunakan**

##### **2.1.1 Unifikasi**

Pada waktu Visual Prolog mencoba untuk mencocokkan suatu panggilan (dari sebuah subgoal) ke klausa (pada section clauses), maka proses tersebut melibatkan suatu prosedur yang dikenal dengan unifikasi (*unification*), yang mana berusaha untuk mencocokkan antara struktur data yang ada di panggilan (subgoal) dengan klausa yang diberikan. Unifikasi pada Prolog mengimplementasikan beberapa prosedur yang juga dilakukan oleh beberapa bahasa tradisional seperti melewati parameter, menyeleksi tipe data, membangun struktur, mengakses struktur dan pemberian nilai (*assignment*). Pada intinya unifikasi adalah proses untuk mencocokkan dua predikat dan memberikan nilai pada variabel yang bebas untuk membuat kedua predikat tersebut identik. Mekanisme ini diperlukan agar Prolog dapat mengidentifikasi klausa-klausa mana yang dipanggil dan mengikat (*bind*) nilai klausa tersebut ke variabel.

##### **2.1.2 Predikat fail**

Visual Prolog akan memulai lacakbalik jika ada panggilan yang gagal. Pada situasi tertentu, ada kebutuhan untuk memaksa lacakbalik dalam rangka mencari alternatif solusi. Visual Prolog menyediakan predikat khusus *fail* untuk memaksa kegagalan sehingga memicu terjadinya lacakbalik. Efek dari *fail* sama dengan efek memberikan perbandingan  $2=3$  atau subgoal yang tidak mungkin (*impossible*) lainnya.

### **2.1.3 Data objek sederhana**

Data *object* sederhana terdiri dari 2 yaitu variabel atau konstanta. Konstanta yang dimaksud tidak sama dengan konstanta simbolis yang ditulis di *section constants* pada bagian program. Yang dimaksud dengan konstanta di sini adalah apapun yang diidentifikasi sebagai sebuah *object* bukan *subject* yang nilainya bisa bervariasi, seperti sebuah karakter (*char*), angka (integer atau real) atau sebuah atom (symbol atau string).

### **2.1.4 Section Clauses**

Section clauses merupakan section yang paling penting pada program Visual Prolog. Pada section inilah kita meletakkan fakta dan aturan. Ketika mencari jawaban, Visual Prolog akan mencari dari bagian paling atas dari section clauses, melihat setiap fakta dan aturan untuk mendapat jawaban benar, hingga ke bagian paling bawah dari section ini.

### **2.1.5 Section Predicates**

Sebelum mendefinisikan predikat di section clauses, maka predikat tersebut harus dideklarasikan terlebih dahulu di section predicates. Kalau tidak, Visual Prolog tidak akan mengenal predikat yang kita tuliskan tersebut. Ketika mendeklarasikan suatu predikat, kita memberitahu Visual Prolog domain dari argumen yang dimiliki predikat tersebut.

### **2.1.6 Section Domains**

Section domains mempunyai 2 manfaat utama, yaitu pertama, kita dapat memberikan nama yang berarti untuk domain, walaupun secara internal domain tersebut sama tipenya dengan domain yang telah ada; yang kedua, kita dapat mendeklarasi domain khusus yang digunakan untuk mendeklarasikan struktur data yang tidak didefinisikan oleh standar domain. Dengan mendeklarasikan domain juga dapat mencegah kesalahan logika pada program.

### 2.1.7 Section Goal

Secara esensial, section *goal* sama dengan *body* dari sebuah aturan (*rule*), yaitu sederetan sub-sub goal. Perbedaan antara section *goal* dengan suatu aturan adalah setelah kata kunci *goal* tidak diikuti tanda :- dan Visual Prolog secara otomatis mengeksekusi *goal* ketika program dijalankan

## 2.2 Teori Keilmuan yang Diimplementasikan

### 2.2.1 Cara mengetahui obesitas

Mengetahui Obesitas? Indeks Masa Tubuh (IMT) adalah perbandingan antara berat badan dalam kg dan tinggi badan dalam meter kuadrat.

Rumus menghitung IMT

IMT : Berat Badan (Kg) X Tinggi Badan (m)<sup>2</sup>

### 2.2.2 Tabel Rumus Menghitung Obesitas

Kategori	IMT
Kurus	<18
Normal	18 – 25
Gemuk	> 26 – 28
Obesitas Ringan	> 29 – 34
Obesitas Sedang	> 34 – 40
Obesitas Berat	> 40

### 2.2.2 Tabel Rumus Menghitung Obesitas

#### 2.2.2 Gejala Obesitas

- Pipi Tampak Tembem
- Kurang Olahraga dan Aktivitas
- Makan Berlebihan dan Ngemil Pada Malam Hari
- Sering Merasa Ngantuk Pada Siang Hari
- Cepat Mengalami Kelelahan saat melakukan aktivitas ringan
- Sering Mengalami Sesak Nafas

- Sering Mengalami Stress
- Sering Pusing atau Sakit Kepala

# BAB III

## METODOLOGI

### 3.1 Alur Pembuatan Sistem



Gambar 3.1.1 Menentukan Alur Cerita Program

## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Tabel Kebenaran

No	Gejala									Kategori
	G1	G2	G3	G4	G5	G6	G7	G8	G9	
1	Y	Y	y							Gemuk
2				y	y	y				Obesitas ringan
3							y	y	y	Obesitas sedang

Tabel 4.1 Tabel Kebenaran

Keterangan : G1 – G9 Merupakan Gejala – Gejala Yang di alami pasien

#### 4.2 Analisis Aplikasi

##### A. DOMAINS

```
domains
  sum = real
  in = char
  nama = string
  group = string*
```

Gambar 4.2.1 Deklarasi Domains

Domains digunakan untuk mendeklarasikan predikat, serta juga bisa digunakan untuk deklarasi tipe data

## B. PREDICATES

```
predicates
    nondeterm run
    nondeterm start
    nondeterm anggota
    nondeterm mulai
    |
    nondeterm kurus
    nondeterm normal
    nondeterm gemuk
    nondeterm obesitasringan
    nondeterm obesitasedang
    nondeterm obesitasberat

    nondeterm salahsaran

    nondeterm sarangemuk1
    nondeterm sarangemuk2
    nondeterm sarangemuk3
    nondeterm sarangemuk4
    nondeterm sarangemuk5

    nondeterm saranringan1
    nondeterm saranringan2
    nondeterm saranringan3
    nondeterm saranringan4
    nondeterm saranringan5

    nondeterm saransedang1
    nondeterm saransedang2
    nondeterm saransedang3
    nondeterm saransedang4
    nondeterm saransedang5
```

Gambar 4.2.2 Deklarasi predikat program

Pada predikat digunakan nondeterm yang fungsinya adalah untuk mengatasi apabila pada saat run terdapat lebih dari satu data yang sama, sehingga tetap dapat muncul hasilnya. Dan cek merupakan variabel baru.



### C. CLAUSES

```
clauses
run:-
    write("[*]Diagnosa Penyakit Obesitas[*]"),nl,nl,
    write("1. Start"),nl,
    write("2. Anggota"),nl,
    readchar(Input),input(Input),nl.

input('1'):-nl,
    start.
input('2'):-nl,
    anggota.
input(X):-nl,
    X<>'1',X<>'2',
    run.

keluar('y'):-
    write("Thank You"),nl.
keluar('Y'):-
    write("Thank You"),nl.
keluar(_):-
    start.

outt('y'):-
    write("Thank You"),nl.
outt('Y'):-
    write("Thank You"),nl.
outt(_):-
    run.

name>Nama):-
    readln>Nama).
```

Gambar 4.2.3 Deklarasi klausa

Source Code diatas digunakan untuk membuat tampilan awal, read dan write diatas berfungsi untuk input data. Serta terdapat predikat cut dan fail yang berfungsi untuk mencegah lacakbalik. Dan fail digunakan untuk memaksa prolog ketika terdapat kegagalan sehingga memicu terjadinya lacakbalik.

## D. UNIFIKASI

```
run:-
    write("[*]Diagnosa Penyakit Obesitas[*]"),nl,nl,
    write("1. Start"),nl,
    write("2. Anggota"),nl,
    readchar(Input),input(Input),nl.

input('1'):-nl,
    start.
input('2'):-nl,
    anggota.
input(X):-nl,
    X<>'1',X<>'2',
    run.
```

Gambar 4.2.4 sintaks unifikasi

Program diatas menggunakan unifikasi karena prolog akan membandingkan satu per satu clause yang ada dengan goal yang diinginkan, untuk readchar(A) maksudnya yaitu untuk pilihan yang harus kita masukkan saat terdapat pilihan pada program.

## E. DATA OBJEK SEDERHANA

```
sum = real
in = char
nama = string
group = string*
```

Gambar 4.2.5 Data Objek Sederhana

Menggunakan data sederhana karena di deklarasikan secara terpisah

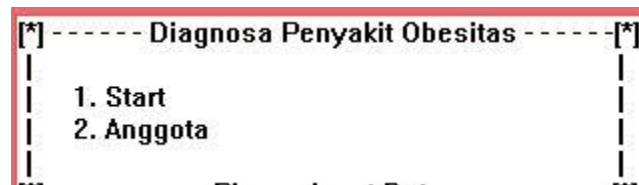
## F. GOAL

```
goal
run.
```

Gambar 4.2.6 Goal Program

Merupakan aturan yang kita buat sendiri dengan tujuan memperoleh hasil yang sesuai dengan keinginan kita.

### 4.3 Pengujian



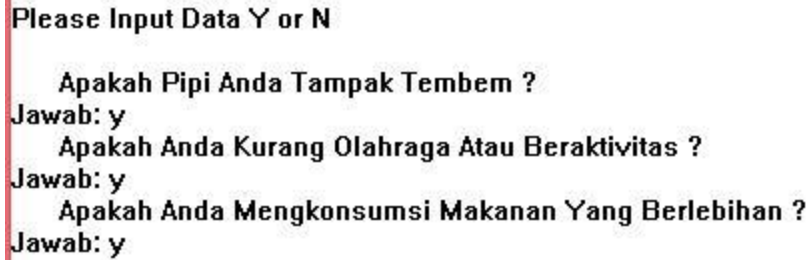
Gambar 4.3.1 Tampilan awal program

Tampilan awal program ketika pertama kali dijalankan, setelah itu akan memasukkan data sesuai dengan keinginan anda yang sudah disediakan oleh program.



Gambar 4.3.2 Memasukkan Data

Masukkan Nama anda, selanjutnya isi Berat Badan dan Tinggi Badan anda sesuai dengan keinginan, kemudian muncul pertanyaan untuk mendiagnosa apakah anda termasuk dalam orang yang obesitas atau tidak.

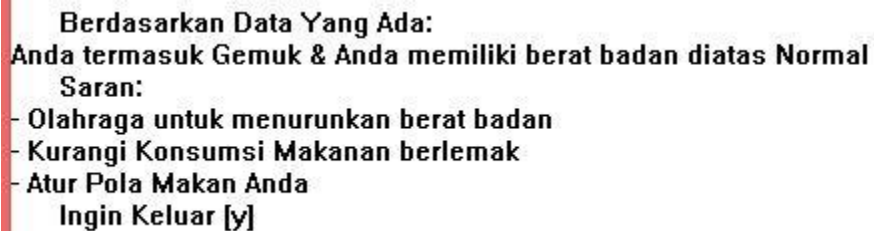


```
Please Input Data Y or N

Apakah Pipi Anda Tampak Tembem ?
Jawab: y
Apakah Anda Kurang Olahraga Atau Beraktivitas ?
Jawab: y
Apakah Anda Mengkonsumsi Makanan Yang Berlebihan ?
Jawab: y
```

Gambar 4.3.3 Pertanyaan yang harus di jawab

Setelah memasukkan data muncul pertanyaan yang hanya bias di jawab dengan huruf “Y” dan “N”, selanjutya akan muncul data sesuai dengan yang telah anda masukkan di awal program



```
Berdasarkan Data Yang Ada:
Anda termasuk Gemuk & Anda memiliki berat badan diatas Normal
Saran:
- Olahraga untuk menurunkan berat badan
- Kurangi Konsumsi Makanan berlemak
- Atur Pola Makan Anda
Ingin Keluar [y]
```

Gambar 4.3.4 Hasil Dan Saran

Setelah mengisi data dan menjawab sebuah pertanyaan, maka akan mendapatkan sebuah hasil dan saran yang sesuai dengan alur program

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Setelah melakukan analisis dan pembahasan, maka dapat diperoleh beberapa kesimpulan, antara lain; aplikasi sistem pakar ini dapat melakukan diagnosis awal terhadap suatu penyakit obesitas, aplikasi ini cukup membantu masyarakat dalam memperoleh informasi mengenai jenis-jenis penyakit obesitas dengan menggunakan rumus yang sesuai dengan rumus di atas , aplikasi sistem pakar ini sedikitnya dapat membantu meminimalisasi pengeluaran uang, tenaga dan waktu untuk memeriksa ke dokter ahli.

#### **5.2 Saran**

1. Program ini belum sempurna untuk digunakan, sangat terbuka bagi yang ingin mengembangkan program ini sehingga lebih
2. Program ini belum terlalu spesifik jika ingin digunakan untuk mengetahui lebar pinggang dan dada secara terperinci, jika ingin melihat data secara lengkap silahkan menghubungi dokter atau rumah sakit terdekat

## DAFTAR PUSTAKA

1. <http://www.kajianpustaka.com/2016/01/pengertian-jenis-faktor-penyebab-obesitas.html>
2. <http://mynewblogupdatekesehatan.blogspot.co.id/2015/12/makalah-obesitas.html>
3. <http://sittiasniar.blogspot.co.id/2014/03/makalah-obesitas.html>
4. <https://adul2008.wordpress.com/2009/04/11/obesitas/>
5. <http://gizi.depkes.go.id/download/Makalah%20Dan%20Artikel/Brosur%20Obesitas%20Rev%2001.pdf>
6. <http://dewisrikandi.com/cara-menghitung-berat-badan-ideal-dengan-cepat-tepat-benar.html>

## LAMPIRAN

### 1. Source Code

domains

sum = real

in = char

nama = string

group = string\*

predicates

nondeterm run

nondeterm start

nondeterm anggota

nondeterm mulai

nondeterm kurus

nondeterm normal

nondeterm gemuk

nondeterm obesitasringan

nondeterm obesitasedang

nondeterm obesitasberat

nondeterm salahsaran

nondeterm sarangemuk1

nondeterm sarangemuk2

nondeterm sarangemuk3

nondeterm sarangemuk4

nondeterm sarangemuk5

nondeterm saranringan1

nondeterm saranringan2

nondeterm saranringan3

nondeterm saranringan4

nondeterm saranringan5

nondeterm saransedang1

nondeterm saransedang2

nondeterm saransedang3

nondeterm saransedang4

nondeterm saransedang5

nondeterm input(in)

nondeterm keluar(in)

nondeterm outt(in)

nondeterm yes(in)

nondeterm noo(in)

nondeterm what(in)

nondeterm jawabgemuk(in,in,in)

nondeterm jawabringan(in,in,in)

nondeterm jawabsedang(in,in,in)

nondeterm name(nama)

nondeterm jenis(sum,sum,sum)



```
nondeterm name(nama)
nondeterm jenis(sum,sum,sum)
nondeterm namegroup(symbol,group)
```

clauses

run:-

```
    write("[*] ----- Diagnosa Penyakit Obesitas -----[*]"),nl,
    write("    1. Start "),nl,
    write("    2. Anggota "),nl,
    readchar(Input),input(Input).
```

input('1'):-nl,

start.

input('2'):-nl,

anggota.

input(X):-nl,

X<>'1',X<>'2',

run.

keluar('y'):-

write(" Thank You"),nl.

keluar('Y'):-

write(" Thank You"),nl.

keluar(\_):-

start.

outt('y'):-

```

start.

outt('y'):-
    write("    Thank You"),nl.

outt('Y'):-
    write("    Thank You"),nl.

outt(_):-
    run.

name>Nama):-
    readln>Nama).

%% Perhitungan IMT

jenis(BB,TB,Sum):-
    Sum = BB/(TB*TB),Sum<18,
    nl,kurus.

jenis(BB,TB,Sum):-
    Sum = BB/(TB*TB),Sum>=18,Sum<=25,
    nl,normal.

jenis(BB,TB,Sum):-
    Sum = BB/(TB*TB),Sum>25,Sum<=29,
    nl,write("[*] - - - - - Please Input Data Y or N - - - - -
[*]"),gemuk.

jenis(BB,TB,Sum):-
    Sum = BB/(TB*TB),Sum>29,Sum<=34,
    nl,write("[*] - - - - - Please Input Data Y or N - - - - -
[*]"),obesitasringan.

jenis(BB,TB,Sum):-
    Sum = BB/(TB*TB),Sum>34,Sum<=40,

```

```

jenis(BB,TB,Sum):-
    Sum = BB/(TB*TB),Sum>34,Sum<=40,
    nl,write("[*] - - - - - Please Input Data Y or N - - - - - ")
    [*]),obesitassedang.

jenis(BB,TB,Sum):-
    Sum = BB/(TB*TB),Sum>40,
    nl,obesitasberat.

%% respon
yes('y').
yes('Y').
noo('n').
noo('N').
what(_).

%% Menu
anggota:-
    namegroup(anggota,[Isi1 | [Isi2 | [Isi3 | [Isi4 | _]]]]),
    write(Isi1),nl,
    write(Isi2),nl,
    write(Isi3),nl,
    write(Isi4),nl,
    write("Ingin Keluar [y]"),nl,nl,
    readchar(Pilih),outt(Pilih).

```

```

start:-
    write("[*] - - - - - Please Input Data - - - - - [*]"),nl,
    write("    Nama: "),
    name>Nama),
    write("    Welcome ",Nama),nl,
    mulai,write(" [*] - - - - - [*]"),nl,
    write("    Ingin Keluar [y] "),nl,nl,
    readchar(Pilih),keluar(Pilih).

%% Menu awal
mulai:-
    write("    Berat Badan Anda (kg): "),
    readreal(BB),
    write("    Tinggi Badan Anda (cm): "),
    readreal(Cm),TB=Cm/100,
    jenis(BB,TB,_).

%% Type Kurus
kurus:-
    write("[*] - - - - - Hasil & Saran - - - - - [*]"),nl,
    write("    Berdasarkan Data Yang Ada:\n\tAnda Tidak termasuk Obesitas & Anda memiliki
berat badan yang tidak ideal"),nl,
    write("    Saran: \n\tOlahraga yang teratur\n\tKonsumsi Makanan Bergizi"),nl.

```

```
%% Type Berat Badan Gemuk
```

```
gemuk:-nl,
```

```
write("  Apakah Pipi Anda Tampak Tembem ?"),nl,
```

```
readchar(One),write("  Jawab: ",One),nl,
```

```
write("  Apakah Anda Kurang Olahraga Atau Beraktivitas ?"),nl,
```

```
readchar(Two),write("  Jawab: ",Two),nl,
```

```
write("  Apakah Anda Mengonsumsi Makanan Yang Berlebihan ?"),nl,
```

```
readchar(Tree),write("  Jawab: ",Tree),nl,
```

```
write("[*] ----- Hasil & Saran ----- [*]"),nl,
```

```
jawabgemuk(One,Two,Tree).
```

```
%% Solusi Gemuk
```

```
jawabgemuk(One,Two,Tree):-
```

```
noo(One),noo(Two),noo(Tree),sarangemuk1.
```

```
jawabgemuk(One,Two,Tree):-
```

```
yes(One),yes(Two),yes(Tree),sarangemuk2.
```

```
jawabgemuk(One,Two,Tree):-
```

```
yes(One),noo(Two),noo(Tree),sarangemuk3;
```

```
yes(One),yes(Two),noo(Tree),sarangemuk3;
```

```
yes(One),noo(Two),yes(Tree),sarangemuk3.
```

```
jawabgemuk(One,Two,Tree):-
```

```
noo(One),yes(Two),noo(Tree),sarangemuk4;
```

```
noo(One),yes(Two),yes(Tree),sarangemuk4.
```

```
jawabgemuk(One,Two,Tree):-
```

```
noo(One),noo(Two),yes(Tree),sarangemuk5.
```

```
jawabgemuk(One,Two,Tree):-
```

```
what(One),what(Two),what(Tree),salahsaran.
```

```
%% sarangemuk
```

```
sarangemuk1:-
```

```
    write("    Berdasarkan Data Yang Ada:\n\tAnda termasuk Gemuk & Anda memiliki berat  
    badan diatas Normal"),nl.
```

```
sarangemuk2:-
```

```
    write("    Berdasarkan Data Yang Ada:\n\tAnda termasuk Gemuk & Anda memiliki berat  
    badan diatas Normal"),nl,
```

```
    write("    Saran: \n\t- Olahraga untuk menurunkan berat badan\n\t- Kurangi Konsumsi  
    Makanan berlemak\n\t- Atur Pola Makan Anda"),nl.
```

```
sarangemuk3:-
```

```
    write("    Berdasarkan Data Yang Ada:\n\tAnda termasuk Gemuk & Anda memiliki berat  
    badan diatas Normal"),nl,
```

```
    write("    Saran: \n\t- Kurangi Konsumsi Makanan Berlebihan di Larut Malam"),nl.
```

```
sarangemuk4:-
```

```
    write("    Berdasarkan Data Yang Ada:\n\tAnda termasuk Gemuk & Anda memiliki berat  
    badan diatas Normal"),nl,
```

```
    write("    Saran: \n\t- Lakukan Olahraga yang teratur untuk menurunkan Bert Badan  
    Anda"),nl.
```

```
sarangemuk5:-
```

```
    write("    Berdasarkan Data Yang Ada:\n\tAnda termasuk Gemuk & Anda memiliki berat  
    badan diatas Normal"),nl,
```

```
    write("    Saran: \n\t- Makan yang teratur"),nl.
```

```
%% Type Obesitas Ringan
```

```
obesitasringan:-nl,
```

```
    write("    Apakah Anda Sering Mengkonsumsi Makan Ringan pada malam hari ?"),nl,
```

```
    readchar(One),write("    Jawab: ",One),nl,
```

```
    write("    Apakah Anda Sering Merasa Kantuk Pada Siang Hari ?"),nl,
```

```
    readchar(Two),write("    Jawab: ",Two),nl,
```

```

write("    Apakah Anda cepat mengalami kelelahan saat melakukan aktivitas ringan"),nl,
    readchar(Tree),write("    Jawab: ",Tree),nl,
    write("[*] ----- Hasil & Saran ----- [*]"),nl,
    jawabringen(One,Two,Tree).

%% Solusi Obesitas Ringan
jawabringen(One,Two,Tree):-
    noo(One),noo(Two),noo(Tree),saranringen1.
jawabringen(One,Two,Tree):-
    yes(One),yes(Two),yes(Tree),saranringen2.
jawabringen(One,Two,Tree):-
    yes(One),noo(Two),noo(Tree),saranringen3;
    yes(One),yes(Two),noo(Tree),saranringen3.
jawabringen(One,Two,Tree):-
    noo(One),yes(Two),yes(Tree),saranringen4;
    noo(One),yes(Two),noo(Tree),saranringen4;
    yes(One),noo(Two),yes(Tree),saranringen4.
jawabringen(One,Two,Tree):-
    noo(One),noo(Two),yes(Tree),saranringen5.
jawabringen(One,Two,Tree):-
    what(One),what(Two),what(Tree),salahsaran.

```

%% Saran Obesitas Ringan

saranringan1:-

write(" Berdasarkan Data Yang Ada: \n\tAnda termasuk Obesitas dengan tingkat Ringan"),nl.

saranringan2:-

write(" Berdasarkan Data Yang Ada: \n\tAnda termasuk Obesitas dengan tingkat Ringan"),nl,

write(" Saran: \n\t- Kurangi Konsumsi Makanan Cepat Saji\n\t- Lakukan aktivitas fisik minimal 30 menit setiap hari\n\t- Kurangi Makanan dan Minuman yang Manis"),nl.

saranringan3:-

write(" Berdasarkan Data Yang Ada: \n\tAnda termasuk Obesitas dengan tingkat Ringan"),nl,

write(" Saran: \n\t- Konsumsi buah-buah dan sayuran\n\t- Hindari susu tinggi lemak, minumlah susu rendah lemak"),nl.

saranringan4:-

write(" Berdasarkan Data Yang Ada: \n\tAnda termasuk Obesitas dengan tingkat Ringan"),nl,

write(" Saran: \n\t- Makanlah saat mulai merasa lapar\n\t- Berhentilah Makan sebelum kekenyangan"),nl.

saranringan5:-

write(" Berdasarkan Data Yang Ada: \n\tAnda termasuk Obesitas dengan tingkat Ringan"),nl,

write(" Saran: \n\t- Perbanyak Minum Air Mineral"),nl.

%% Type Obesitas Sedang

obesitasedang:-nl,

write(" Apakah Anda Sering mengalami Sesak Nafas ?"),nl,

readchar(One),write(" Jawab: ",One),nl,



```

write("    Apakah Anda Sering mengalami stress ?"),nl,
    readchar(Two),write("    Jawab: ",Two),nl,
    write("    Apakah Anda Sering Merasa Pusing atau Sakit Kepala ?"),nl,
    readchar(Tree),write("    Jawab: ",Tree),nl,
    write("[*] ----- Hasil & Saran ----- [*]"),nl,
    jawabsedang(One,Two,Tree).

%% Solusi Obesitas Sedang
jawabsedang(One,Two,Tree):-
    noo(One),noo(Two),noo(Tree),saransedang1.
jawabsedang(One,Two,Tree):-
    yes(One),yes(Two),yes(Tree),saransedang2.
jawabsedang(One,Two,Tree):-
    yes(One),noo(Two),yes(Tree),saransedang3;
    noo(One),noo(Two),yes(Tree),saransedang3.
jawabsedang(One,Two,Tree):-
    yes(One),yes(Two),noo(Tree),saransedang4;
    yes(One),noo(Two),noo(Tree),saransedang4.
jawabsedang(One,Two,Tree):-
    noo(One),yes(Two),yes(Tree),saransedang5;
    noo(One),yes(Two),noo(Tree),saransedang5.
jawabsedang(One,Two,Tree):-
    what(One),what(Two),what(Tree),salahsaran.

```

%% Saran Obesitas Sedang

saransedang1:-

```
write("      Berdasarkan Data Yang Ada: \n\tAnda termasuk Obesitas dengan tingkat  
Sedang"),nl.
```

saransedang2:-

```
write("      Berdasarkan Data Yang Ada: \n\tAnda termasuk Obesitas dengan tingkat  
Sedang"),nl,
```

```
write("      Saran: \n\t- Lakukan olahraga renang\n\t- Mengurangi makanan yang digoreng\n\t- Penuhi kebutuhan sayur dan buah"),nl.
```

saransedang3:-

```
write("      Berdasarkan Data Yang Ada: \n\tAnda termasuk Obesitas dengan tingkat  
Sedang"),nl,
```

```
write("      Saran: \n\t- Lakukan aktivitas Fitness"),nl.
```

saransedang4:-

```
write("      Berdasarkan Data Yang Ada: \n\tAnda termasuk Obesitas dengan tingkat  
Sedang"),nl,
```

```
write("      Saran: \n\t- Konsumsi Obat diet atau obat tradisional untuk menurunkan berat  
badan"),nl.
```

saransedang5:-

```
write("      Berdasarkan Data Yang Ada: \n\tAnda termasuk Obesitas dengan tingkat  
Sedang"),nl,
```

```
write("      Saran: \n\t- Lakukan Olahraga di siang hari"),nl.
```

```
%% Type Obesitas Berat
```

```
obesitasberat:-
```

```
    write("[*] ----- Hasil & Saran ----- [*]"),nl,  
    write("    Anda termasuk Obesitas Berat & Perlu Penanganan"),nl,  
    write("    Saran: \n\t- Kunsultasi dengan dokter Spesialist Obesitas\n\t- melakukan terapi  
menurunkan lemak\n\t- Atau melakukan operasi sedot lemak"),nl.
```

```
%% Salah Saran
```

```
salahsaran:-nl,  
    write("    Input Salah"),nl.
```

```
%% List Nama
```

```
    namegroup(anggota,["yaya      Nusi\n\t1515015\n\tDr.      Spesialist      Tingkatan  
Obesitas\n\t","Fernando Elda Pati\n\t1515015115\n\tDr. Spesialist Data Obesitas\n\t","Rulita  
\n\t1515015\n\tDr. Spesialist Saran Obesitas\n\t","Ahmad Zeka Renaldi\n\t1515015046\n\tDr.  
Spesialist Test Obesitas\n\t"]).
```

```
goal
```

```
run.
```

Tanggal Konsultasi : Rabu . 3 Mei 2017
Uraian : - finalisasi judul dan konsep
Tanda Tangan Asisten : 

Tanggal Konsultasi : Jumat . 5 Mei 2017
Uraian : - Penambahan Variabel - finalisasi Program.
Tanda Tangan Asisten : 