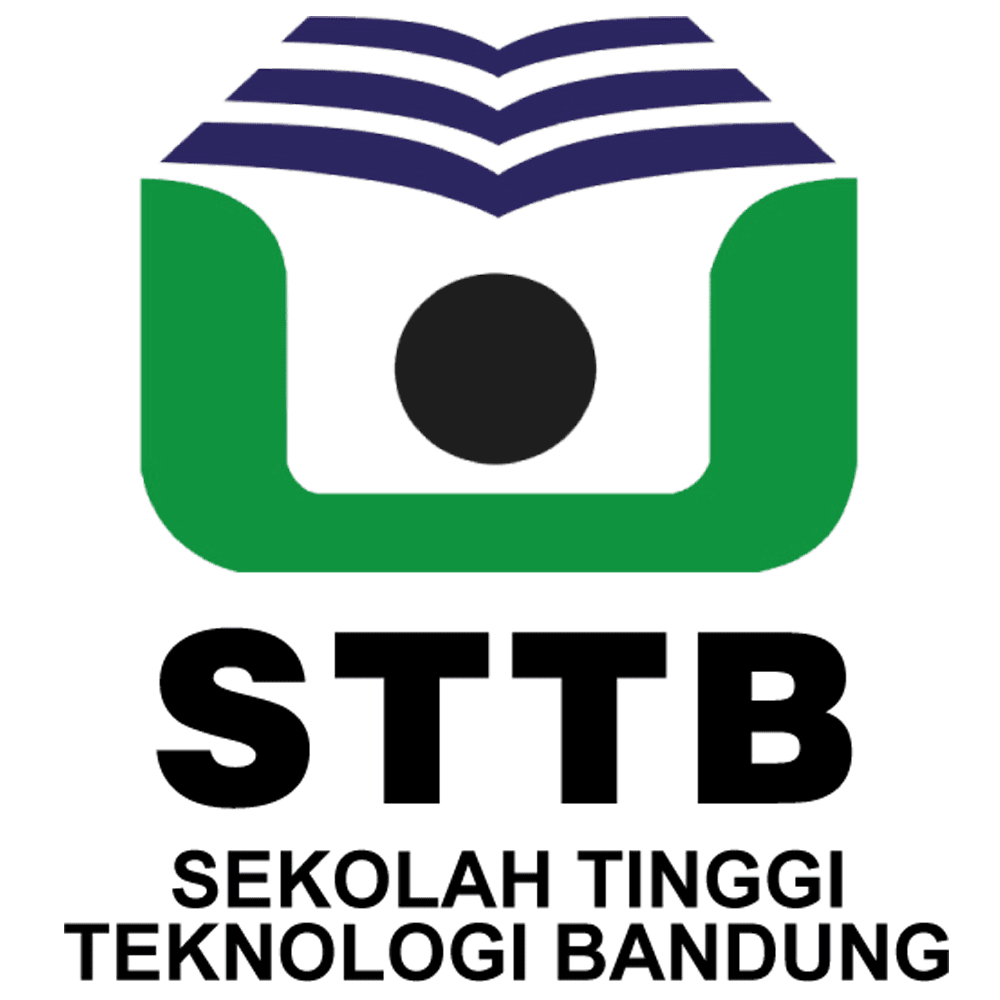
**LAPORAN PROJEK**

**ALGORITMA DAN PEMROGRAMAN**

**APLIKASI TES PENGETAHUAN UMUM (GAME EDUKASI)**

****

**Disusun Oleh :**

**Irvan Maulana 18111167**

**Wildan Ma’ruf 18111170**

**SEKOLAH TINGGI TEKNOLOGI BANDUNG**

**Jl. Soekarno-Hatta No.378, Kb. Lega, Bojongloa Kidul, Kota Bandung, Jawa Barat 40235**

**2018 / 2019**

**Kata Pengantar**

Segala puji dan syukur kami panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan limpahan rahmat-Nya maka Kami bisa menyelesaikan makalah ini dengan tepat waktu.

Berikut ini saya mempersembahkan sebuah makalah untuk memenuhi salah satu tugas pada mata kuliah Algoritma dan Pemrograman yang membahas tentang Pembuatan Aplikasi Tes Pengetahuan Umum & Psikologi dan semoga dengan dibuatnya makalah ini dapat membantu menambah ilmu pengetahuan bagi pembaca

Melalui kata pengantar ini kami terlebih dahulu meminta maaf dan memohon permakluman bilamana isi makalah ini ada kekurangan baik dalam isi maupun penulisan.Terima kasih

19 Desember 2018

Penulis,

**DAFTAR ISI**

**Kata Pengantar** i

**Daftar isi** ii

**BAB I PENDAHULUAN** 1

1.1 Latar Belakang 1

1.2 Rumusan Masalah 1

1.3 Tujuan 1

1.4 Manfaat Dari Program

1.5 Metode Pengambilan Cara

1.6 Sistematika Penulisan

**BAB** **II** **TINJAUAN PUSTAKA**

2.1 Landasan Teori

**BAB III DIAGRAM ALIR**

3.1 Pseudocode

3.2 Analisa

3.3 Flowchart

3.4 Rancangan Aplikasi

**BAB IV IMPLEMENTASI**

4.1 Deskripsi Aplikasi

4.2 Cara Kerja Program

**BAB V PENUTUP**

5.1 Kesimpulan

5.2 Saran

**DAFTAR PUSTAKA**

**BAB I**

**PENDAHULUAN**

* 1. **Latar Belakang**

Pendidikan merupakan salah satu aspek yang memegang peranan penting bagi kemajuan suatu Bangsa karena pendidikan berperan dalam menciptakan kehidupan yang cerdas, damai, terbuka, dan demokratis, serta berperan dalam memajukan kesejahteraan masyarakat.

Pembaharuan pendidikan harus selalu dilakukan untuk meningkatkan kualitas pendidikan Nasional. Oleh karena itu, kemajuan suatu bangsa, tidak pernah lepas dari peranan pendidikan yang berkualitas. Terdapat beberapa faktor yang mempengaruhi kualitas pendidikan, misalnya proses pembelajaran,sarana prasarana, pendidik, dan prosedur evaluasi yang mencakup penilaian dan pengukuran.Kualitas pendidikan tidak dapat dilepaskan dari prosedur evaluasi berupa pengukuran hasil belajar pendidikan karena pengukuran merupakan salah satu dari sekian faktor dalam sistem yang sangat menentukan keberhasilan pendidikan.

memperbaiki kualitas pendidikan dapat dilakukan dengan membuat sistem pengukuran yang baik. Prosedur pada kegiatan pengukuran harus sistematis, serta hasilnya diharapkan dapat menjelaskan kemampuan yang sebenarnya. dengan dilakukannya pengukuran, bahkan dengan mudah dapat dihimpun informasi sampai sejauh mana kita mampu mengaplikasikan, dan menganalisis materi yang diajarkan.

* + 1. **Teknologi Yang Dipakai**

1. **Pengertian Java**

Java adalah bahasa pemrograman tingkat tinggi yang berorientasi objek dan program java tersusun dari bagian yang disebut kelas, kelas terdiri dari metode-metode yang melakukan pekerjaan dan mengembalikan informasi setelah melakukan tugasnya, yang disebut dengan *java application programming interface* (API), kelas-kelas ini diorganisasikan menjadi sekelompok yang disebut paket (*package*), java API telah menyediakan fungsionalitas yang memadai utuk menciptakan *applet* dan aplikasi canggih. Jadi ada dua hal yang harus dipelajari dalam aplikasi java, yaitu mempelajari aplikasi java dan bagaimana mempergunakan kelas pada java API. Kelas merupakan satu-satunya cara menyatakan bagian eksekusi program, tidak ada cara lain.

1. **NetBeans**

NetBeans adalah suatu [serambi](https://id.wikipedia.org/wiki/Platform_(komputer)) [pengembangan perangkat lunak](https://id.wikipedia.org/wiki/Proses_pengembangan_perangkat_lunak) yang ditulis dalam [bahasa pemrograman Java](https://id.wikipedia.org/wiki/Java). Serambi Pada NetBeans, pengembangan suatu aplikasi dapat dilakukan dimulai dari setelan perangkat lunak modular bernama *modules*.

Semula, aplikasi NetBeans IDE ini diperuntukkan bagi pengembangan dalam Java. Namun, aplikasi ini juga mendukung program-program pembuatan bahasa lain secara khusus seperti PHP, C/C++ dan HTML5

NetBeans adalah alat lintas serambi serta penerapannya dijalankan pada Microsoft Windows, Mac OS X, Linux, Solaris dan serambi-serambi lainnya yang mendukung JVM yang sepadan.

* + 1. **Tes Pengetahuan Umum**

Supaya lebih simple dalam menjawab soal pertanyaan tanpa harus menulis dikertas atau buku tulis.

* + 1. **Solusi**

Dizaman yang hampir semuanya menggunakan teknologi, kami mencoba membuat program aplikasi tentang tes pengetahuan umum, supaya kita bisa lebih menghemat kertas ataupun buku tulis, seperti yang kita ketahui beberapa sekolah menengah di indonesia sudah menggunakan teknologi computer dalam melaksanakan ujiannya.

* 1. **Rumusan Masalah**
* Proses tes pengetahuan umum masih banyak yang menggunakan kertas
* Proses tes pengetahuan masih belum banyak beredar di masyarakat umum
  1. **Tujuan**
* Untuk dapat membuat aplikasi yang memudahkan dalam tes pengetahuan
* Aplikasi ini dapat memudahkan dalam penyebaran tes pengetahuan di masyarakat
  1. **Manfaat dari Program yang dibuat**

Manfaat dari program yang kami buat adalah sebagai media Hiburan juga sebagai ajang tes untuk lebih mengenali siapa diri kita melalui tes yang tersedia

* 1. **Metode Pengambilan Cara**

1.Studi Literatur = Google = - Bank Soal Pengetahuan Umum

- Contoh Coding

- Verifikasi Jawaban Soal Tes

* 1. **Sistematik Penulisan**

**1.Bab I Pendahuluan** = Menjelaskan tentang latar belakang dari aplikasi yang dibuat oleh kami.

**2.Bab II Tinjauan Pustaka =** Menjelaskan tentang landasan teori dan definisi yang terdapat di aplikasi kami.

**3.Bab III Diagram Alir =** Berisi penjelasan tentang pseudocode ,flowchart,dan analisis dari aplikasi kami**.**

**4.Bab IV Implementasi =** Berisi tentang pembahasan dan screenshot dari aplikasi yang dbuat.

**5.Bab V Penutup =** Berisi tentang kesimpulan dan saran.

**BAB II**

**TINJAUAN PUSTAKA**

**2.1 Landasan Teori**

**2.1.1 Algoritma**

Algoritma merupakan metode yang efektif yang ditujukkan pada daftar yang terbatas dari kumpulan perintah yang telah didefinisikan untuk menghitung suatu fungsi. Dalam penyelesaian masalah, ada kriteria-kriteria tertentu pada kondisi awal yang harus dipenuhi sebelum mengeksekusi algoritma. Algoritma akan dapat selalu berakhir untuk semua kondisi awal yang memenuhi kriteria. Dimulai dari nilai awal, kemudian kumpulan perintah yang pada saat dieksekusi akan memproses kondisi-kondisi yang telah ditetapkan hingga menghasilkan output dan kemudian menentukan kondisi akhir. Pada sistem komputer, algoritma merupakan gambaran langsung dari logika yang dituliskan oleh pembangun perangkat lunak agar lebih efektif dalam pencapaian target perangkat lunak tersebut, agar dapat memperoleh hasil keluaran dari masukan yang diberikan (terkadang null).

Menurut Heriyanto dan Abdul kadir ( 2005 : 6) Ada lima ciri-ciri penting yang harus dimiliki sebuah algoritma, yaitu berupa finiteness, definiteness, masukan, keluaran, dan efektivitas.

1. Finiteness, menyatakan bahwa suatu algoritma harus berakhir untuk semua kondisi setelah memproses sejumlah langkah.
2. Defineteness, menyatakan bahwa setiap langkah harus dinyatakan dengan jelas (tidak rancu atau mendua arti).
3. Masukan, Setiap algoritma dapat tidak memiliki masukan atau mempunyai satu atau beberapa masukan. Masukan merupakan suatu besaran yang diberikan di awal sebelum algoritma diproses.
4. Keluaran, Setiap algoritma memiliki keluaran, entah hanya sebuah keluaran atau banyak keluaran. Keluaran merupakan besaran yang mempunyai kaitan atau hubungan dengan masukan.
5. Efektivitas, Setiap algoritma diharapkan bersifat efektif, dalam arti semua operasi yang dilaksanakan oleh algoritma harus sederhana dan dapat dikerjakan dalam waktu yang terbatas. Secara prinsip, setiap intruksi dalam algoritma dapat dikerjakan oleh orang dengan hanya menggunakan kertas dan pensil.

Algoritma merupakan langkah detail yang ditujukan untuk komputer gunamenyelesaikan satu masalah. Namun algoritma berbeda dengan program, karenaalgoritma bersifat bebas terhadap bahasa pemrograman.Istilah algoritma berasal dari nama seorang pengarang berkebangsaan Arab bernama Abu Ja‟far Mohammed ibn Musa al Khowarizmi (tahun 790-840), yang sangat terkenalsebagai „Bapak Aljabar

* **Sejarah Java**

[Bahasa pemrograman](https://id.wikipedia.org/wiki/Bahasa_pemrograman) Java terlahir dari [The Green Project](https://id.wikipedia.org/w/index.php?title=The_Green_Project&action=edit&redlink=1), yang berjalan selama 18 bulan, dari awal tahun [1991](https://id.wikipedia.org/wiki/1991) hingga musim panas [1992](https://id.wikipedia.org/wiki/1992). Proyek tersebut belum menggunakan versi yang dinamakan [Oak](https://id.wikipedia.org/wiki/Oak). Proyek ini dimotori oleh Patrick Naughton, Mike Sheridan, dan [James Gosling](https://id.wikipedia.org/wiki/James_Gosling), beserta sembilan pemrogram lainnya dari [Sun Microsystems](https://id.wikipedia.org/wiki/Sun_Microsystems). Salah satu hasil proyek ini adalah maskot *Duke* yang dibuat oleh [Joe Palrang](https://id.wikipedia.org/w/index.php?title=Joe_Palrang&action=edit&redlink=1).

Pertemuan proyek berlangsung di sebuah gedung perkantoran *Sand Hill Road* di [Menlo Park](https://id.wikipedia.org/w/index.php?title=Menlo_Park&action=edit&redlink=1). Sekitar musim panas 1992 proyek ini ditutup dengan menghasilkan sebuah program *Java Oak* pertama, yang ditujukan sebagai pengendali sebuah peralatan dengan teknologi layar sentuh (*touch screen*), seperti pada PDA sekarang ini. Teknologi baru ini dinamai "\*7" (*Star Seven*).Setelah era *Star Seven* selesai, sebuah anak perusahaan [Tv kabel](https://id.wikipedia.org/wiki/Tv_kabel) tertarik ditambah beberapa orang dari proyek The Green Project. Mereka memusatkan kegiatannya pada sebuah ruangan kantor di 100 Hamilton Avenue, [Palo Alto](https://id.wikipedia.org/wiki/Palo_Alto).

Perusahaan baru ini bertambah maju: jumlah karyawan meningkat dalam waktu singkat dari 13 menjadi 70 orang. Pada rentang waktu ini juga ditetapkan pemakaian [Internet](https://id.wikipedia.org/wiki/Internet) sebagai medium yang menjembatani kerja dan ide di antara mereka. Pada awal tahun 1990-an, Internet masih merupakan rintisan, yang dipakai hanya di kalangan [akademisi](https://id.wikipedia.org/wiki/Akademisi) dan [militer](https://id.wikipedia.org/wiki/Militer).

Mereka menjadikan [perambah](https://id.wikipedia.org/wiki/Perambah) (*browser*) [Mosaic](https://id.wikipedia.org/w/index.php?title=Mosaic&action=edit&redlink=1) sebagai landasan awal untuk membuat perambah Java pertama yang dinamai Web Runner, terinsipirasi dari film 1980-an, Blade Runner. Pada perkembangan rilis pertama, Web Runner berganti nama menjadi Hot Java.

Pada sekitar bulan [Maret](https://id.wikipedia.org/wiki/Maret) [1995](https://id.wikipedia.org/wiki/1995), untuk pertama kali kode sumber Java versi 1.0a2 dibuka. Kesuksesan mereka diikuti dengan untuk pemberitaan pertama kali pada surat kabar *San Jose Mercury News* pada tanggal [23 Mei](https://id.wikipedia.org/wiki/23_Mei) [1995](https://id.wikipedia.org/wiki/1995).

Sayang terjadi perpecahan di antara mereka suatu hari pada pukul 04.00 di sebuah ruangan hotel [Sheraton Palace](https://id.wikipedia.org/w/index.php?title=Sheraton_Palace&action=edit&redlink=1). Tiga dari pimpinan utama proyek, [Eric Schmidt](https://id.wikipedia.org/wiki/Eric_Schmidt) dan [George Paolini](https://id.wikipedia.org/w/index.php?title=George_Paolini&action=edit&redlink=1) dari [Sun Microsystems](https://id.wikipedia.org/wiki/Sun_Microsystems) bersama [Marc Andreessen](https://id.wikipedia.org/w/index.php?title=Marc_Andreessen&action=edit&redlink=1), membentuk [Netscape](https://id.wikipedia.org/wiki/Netscape).

Nama Oak, diambil dari pohon oak yang tumbuh di depan jendela ruangan kerja "Bapak Java", [James Gosling](https://id.wikipedia.org/wiki/James_Gosling). Nama Oak ini tidak dipakai untuk versi release Java karena sebuah perangkat lunak lain sudah terdaftar dengan merek dagang tersebut, sehingga diambil nama penggantinya menjadi "Java". Nama ini diambil dari kopi murni yang digiling langsung dari biji (kopi tubruk) kesukaan Gosling. Konon kopi ini berasal dari Pulau [Jawa](https://id.wikipedia.org/wiki/Jawa). Jadi nama bahasa pemrograman Java tidak lain berasal dari kata Jawa (bahasa Inggris untuk Jawa adalah Java).

* Versi Awal

Versi awal Java pada tahun [1996](https://id.wikipedia.org/wiki/1996) sudah merupakan versi release sehingga dinamakan Java Versi 1.0. Java versi ini menyertakan banyak paket standar awal yang terus dikembangkan pada versi selanjutnya:

* java.lang: Peruntukan kelas elemen-elemen dasar.
* java.io: Peruntukan kelas *input* dan *output*, termasuk penggunaan berkas.
* java.util: Peruntukan kelas pelengkap seperti kelas struktur data dan kelas kelas penanggalan.
* java.net: Peruntukan kelas [TCP/IP](https://id.wikipedia.org/wiki/TCP/IP), yang memungkinkan berkomunikasi dengan komputer lain menggunakan jaringan TCP/IP.
* java.awt: Kelas dasar untuk aplikasi antarmuka dengan pengguna (GUI)
* java.applet: Kelas dasar aplikasi antar muka untuk diterapkan pada penjelajah web.
* **Sejarah NeatBeans**

Awal kemunculan aplikasi NetBeans telah ada sejak tahun [1997](https://id.wikipedia.org/wiki/1997) yaitu sebagai sebuah proyek kuliah. Pada tahun tersebut, suatu perusahaan dibangun oleh Roman Staněk di sekitar proyek kuliah tersebut lalu perusahaan tersebut memulai memproduksi versi NetBeans IDE komersial hingga akhirnya dibeli oleh [Sun Microsystems](https://id.wikipedia.org/wiki/Sun_Microsystems) pada tahun 1999 lalu menjadikan NetBeans IDE sebagai serambi bersifat sumber terbuka pada bulan Juni 1999**.**

* **Fitur**

* Smart code completion : mengusulkan nama variable dari suatu tipe, melengkapi keyword, dan mengusulkan tipe parameter dari method
* Menggunakan code generator : dengan menggunakan fitur ini kita dapat meng-generate constructor, setter and getter method, dll.
* Error stripe : fitur yang menandai baris yang error dengan menghiglight merah.
* Bookmarking : fitur yang digunakan untuk menandai baris yang suatu saat hendak kita modifikasi
* go to commands : fitur yang digunakan untuk jump ke deklarasi variable, source code atau file yang ada pada project yang sama

**2.1.2 A. Tipe Data Sederhana**

**1. Integer(Bilangan Bulat)**

|  |  |  |
| --- | --- | --- |
| **Tipe Data** | **Ukuran (bit)** | **Range** |
| *Byte* | 8 | -128 s.d. 127 |
| *Short* | 16 | -32768 s.d. 32767 |
| *Int* | 32 | -2147483648 s.d. 2147483647 |
| *Long* | 64 | -9223372036854775808  s.d. 9223372036854775807 |

Tipe data yang masuk menjadi bagian ini adalah byte, short, int dan long. Semua tipe data ini bersifat       *Signed,*yaitu bisa mempresentasikan nilai positif dan negatif. Tidak seperti tipe data lainnya, Java tidak mendukung tipe data *unsigned* yang hanya bisa mempresentasikan nilai postif. Untuk jelasnya akan dijelaskan oleh tabel dan penjelasan di bawah ini :

* 1. **Floating-Point (Bilangan Pecahan)**

Tipe *floating-point* digunakan untuk merepresentasikan nilai-nilai yang mengandung pecahan atau angka decimal di belakang koma, seperti 3.1416,5.25, dan sebagainya. Bilangan semacam ini disebut sebagai bilangan riil. Dalam Java tipe ini dibedakan menjadi dua jenis, yaitu *float*, dan *double*. Untuk jelasnya akan dijelaskan oleh tabel dan penjelasan di bawah ini :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tipe** | **Ukuran** | | **Range** | **Presisi (jumlah digit)** |
| **Bytes** | **Bit** |
| Float | 4 | 32 | +/- 3.4 x 1038 | 6-7 |
| Double | 8 | 64 | +/- 1.8 x 10308 | 15 |

**Float**  
Tipe ini digunakan untuk menandakan nilai–nilai yang mengandung presisi atau ketelitan tunggal (single-precision) yang menggunakan ruang penyimpanan 32-bit. Presisi tunggal biasanya lebih cepat untuk processor-processor tertentu dan memakan ruang penyimpanan setengah kali lebih sedikit dibandingkan presisi ganda (double precision). Permasalahan yang timbul dari pemakaian tipe float untuk nilai-nilai yang terlalu kecil atau justru terlalu besar, karena nilai yang dihasilkan akan menjadi tidak akurat. Contoh penggunaan variabel :

**Contoh**:

float suhu;

**Double**  
Tipe ini mengandung tingkat ketelitian ganda atau presisi ganda (double precision) dan menggunakan ruang penyimpanan 64-bit untuk menyimpan nilai. Tipe double tentu lebih cepat untuk melakukan perhitungan-perhitungan matematis daripad tipe float. Untuk perhitungan yang bersifat bilangan riil dan menghasilkan hasil yang lebih akurat, maka lebih baik menggunakan tipe double.

**Contoh**:

class KelilingLingkaran {

public static void main (String[] args) {

double pi = 3.1416;

double r = 2.12;

double keliling;

keliling = 2\*pi\*r;

System.out.println(“Keliling Lingkaran = ”+ keliling);

}

}

**Char**

Tipe data char merupakan tipe untuk menyatakan sebuah karakter.  Java menggunakan karakter *Unicode* untuk merepresentasikan semua karakter yang ada . *Unicode* ialah sekumpulan  karakter yang terdapat pada semua bahasa, seperti bahasa Latin, Arab, Yunani dan lain-lainnya. Karena bahasa Java dirancang untuk dapat diterapkan di berbagai macam *platform*, maka Java menggunakan karakter *Unicode* yang membutuhkan ukuran 16-bit. Untuk karakter-karakter yang tidak dapat diketikkan secara langsung melalui keyboard, java menyediakan beberapa *escape sequence* (pasangan karakter yang dianggap sebagai karakter tunggal). *Escape sequence*tidak dianggap sebagai *String*, melainkan tetap sebagai tipe karakter khusus. Di bawah ini akan dijelaskan beberapa contoh tentang escape sequence.

|  |  |
| --- | --- |
| ***Escape Sequence*** | **Keterangan** |
| \ddd | Karakter octal (ddd) |
| \uxxxx | Karakter Unicode heksadecimal (xxxx) |
| \’ | Petik tunggal |
| \’’ | Petik ganda |
| \\ | *Backslash* |
| \r | *Carriage return* |
| \n | Baris baru (*line feed*) |
| \f | *Form feed* |
| \t | *Tab* |
| \b | *Backspace* |

**Contoh:**

class ContohKarakter {

public static void main (String[] args) {

char ch = 65;// 65 merupakan kode untuk karakter A;

System.out.println(“ch1=”+ch);

ch++; //increment(penaikan nilai sebesar 1)

System.out.println(“ch2 = ”+ ch);

}

}

* 1. **Boolean**

Tipe *boolean* adalah tipe data yang digunakan untuk menampung nilai logika, yaitu nilai yang hanya memiliki dua buah kemungkinan (benar atau salah). Tipe ini ditandai dengan kata kunci *Boolean*. Dalam bahasa Java, nilai benar dipresentasikan dengan kata kunci *true* dan nilai salah dengan kata kunci *false*. Contoh :

class ContohBolean {

public static void main (String[] args) {

boolean a = true;

if (a) {

System.out.println(“Perintah dilaksanakan ”);

}//negasi dari a

If (!a) {

System.out.println(“Perintah tidak dilaksanakan ”);

}

}

}

* 1. **Array**

Array adalah sebuah variabel yang bisa menyimpan banyak data dalam satu variabel.Array menggunakan indeks untuk memudahkan akses terhadap data yang disimpannya., Indeks array selalu dimulai dari 0, dan perlu diketahui juga, indeks tidak selalu dalam bentuk angka. Bisa juga karakter atau teks.

Contoh :

Cara membuat array kosong:

// cara pertama

String[] nama;

// cara kedua

String nama[];

// cara ketiga dengan kata kunci new

String[] nama = new String[5];

Contoh latihan:

import java.util.Scanner;

public class Buah {

public static void main(String[] args) {

// membuat array buah-buahan

String[] buah = new String[5];

// membuat scanner

Scanner scan = new Scanner(System.in);

// mengisi data ke array

for( int i = 0; i < buah.length; i++ ){

System.out.print("Buah ke-" + i + ": ");

buah[i] = scan.nextLine();

}

System.out.println("---------------------------");

// menampilkan semua isi array

for( String b : buah ){

System.out.println(b);

}

}

}

* 1. **OOP (*Object Oriented Programming*)**

OOP (*Object Oriented Programming*) atau dalam bahasa indonesia dikenal dengan pemrograman berorientasikan objek (PBO) merupakan sebuah paradigma atau teknik pemrograman yang berorientesikan Objek. Pada OOP, Fungsi dan variabel **dibungkus** dalam sebuah **objek** atau *class*yang dapat saling  brinteraksi, sehingga membentuk sebuah program. Variabel dalam objek akan menyimpan data dari objek. Sedangkan fungsi akan menentukan operasinya.

**Class dan Object**

Class adalah rancangan atau *blue print* dari sebuah objek.

Sedangkan objek dalam pemrograman adalah sebuah variabel yang merupakan *instance* dari *Class*.

*Instance* bisa diartikan sebagai wujud dari *class*.

*Class* berisi definisi variabel dan fungsi yang menggambarkan sebuah objek.

Dalam OOP:

* **Variabel** disebut **atribut** atau **properti**;
* **Fungsi** disebut **method**.

Contoh *Class*:

class NamaClass {

String atribut1;

String atribut2;

void namaMethod(){ ... }

void namaMethodLain(){ ... }

}

Lalu, Kita biasanya membuat objek *(instance)* seperti ini:

NamaClass namaObj = new NamaClass();

Setelah membuat objek, kita bisa mengakses atribut dan method dari objek tersebut.

Contoh:

namaObj.namaMethod();

namaObj.atribut1;

* 1. **Looping (Perulangan)**

### Perulangan adalah melakukan perintah yang ada di dalam blok perulangan tersebut secara berulang - ulang sesuai dengan nilai yang ditentukan atau sampai mencapai sebuah batas tertentu dari sebuah perulangan tersebut. Sekarang kita akan membuat contoh programnya di java. While

Perulangan while bekerja dengan cara apa bila kondisi while itu terpenuhi atau bernilai true maka perulangan tersebut akan terus dilakukan sapai bernilai false.

Contoh :

package looping;

/\*\*

\*

\* @author Jin Toples

\*/

public class Looping {

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) {

int i=1;

while (i <= 5){

System.out.println(i);

i++;

}

}

}

Perhatikan script di atas ada "i++" itu adalah INCEREMENT ia akan menambah 1 nilai i secara terus menerus. ada juga yang di sebut DECREMENT, contoh : i-- maka ia akan mengurangi 1 nilai i tersebut secara terus menerus. Decrement di atas digunakan untuk menambahkan nilai "i" agar mencapai nilai "5" dan kemudian keluar perulangan. jika kita tidak memberikan decrement maka perulangan tersebut tidak akan berheti - henti ( Infinity looping ).

### Do...While

Do-while seperti while tetapi jika do-while minimal melakukan satu kali pekerjaan yang ada di dalam blok do-while tersebut. do-while akan mengulang terus sampai while bernilai flase.

Contoh :

package looping;

/\*\*

\*

\* @author Jin Toples

\*/

public class Looping {

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) {

int i=1;

do {

System.out.println(i);

i++;

}while(i>=5);

}

}

Jika anda coba script di atas maka akan mengasilkan "1" karna meskipun while bernilai false ia akan tetap melakukan pekerjaan satu kali, karna pengecekan berada di bagian bawah blok program.

### For

For adalah perulangan yang jumlah perulangannya sudah ditentukan sebelumnya, dengan kata lain perulangan blok dalam for sudah ditetukan sebelumnya.

Contoh :

package looping;

/\*\*

\*

\* @author Jin Toples

\*/

public class Looping {

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) {

for(int i=1; i<=5; i++){

System.out.println(i);

}

}

}

Penulisan for lebih singkat kara definisi variabel bisa dilakukan di dalam for tersebut. sehingga ia lebih sedikit dalam script yang digunakan dibandingkan dengan while dan do-while

**2.1.3 Definisi**

1. **Definisi Pengetahuan**

Maksud dari pengetahuan (knowledge) adalah sesuatu yang hadir dan terwujud dalam jiwa dan pikiran seseorang dikarenakan adanya reaksi, persentuhan, dan hubungan dengan lingkungan dan alam sekitarnya. Pengetahuan ini meliputi emosi, tradisi, keterampilan, informasi, akidah, dan pikiran-pikiran. Dalam komunikasi keseharian. Ketika mengamati atau menilai suatu perkara, kita biasanya menggunakan kalimat-kalimat seperti, saya mengetahuinya, saya memahaminya, saya mengenal, meyakini dan mempercayainya. Berdasarkan realitas ini, bisa dikatakan bahwa pengetahuan itu memiliki derajat dan tingkatan. Disamping itu, bisa jadi hal tersebut bagi seseorang adalah pengetahuan, sementara bagi yang lainnya bukan pengetahuan. Terkadang seseorang mengakui bahwa sesuatu itu diketahuinya dan mengenal keadaannya dengan baik, namun, pada hakikatnya, ia salah memahaminya dan ketika ia berhadapan dengan seseorang yang sungguh-sungguh mengetahui realitas tersebut, barulah ia menyadari bahwa ia benar-benar tidak memahami permasalahan tersebut sebagaimana adanya.

Pengetahuan adalah suatu keadaan yang hadir dikarenakan persentuhan kita dengan suatu perkara. Keluasan dan kedalaman kehadiran kondisi-kondisi ini dalam pikiran dan jiwa kita sangat bergantung pada sejauh mana reaksi, pertemuan, persentuhan, dan hubungan kita dengan objek-objek eksternal. alhasil, makrifat dan pengetahuan ialah suatu keyakinan yang kita miliki yang hadir dalam syarat-syarat tertentu dan terwujud karena terbentuknya hubungan-hubungan khusus antara subjek (yang mengetahui) dan objek (yang diketahui) dimana hubungan ini sama sekali kita tidak ragukan. John Dewey menyamakan antara hakikat itu sendiri dan pengetahuan dan beranggapan bahwa pengetahuan itu merupakan hasil dan capaian dari suatu penelitian dan observasi. Menurutnya, pengetahuan seseorang terbentuk dari hubungan dan jalinan ia dengan realitas-realitas yang tetap dan yang senantiasa berubah.

1. **Definisi umum**

Mengenai seluruhnya atau semuanya, secara menyeluruh, tidak menyangkut yang khusus (tertentu) saja.

1. **Definisi Software**

Software adalah kumpulan instruksi  yang berfungsi untuk menjalankan suatu perintah, seperti memberikan informasi tentang hardware, menentukan fungsi hardware, dan menjalankan sistem. Agar komputer dapat membaca, mengingat, membuat keputusan (membandingkan), menghitung, menyortir, dan menghasilkan keluaran berupa informasi dalam monitor atau cetakan, komputer harus dapat membaca dan memasukkan program ke dalam memori utamanya. Program adalah instruksi dalam bahasa mesin atau yang dapat dibaca oleh komputer yang dirancang untuk tujuan tertentu sehingga kalau operator menjalankan komputer dan memijat tombol tertentu (misalnya untuk memroses data akuntansi) disebut dengan program aplikasi (aplication program). Pengertian perangkat lunak menunjuk pada program dan alat bantu lain yang bersifat menambah kemampuan komputer sebagai alat untuk melaksanakan tugas atau operasi tertentu. Program aplikasi dapat dibuat secara khusus untuk memenuhi kebutuhan khusus pula (tailor-made) atau berupa paket yang mempunyai aplikasi umum.

1. **Program aplikasi tes pengetahuan umum (game edukasi)**

* Menurut Papert (1993) mengemukakan "notes that software games teach children that some forms of learning are fast-paced, immensely compelling and rewarding where as by comparison school strikes many young people as slow and boring".( bahwa software game untuk mengajar anak-anak baik untuk kecepatan pemahaman dan sangat menarik serta bermanfaat, ini merupakan sebagai perbandingan cara belajar disekolah yang lama dan membosankan)
* Menurut Boyle (1997) mengemukakan "points out that games can produce engagement and delight in learning; they thus offer a powerful format for educational environments. Moreover, there are studies that have shown that the use of carefully selected computer games may improve thinking". (Menunjukkan bahwa game dapat menghasilkan keterlibatan dan senang belajar; dengan demikian menawarkan format yang kuat untuk pendidikan lingkungan. Selain itu, ada studi yang telah menunjukkan bahwa penggunaan permainan komputer yang dipilih dengan cermat mungkin meningkatkan cara berpikir)
* Menurut Wahono (ilmukomputer.com.2007) mengemukakan game merupakan aktifitas terstruktur atau semi terstruktur yang biasanya bertujuan untuk hiburan dan kadang dapat digunakan sebagai sarana pendidikan.
* Menurut Foreman (2009) mengemukakan bahwa game merupakan *potential learning environment,*bermain game merupakan sebuah literatur baru dalam pendidikan.
* Menurut Buckingham dan Scalon (2002) mengemukakan dalam terjemahan bahwa sebenarnya tanpa disadari game dapat mengajarkan banyak keterampilan dan game dapat dijadikan sebagai salah satu alternatif pendidikan.
* Menurut Edward (2009) mengemukakan dalam terjemahan bahwa game merupakan sebuah *tools/*alat yang efektif untuk mengajar karena mengandung prinsip-prinsip pembelajaran dan teknik instruksional yang efektif digunakan dalam penguatan pada level-level yang sulit.

1. **Flowchart**

Flowchart merupakan gambar atau bagan yang memperlihatkan urutan dan hubungan antar proses beserta instruksinya. Gambaran ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan hubungan antar proses digambarkan dengan garis penghubung.

Flowchart ini merupakan langkah awal pembuatan program. Dengan adanya flowchart urutan poses kegiatan menjadi lebih jelas. Jika ada penambahan proses maka dapat dilakukan lebih mudah. Setelah flowchart selesai disusun, selanjutnya pemrogram (programmer) menerjemahkannya ke bentuk program dengan bahasa pemrograman.

* **Simbol-simbol flowchart**

Flowchart disusun dengan simbol-simbol. Simbol ini dipakai sebagai alat bantu menggambarkan proses di dalam program. Simbol-simbol yang dipakai antara lain :

**[](http://2.bp.blogspot.com/-e5q3DP1EALI/UIFe21JECZI/AAAAAAAAABk/GD3WOzewHK8/s1600/flowdirectionsymbol.jpg)**

**Flow Direction symbol**

Yaitu simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga connecting line.

**[](http://4.bp.blogspot.com/-Tm5F5lr87GA/UIFhcFMmWtI/AAAAAAAAACQ/oBVnNwVm4Qc/s1600/terminatorsymbol.jpg)**

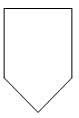
**Terminator Symbol**

Yaitu simbol untuk permulaan (start) atau akhir (stop) dari suatu kegiatan

**[](http://4.bp.blogspot.com/-X__zp7Th5vM/UIFh9qpAQqI/AAAAAAAAACY/KSvlHUokRCw/s1600/connectorsymbol.jpg)**

**Connector Symbol**

Yaitu simbol untuk keluar – masuk atau penyambungan proses dalam lembar / halaman yang sama.

**[](http://2.bp.blogspot.com/-7gu6ySccKrE/UIFiRsUoxBI/AAAAAAAAACg/MCZTpTSMXWg/s1600/connectorsymbol2.jpg)**

**Connector Symbol**

Yaitu simbol untuk keluar – masuk atau penyambungan proses pada lembar / halaman yang berbeda.

**[http://2.bp.blogspot.com/-XtFf-twI3Jo/UIFimY4WkZI/AAAAAAAAACo/UUw-2TgSYbk/s1600/processingsymbol.jpg](http://2.bp.blogspot.com/-XtFf-twI3Jo/UIFimY4WkZI/AAAAAAAAACo/UUw-2TgSYbk/s1600/processingsymbol.jpg)**

**Processing Symbol**

Simbol yang menunjukkan pengolahan yang dilakukan oleh komputer

**[http://4.bp.blogspot.com/-wgF3yHj1vCg/UIFi8oFbHWI/AAAAAAAAACw/wCJxQK_0cko/s1600/simbolmanualoperation.jpg](http://4.bp.blogspot.com/-wgF3yHj1vCg/UIFi8oFbHWI/AAAAAAAAACw/wCJxQK_0cko/s1600/simbolmanualoperation.jpg)**

**Simbol Manual Operation**

Simbol yang menunjukkan pengolahan yang tidak dilakukan oleh komputer

**[http://2.bp.blogspot.com/-i8hxVNiw-Hk/UIFjUJ9gtdI/AAAAAAAAAC4/4R8DqJYV6RY/s1600/simboldecision.jpg](http://2.bp.blogspot.com/-i8hxVNiw-Hk/UIFjUJ9gtdI/AAAAAAAAAC4/4R8DqJYV6RY/s1600/simboldecision.jpg)**

**Simbol Decision**

Simbol pemilihan proses berdasarkan kondisi yang ada.

**[http://3.bp.blogspot.com/-_sSAm3mfng4/UIFjkCUfrKI/AAAAAAAAADA/ZtWNjgsw2VI/s1600/simbolinput-output.jpg](http://3.bp.blogspot.com/-_sSAm3mfng4/UIFjkCUfrKI/AAAAAAAAADA/ZtWNjgsw2VI/s1600/simbolinput-output.jpg)**

**Simbol Input-Output**

Simbol yang menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya

**[](http://4.bp.blogspot.com/-_WgVglMfCvQ/UIFjzpG_h1I/AAAAAAAAADI/vPw-KrxlcKE/s1600/simbolmanualinput.jpg)**

**Simbol Manual Input**

Simbol untuk pemasukan data secara manual on-line keyboard

**[](http://3.bp.blogspot.com/-hyj2x_C0ndo/UIFkCccNieI/AAAAAAAAADQ/hWKwpqyQ4vk/s1600/simbolpreparation.jpg)**

**Simbol Preparation**

Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan di dalam storage.

**[http://3.bp.blogspot.com/-zyQVHGENvlg/UIFkS3rOG7I/AAAAAAAAADY/3RmchY1ETh0/s1600/simbolpredefineproses.jpg](http://3.bp.blogspot.com/-zyQVHGENvlg/UIFkS3rOG7I/AAAAAAAAADY/3RmchY1ETh0/s1600/simbolpredefineproses.jpg)**

**Simbol Predefine Proses**

Simbol untuk pelaksanaan suatu bagian (sub-program)/prosedure

**[](http://1.bp.blogspot.com/-6jyHJ9tspEw/UIFkzvuejsI/AAAAAAAAADg/p_88tO33Uj0/s1600/simboldisplay.jpg)**

**Simbol Display**

Simbol yang menyatakan peralatan output yang digunakan yaitu layar, plotter, printer dan sebagainya.

**[](http://1.bp.blogspot.com/-zBPCN3NdSJ8/UIFlEU62dII/AAAAAAAAADo/jqhbkmgARrs/s1600/simboldiskandon-linestorage.jpg)**

**Simbol disk and On-line Storage**

Simbol yang menyatakan input yang berasal dari disk atau disimpan ke disk.

* **Kaidah-kaidah pembuatan Flowchart**

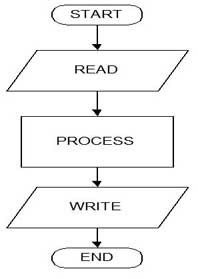
Dalam pembuatan flowchart tidak ada rumus atau patokan yang bersifat mutlak. Karena flowchart merupakan gambaran hasil pemikiran dalam menganalisa suatu masalah dengan komputer. Sehingga flowchart yang dihasilkan dapat bervariasi antara satu pemrogram dengan pemrogram lainnya.

Namun secara garis besar, setiap pengolahan selalu terdiri dari tiga bagian utama, yaitu;

1. Input berupa bahan mentah
2. Proses pengolahan
3. Output berupa bahan jadi.

Untuk pengolahan data dengan komputer, dapat dirangkum urutan dasar untuk pemecahan suatu masalah, yaitu;

* **START**: berisi instruksi untuk persiapan perlatan yang diperlukan sebelum menangani  pemecahan masalah.
* **READ:** berisi instruksi untuk membaca data dari suatu peralatan input.
* **PROCESS:** berisi kegiatan yang berkaitan dengan pemecahan persoalan sesuai dengan data yang dibaca.
* **WRITE:** berisi instruksi untuk merekam hasil kegiatan ke perlatan output.
* **END:**mengakhiri kegiatan pengolahan

**[](http://3.bp.blogspot.com/-aGRsnaR29RA/UIFllWPewAI/AAAAAAAAADw/Y5DV7rWuNmc/s1600/dasar1.jpg)**

**Gambar berikut memperlihatkan flowchart dari kegiatan dasar diatas.**

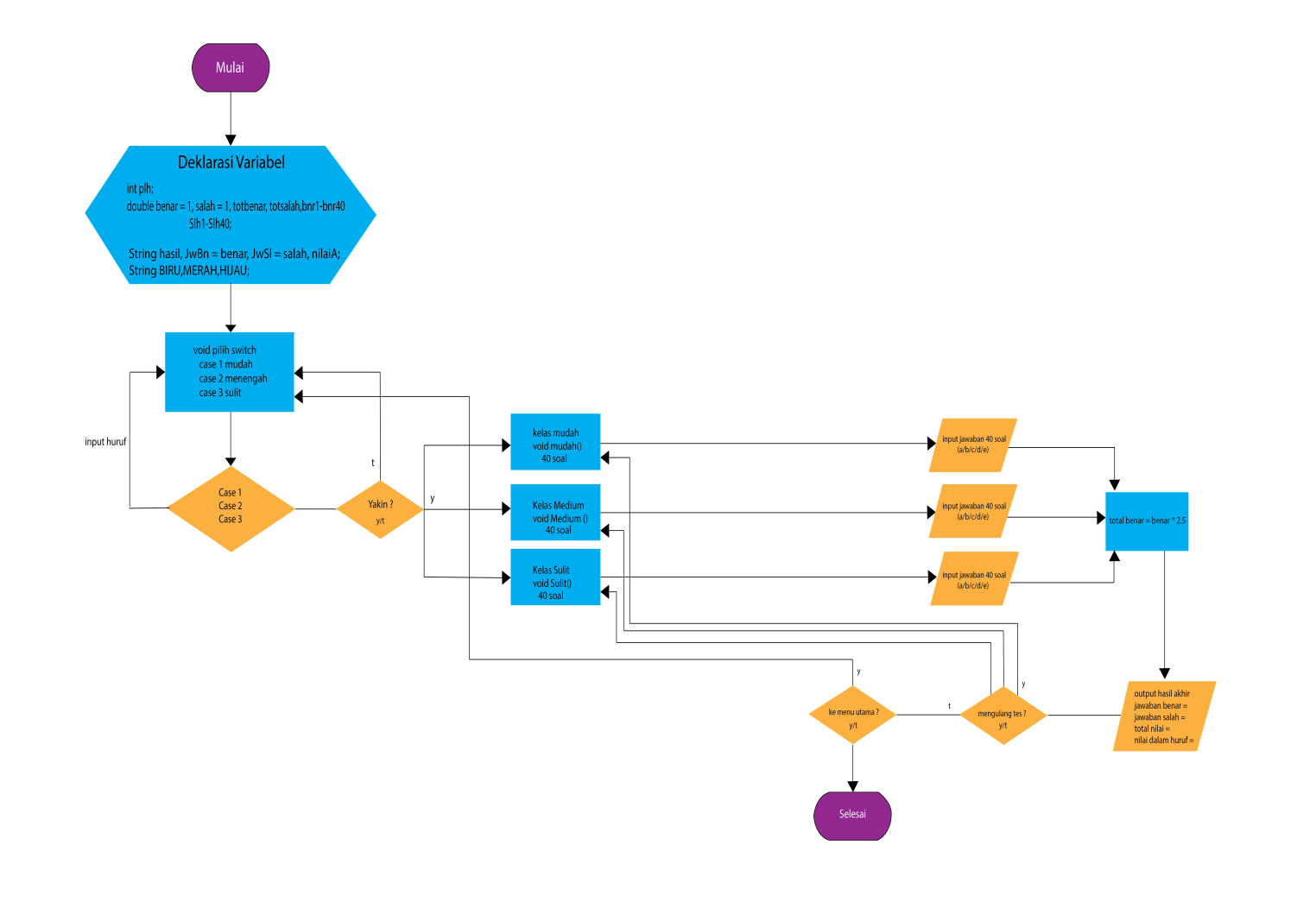
Dari gambar flowchart di atas terlihat bahwa suatu flowchart harus terdapat proses persiapan dan proses akhir. Dan yang menjadi topik dalam pembahasan ini adalah tahap proses. Karena kegiatan ini banyak mengandung variasi sesuai dengan kompleksitas masalah yang akan dipecahkan. Walaupun tidak ada kaidah-kaidah yang baku dalam penyusunan flowchart, namun ada beberapa anjuran yaitu:

* Hindari pengulangan proses yang tidak perlu dan logika yang berbelit sehingga jalannya proses menjadi singkat
* Penggambaran flowchart yang simetris dengan arah yang jelas.
* Sebuah flowchart diawali dari satu titik START dan diakhiri dengan END.

**BAB III**

**DIAGRAM ALIR**

**3.1 Flowchart**

****

**BAB IV**

**IMPLEMENTASI**

**4.1 Deskripsi Aplikasi**

Program Dibuat untuk hiburan dan ajang tes pengetahuan dan untuk mengatahui lebih tentang diri sendiri**.**

**4.2 Cara Kerja Program**

1.Program Dijalankan

2.Muncul Menu utama

3.Muncul pilihan Tes yang akan dipilih

4.User memasukkan input yang akan dipilih

5.User ditanya apakah yakin ingin meneruskan tes yang dipilih, jika ya maka akan dilanjutkan ke tes yang dipilih.

5.Program yang dipilih akan terbuka

6.User memasukkan input jawaban dari soal yang ada

7.Setelah selesai, User akan mendapat nilai dari tes yang dijalaninya

8.Muncul Nilai Tes dan Grade nya dalam bentuk Huruf

9.Muncul pertanyaan apakah ingin mengulang?

10.jika ya maka program akan mengulang kembali

11.Muncul pertanyaan apakah ingin ke menu utama?

12.jika ya maka program akan kembali ke menu utama.

13.jika tidak maka program telah selesai.

**4.2.1 Proses**

****

Terdapat 5 kelas dalam projek kami yaitu : Class =

1.Driver Soal = main kelas untuk menjalankan keseluruhan program.

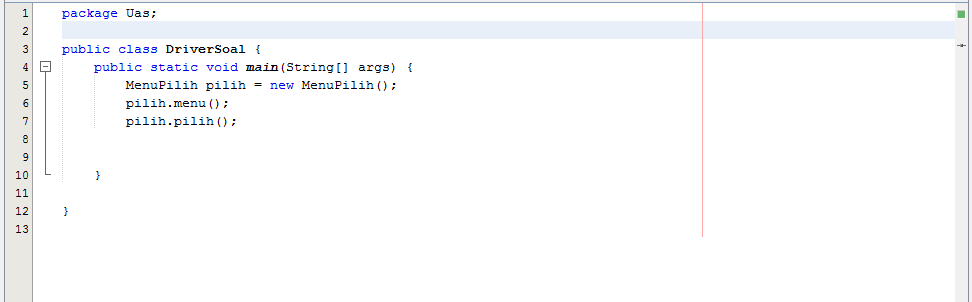
2.Menu pilih = didalamnya terdapat berbagai menu dan pilihan seperti Void

3.Mudah = Kelas yang menampung 40 soal pengetahuan umum tingkat mudah.

4.Medium = Kelas yang menampung 40 soal pengetahuan umum tingkat sedang.

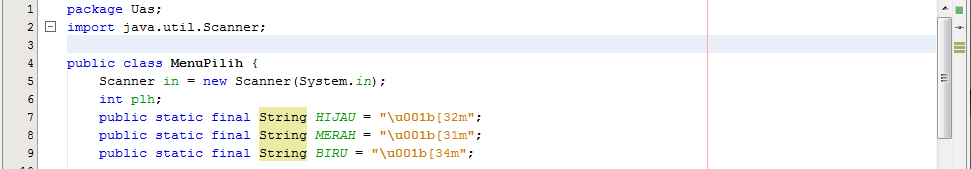
5.Sulit = Kelas yang menampung 40 soal pengetahuan umum tingkat sulit.

**1.Kelas DriverSoal**

****

Ini tampilan code yang ada di kelas DriverSoal kode ini berguna untuk memanggil void yang ada di kelas lain dan menjalankannya.

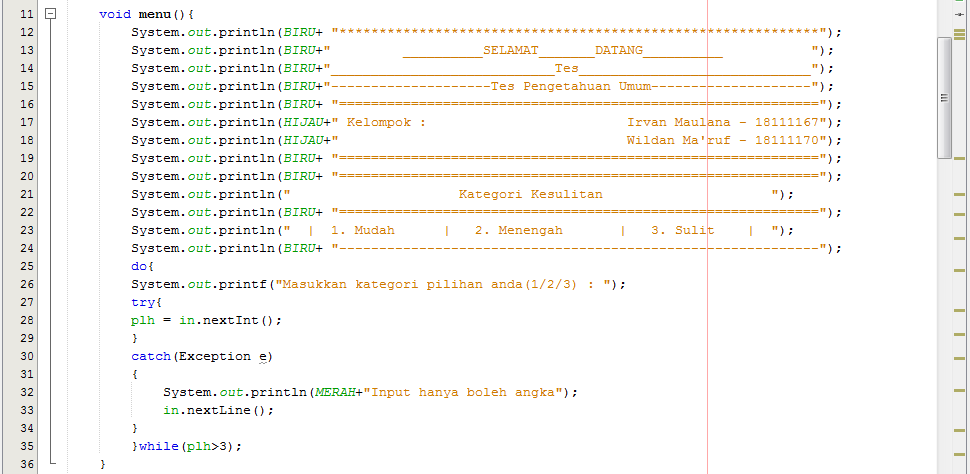
**2.Kelas MenuPilih**

****

Ini tampilan kelas MenuPilih yang menampilkan variable plh untuk ditempatkan pada void pilih()

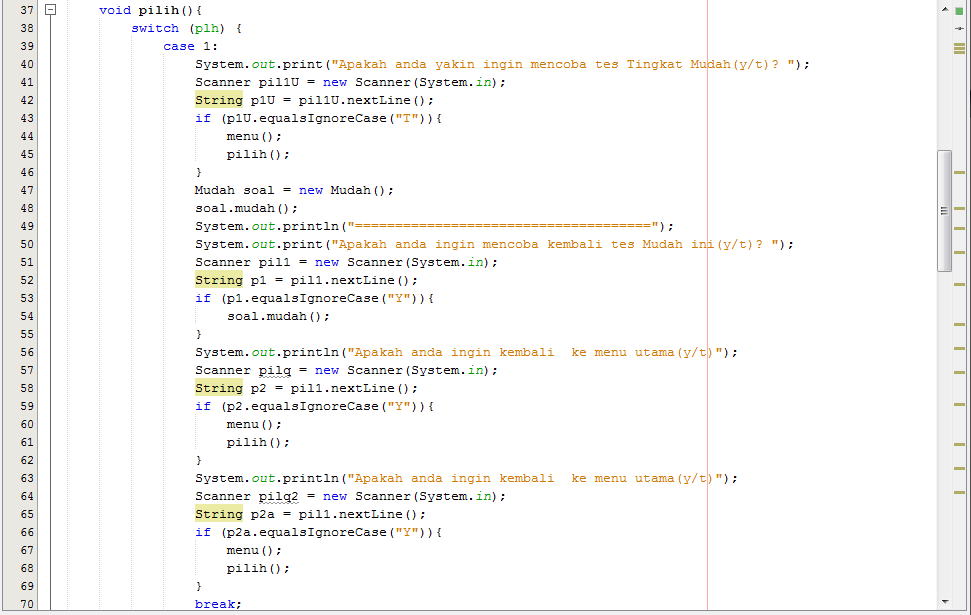
Dan beberapa kode warna untuk menghias menu dalam format variabel jadi tinggal di panggil variable nya.

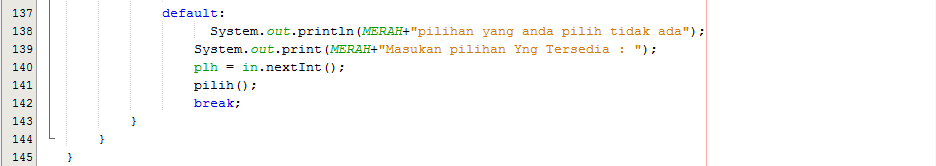
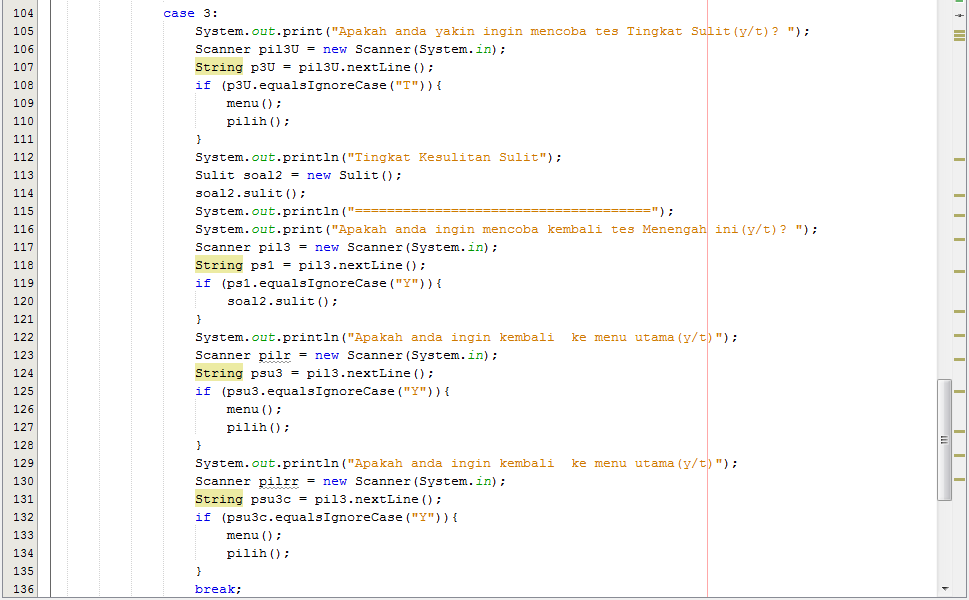
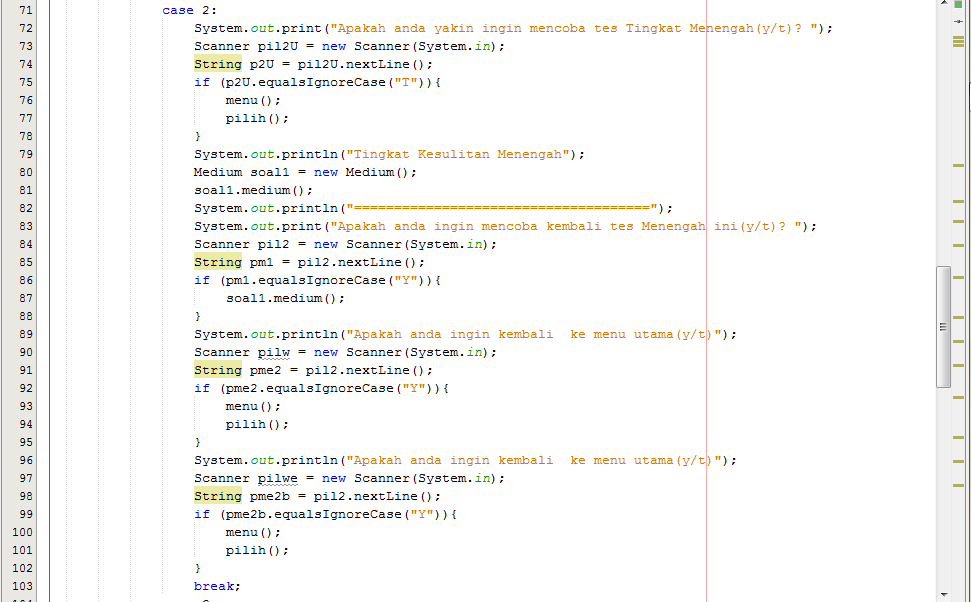
Void Menu ( )

****

Ini tampilan dari Void menu() yang terdapat di kelas MenuPilih void menu ini adalah rancangan menu utama yang akan ditampilkan.

Void Pilih ( )

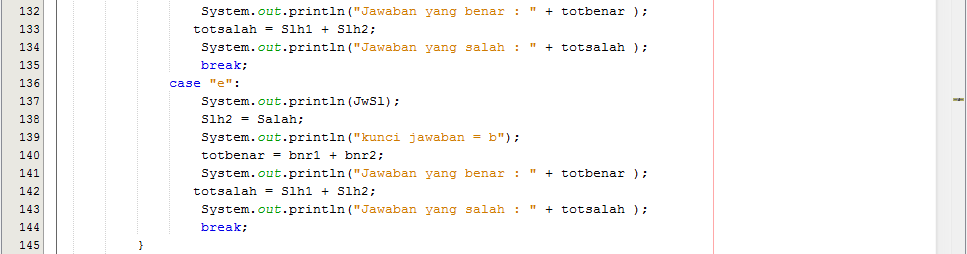
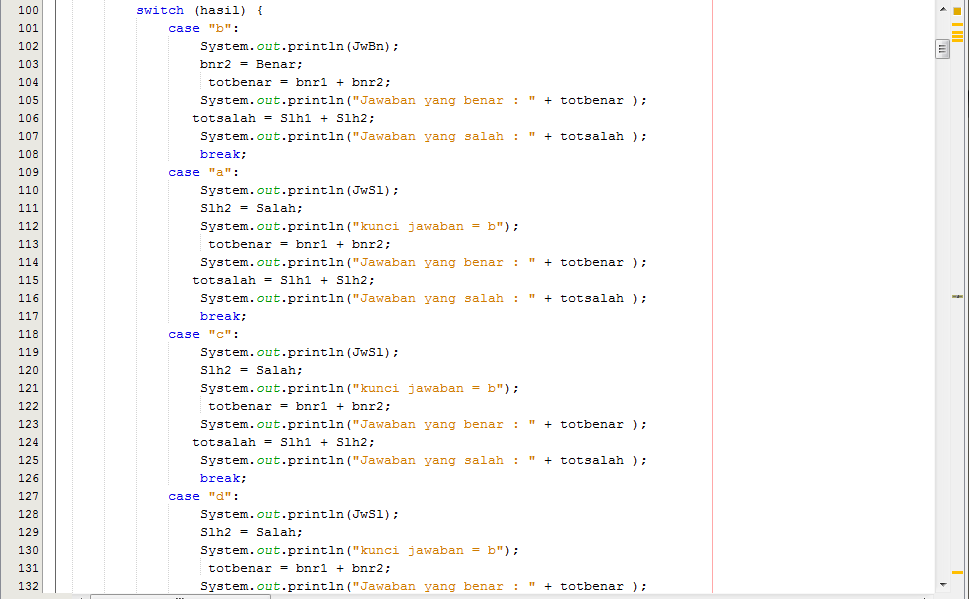
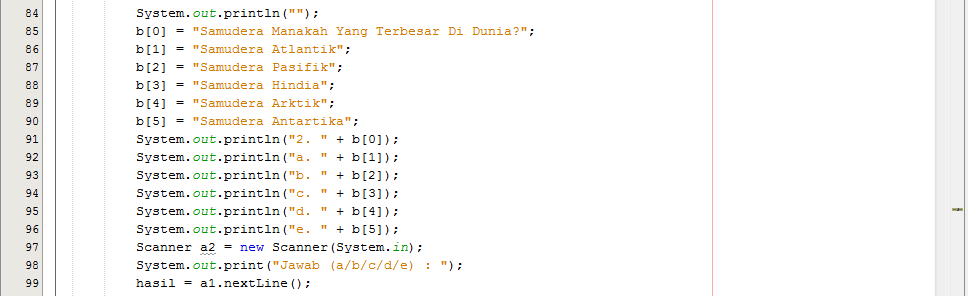
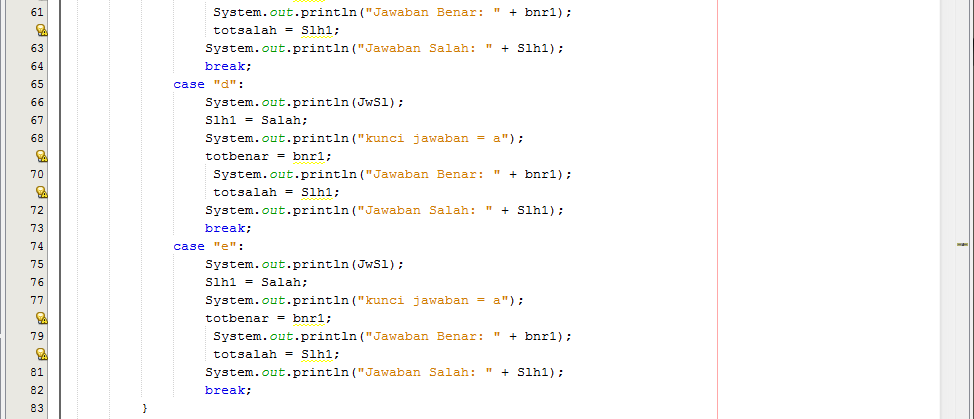
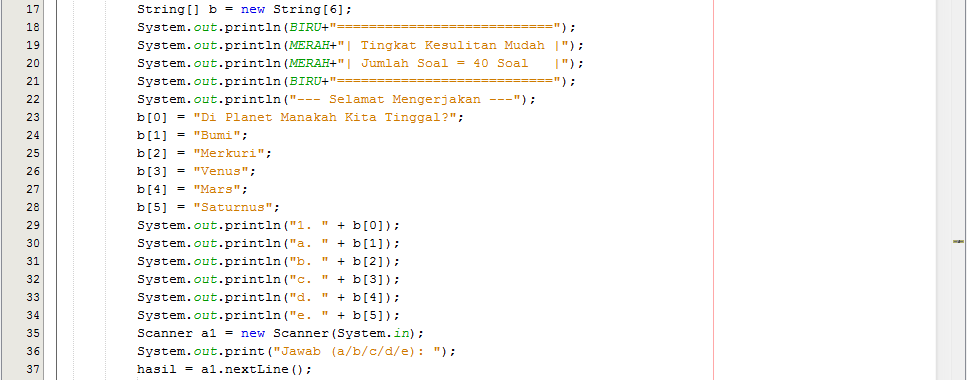
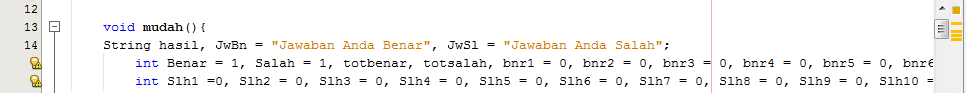
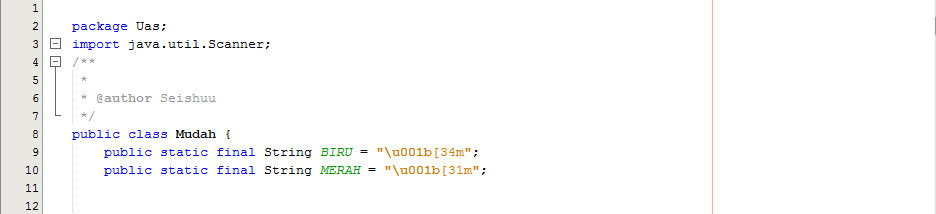


****

Ini tampilan dari Void Pilih yang berisikan Switch case dari beberapa pilihan kategori kesulitan yang ada di void menu.

Void ini berisikan 3 case yaitu case 1 = kesulitan mudah, case 2 = kesulitan menengah, case 3 = kesulitan sulit.

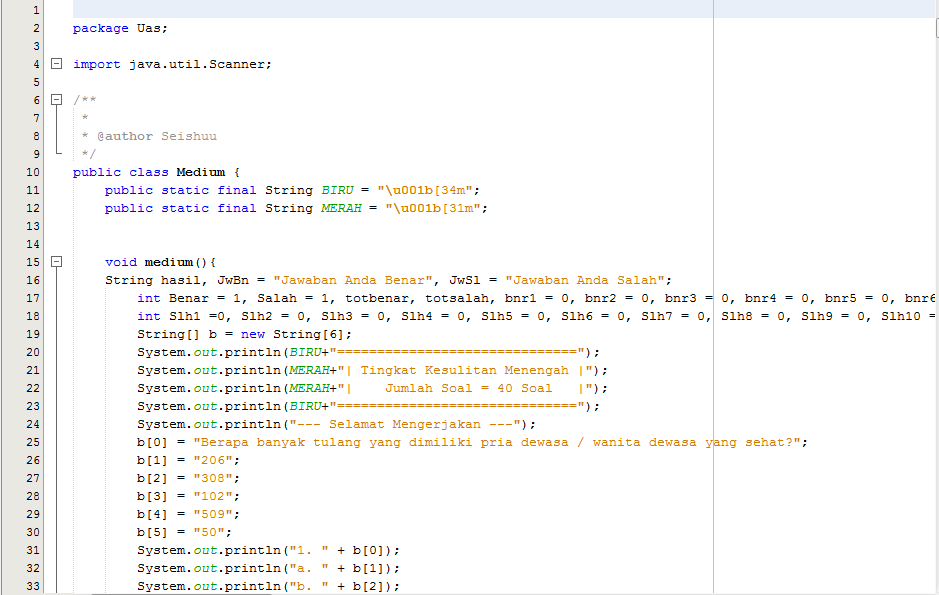
**Kelas Mudah**

****

Ini tampilan kelas mudah dari variable sampai soal 1 dan 2 tidak bisa saya screenshoot semua karena sangat banyak. Gambar berikutnya akan saya tampilkan hasil penghitungan dari program tes pengetahuan.

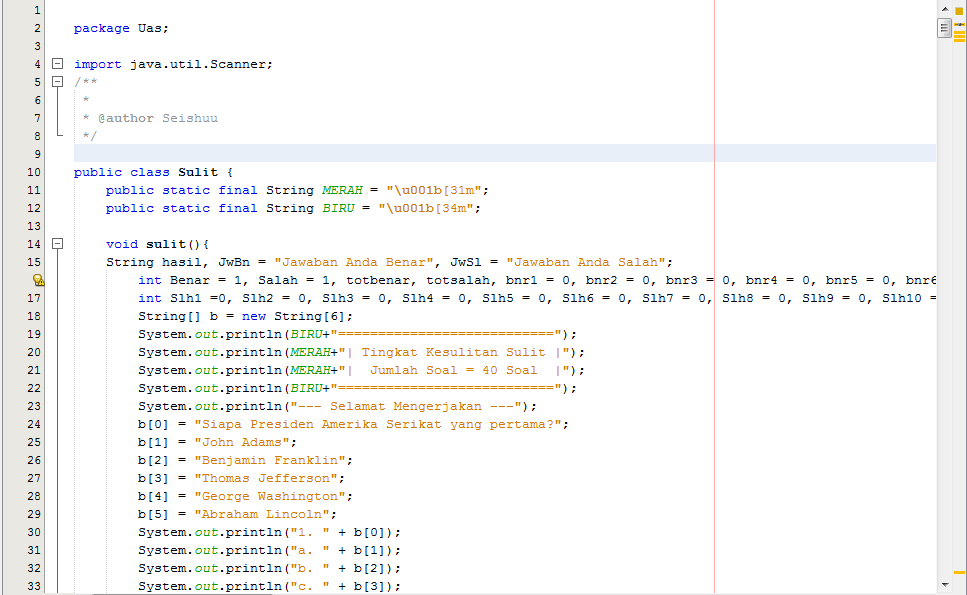


**Kelas Medium**



Ini tampilan dari kelas Medium tidak bisa saya tampilkan semua karena sangat banyak, karena sistem nya sama dengan kelas soal mudah namun hanya beda pada soal saja.

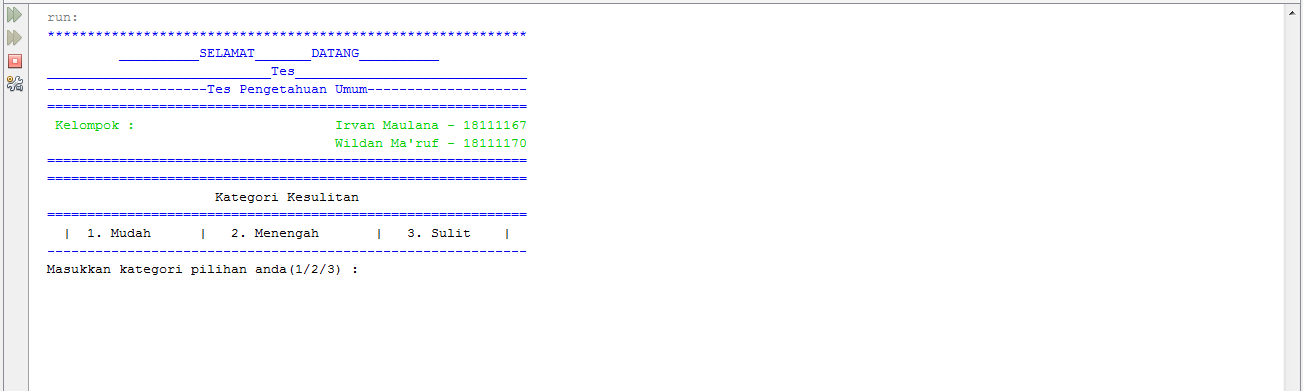
**Kelas Sulit**

****

Ini tampilan dari kelas Medium tidak bisa saya tampilkan semua karena sangat banyak, karena sistem nya sama dengan kelas soal mudah namun hanya beda pada soal saja.

**Output**

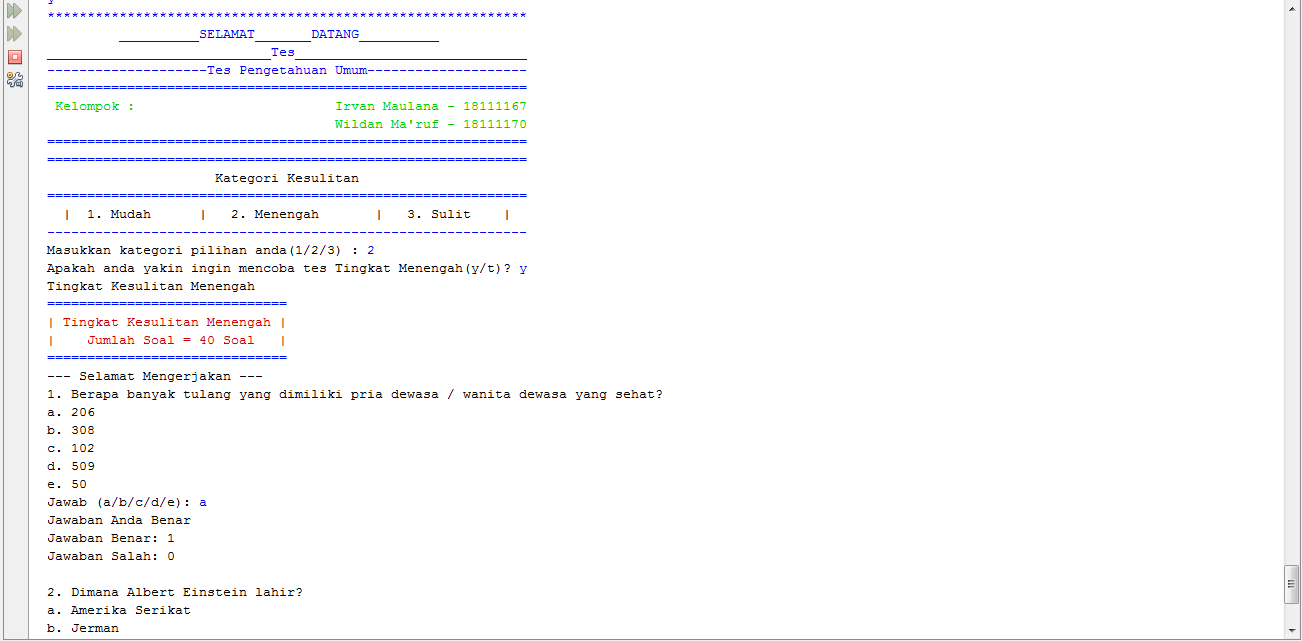
**Menu**

****

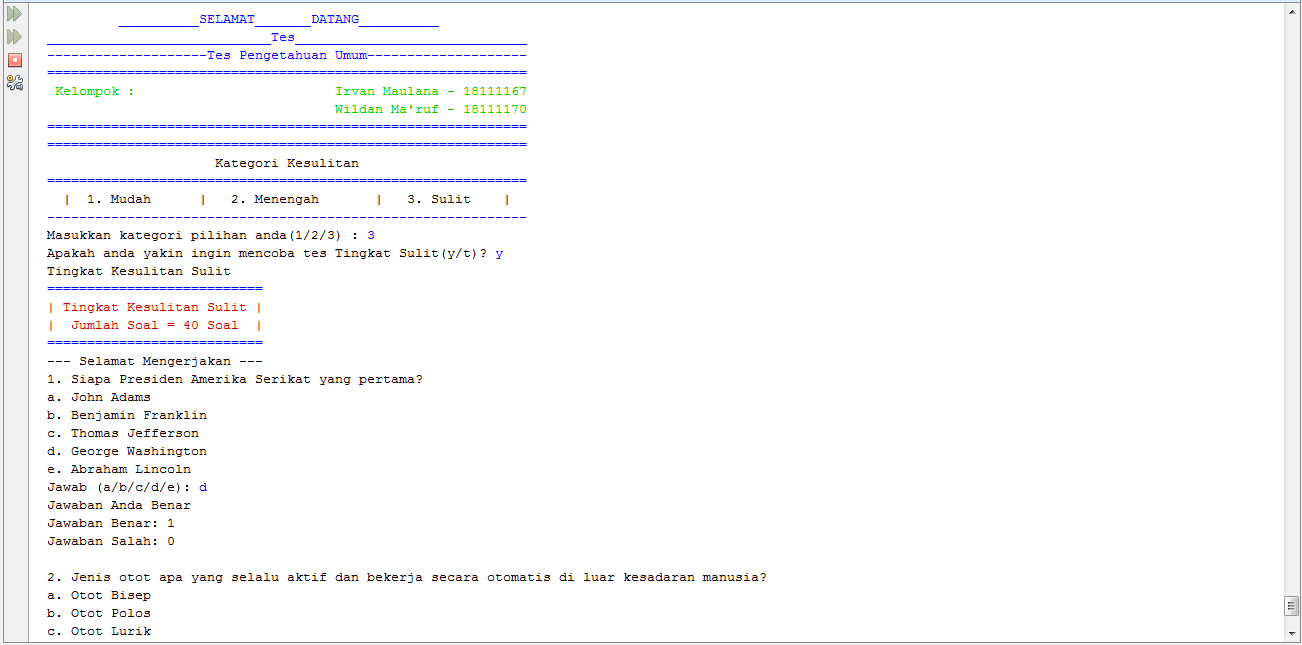
**Input kategori Mudah**

* + ****

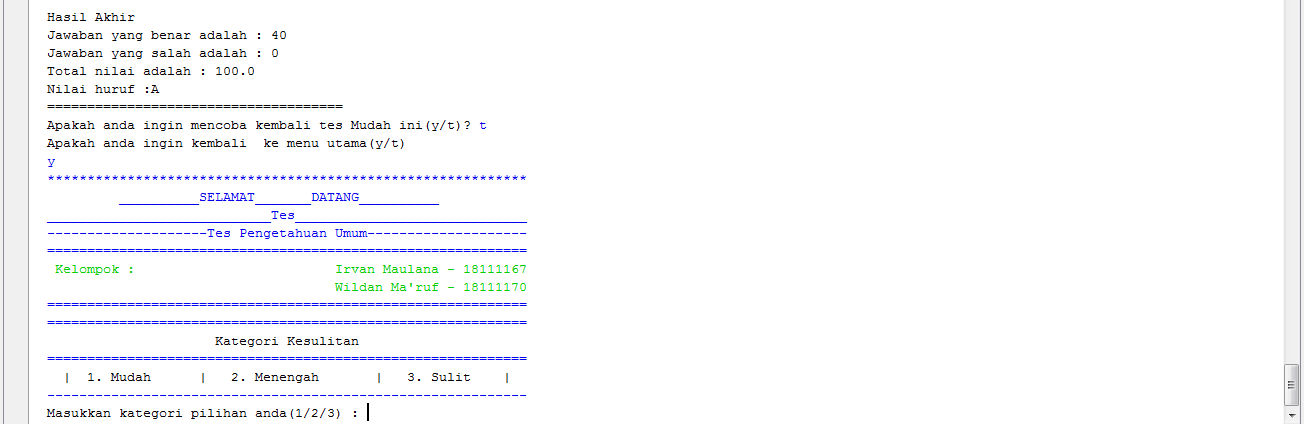
**Input kategori menengah**

****

**Input kategori Sulit**

****

**Hasil akhir**

****

**BAB V**

**PENUTUP**

* 1. **Kesimpulan**

Aplikasi pengetahuan umum dapat membantu untuk mempermudah dalam menjawab soal edukasi pengetahuan umum, lebih hemat dalam menggunakan kertas dan lebih simple untuk menjawab soal soal yang telah tersedia, karena teknologi semakin canggih kita juga bisa memanfaatkannya dengan hal yang positif.

* 1. **Saran**

Menyadari bahwa penulis masih jauh dari kata sempurna, Untuk saran bisa berisi kritik atau saran terhadap penulisan juga bisa untuk menanggapi terhadap kesimpulan dari bahasan yang telah di jelaskan. Untuk bagian terakhir adalah daftar pustaka*.*

**DAFTAR PUSTAKA**

https://id.wikipedia.org/wiki/Java

https://id.wikipedia.org/wiki/NetBeans

<http://jintoples.blogspot.com/2012/12/perulangan-pada-java.html#.XBvyQu_Mkuw>

<http://andreyanto-gunadarma.blogspot.com/2012/10/pengenalan-flowchart-flowchart.html>

<http://jaribandel.blogspot.com/2015/06/definisi-game-edukasi-menurut-para-ahli.html>