

## KOMPUTERISASI KONVERSI BILANGAN MENGUNAKAN BORLAND DELPHI 7.0

Prihastuti Harsani dan Reni Megawati

Program Studi Ilmu Komputer FMIPA UNPAK

### I. PENDAHULUAN

#### 1.1 Latar Belakang

Dalam bidang matematika, sistem bilangan merupakan salah satu bidang yang cukup penting. Sistem bilangan merupakan suatu cara untuk menuliskan dan melambangkan nilai suatu bilangan dengan basis tertentu, dilambangkan dengan  $XXX(y)$ .

Dalam elektronika dan komputer penggunaan basis yang paling mendasar adalah basis-2 dan perangkatannya (basis-8 dan basis-16). Nilai dari setiap angka pada suatu sistem bilangan dapat dikonversikan ke dalam sistem bilangan yang lain. Dalam penelitian ini sistem bilangan yang akan dikonversikan adalah bilangan desimal menjadi bilangan biner, oktal, hexadesimal dan sebaliknya.

Untuk lebih mudah mengingat dan memahami proses konversi sistem bilangan tersebut, maka dibuatlah aplikasi mengenai sistem konversi bilangan.

#### 1.2 Tujuan Penelitian

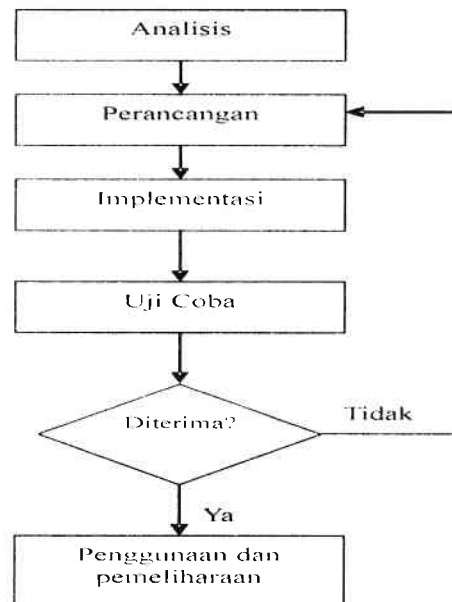
Tujuan diadakan penelitian ini adalah untuk merancang dan mengimplementasikan Komputerisasi Konversi Bilangan Menggunakan Borland Delphi 7.0.

### I. METODE PENELITIAN

Dalam metodologi penelitian ini dilakukan pendekatan sistem yaitu

### II. METODE PENELITIAN

Dalam metodologi penelitian ini dilakukan pendekatan sistem yaitu menggunakan metode SDLC (*System Development Life Cycle*). Metode SDLC merupakan suatu proses evolusioner dalam menerapkan suatu sistem atau subsistem informasi berbasis komputer (Macleod, 1996). Metode ini memiliki empat tahap dan dapat dilihat secara lengkap pada gambar 1.



Gambar 1. Diagram Proses Pembuatan Sistem

## 11. PEMBAHASAN

### 3.1. Analisis

Pada dasarnya sistem yang akan dibuat mempunyai dua teknik konversi yang berbeda, yaitu konversi bilangan dari desimal ke biner, oktal, hexadesimal dan sebaliknya. Untuk proses konversi bilangan dari desimal ke biner, oktal, hexadesimal digunakan rumus pembagian sisa. Dimana setiap bilangan desimal yang dikonversikan dibagi dengan basis bilangan konversi dan sisa pembagian tersebut merupakan hasil konversi. Sedangkan konversi bilangan sebaliknya dari biner, oktal, hexadesimal ke desimal digunakan rumus perkalian masing-masing bit dalam bilangan dengan *place value*nya.

### 3.2. Antar Muka Pengguna (User Interface)

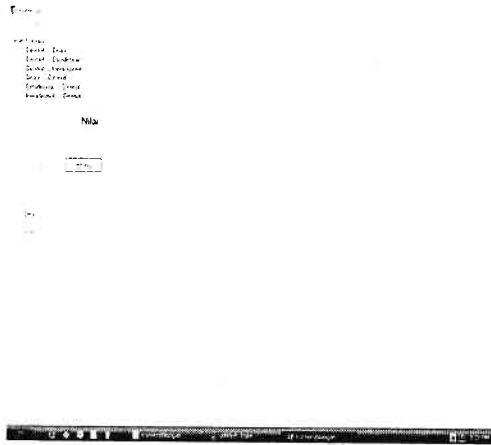
Perancangan antar muka pengguna merupakan salah satu bagian yang harus diperhatikan. Antar muka pengguna berperan penting dalam melakukan komunikasi antara pemakai yang memasukkan data dengan program yang mengatasi permasalahan. Komunikasi antara pengguna dengan sistem sebaiknya dapat dimengerti dan dibuat semudah mungkin. Antar muka pengguna dibuat dengan menggunakan Borland Delphi 7.0.

### 3.3. Form Splash

Form Splash digunakan sebagai media untuk menampilkan antar muka antara user dengan komputer, yaitu berupa tampilan identitas pembuat program, gambar logo dan judul program.

### 3.4. Form Menu Utama

Form Menu Utama digunakan sebagai media interaksi menuju sub-sub menu yang ada didalamnya. Antar muka dari form menu utama dapat dilihat pada gambar 2.



Gambar 2. Tampilan Menu Utama

Pada tampilan menu utama program aplikasi konversi desimal terdapat 2 menu utama. Setiap menu utama memiliki sub menu yang merupakan bagian dari menu utama. Dari menu *File* terdapat satu sub menu yaitu *close*. Pada bagian menu *Konversi* terdapat tujuh sub menu yang terdiri dari konversi desimal ke biner, konversi desimal ke oktal, konversi desimal ke hexadesimal, konversi biner ke desimal, konversi oktal ke desimal, konversi hexadesimal ke desimal. Untuk print hasil konversi yang telah dihitung dengan menekan tombol print. Pada form utama ini user dapat langsung menggunakan program untuk menggambarkan konversi bilangan. Inputkan bilangan yang dikonversikan kemudian klik tombol hitung dengan otomatis hasil konversi tampil pada kolom panel canvas.

kemudian klik tombol hitung dengan otomatis hasil konversi tampil pada kolom panel canvas.

### 3.5. Uji Coba Sistem

Uji coba sistem dilakukan setelah pembuatan modul-modul sistem selesai dibuat dengan percobaan pada komputer user interface. Dengan melakukan uji coba ini dapat diketahui pula kekurangan dari sistem yang telah dibuat, diantaranya apakah sistem berjalan dengan baik, apakah sistem yang dibuat sesuai dengan perancangan pada sistem yang dirancang dan apakah penanganan kesalahan berfungsi dengan baik. Pengujian yang baik bukan untuk memastikan tidak ada kesalahan yang ada pada program, melainkan untuk mencari sebanyak mungkin kesalahan yang ada pada program. Adapun kesalahan yang sering terjadi diantaranya:

1. Kesalahan bahasa atau kesalahan penulisan sintak (*syntax error*).
2. Kesalahan sewaktu proses (*run time error*) adalah kesalahan yang terjadi sewaktu mengeksekusi program.
3. Kesalahan logika (*Logical error*) adalah kesalahan dari logika program yang dibuat.

Berikut ini adalah beberapa pengujian yang dilakukan terhadap sistem yang telah dibuat.

#### 3.5.1. Uji Coba Struktural

Uji coba struktural adalah uji coba yang dilakukan pada saat pembuatan Aplikasi Konversi Desimal dan memastikan kinerja dari sistem yang dibuat. Uji coba ini dilakukan dengan cara menjalankan setiap form atau menu yang telah dirancang. Jika terjadi kesalahan atau tidak berfungsi, maka proses akan kembali ketahap implementasi. Hal ini

akan berulang hingga didapat hasil yang diinginkan. Hasil uji coba struktural ditampilkan pada tabel 1.

Tabel 1. Uji Coba Struktural

Halaman	Sub Menu	Hasil
Form Splash	-	Sesuai
Menu Utama	Konversi Desimal ke Biner	Sesuai
	Konversi Desimal ke Oktal	Sesuai
	Konversi Desimal ke Hexa	Sesuai
	Konversi Biner ke Desimal	Sesuai
	Konversi Oktal ke Desimal	Sesuai
	Konversi Hexa ke Desimal	Sesuai
	Close	Sesuai

Dari tabel diatas dapat diketahui bahwa pada saat program dijalankan maka yang akan pertama kali ditampilkan adalah Form Splash. Selanjutnya dijalankan Menu Utama yang terdiri dari sub menu Konversi yang berfungsi untuk menampilkan arah konversi, Clear Canvas dan Close.

#### 3.5.2. Uji Coba Fungsional

Didalam uji coba ini dilakukan pengujian terhadap form menu utama. Uji coba ini bertujuan untuk menguji fungsi dari tombol-tombol yang disediakan oleh form. Proses pengujian form ini dapat dilihat pada gambar 3.

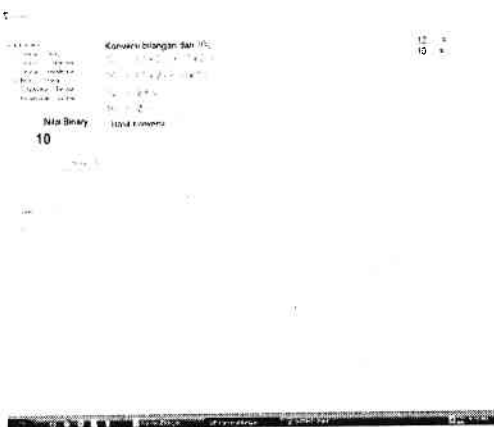


Gambar 3. Pengujian Konversi

Untuk menguji Desimal ke Biner, Oktal dan Hexa, Inputkan nilai bilangan yang akan dikonversikan klik tombol hitung tunggu sistem memproses data yang kemudian sistem langsung menampilkan hasil konversi pada kolom panel. Potongan script programnya seperti berikut:

```
var Ni_A, Ni_B, Ni_C : Extended;
StrNi_Bi, Ni_Hasil : String;
LBA.Items.Clear;
LBB.Items.Clear;
LBC.Items.Clear;
LBD.Items.Clear;
Ni_A := StrToFloat(Ni_Asal);
While Ni_A > 0 do
Begin
Ni_B := DoRoundDown(Ni_A /
Ni_Pembagi);
Ni_C := Ni_A - (Ni_B * Ni_Pembagi);
StrNi_Bi := FloatToStr(Ni_C) +
StrNi_Bi;
Ni_Hasil := FloatToStr(Ni_C);
```

Untuk menguji konversi biner, oktal dan hexa ke desimal lakukan hal yang sama seperti diatas. Inputkan bilangan kemudian klik tombol hitung tunggu sistem memprosesnya maka hasil konversi akan tampil di kolom panel. Berikut ini adalah hasil pengujian dari konversi hexa ke desimal.

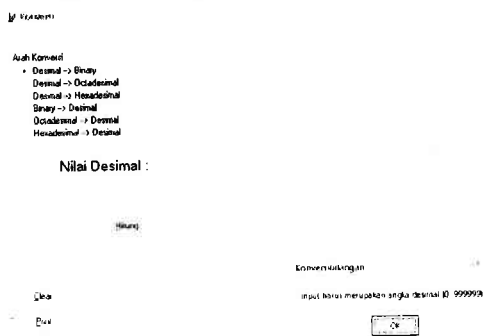


Gambar 4. Pengujian Konversi Biner, Oktal dan Hexa ke Desimal

Berikut ini potongan script program untuk konversi biner, oktal, hexadesimal ke desimal:

```
HasilNi_A : String;
Ni_A, Ni_B, Ni_C, Ni_D : extended;
c : Integer;
StrNi : String;
for c := 1 to length(Ni_Asal) do
begin
StrNi := copy(Edit1.Text, c, 1);
HasilNi_A := StrNi;
```

Pengujian dengan menginput bilangan-bilangan dengan nilai ekstrim seperti Bilangan float, negatif dan bilangan 0 jika dicoba dengan sistem maka menghasilkan input seperti berikut:



Gambar 5. Pengujian Bilangan Dengan Nilai Ekstrim

### 3.4.3. Uji Coba Validasi

Pengujian validasi merupakan pemeriksaan keakuratan hasil data yang telah dimasukkan kedalam sistem aplikasi. Dimana data yang diinput kemudian dihitung dengan menggunakan sistem sesuai dengan data yang dihitung dengan proses manual. Berikut ini uji coba validasi sistem konversi desimal:

Misalkan contoh bilangan yang dikonversikan adalah bilangan  $5_{(10)}$  dan  $10_{(2)}$ .

1. Konversi Dari Desimal ke Biner, Oktal dan Hexa

Bilangan 5

Perhitungan manual :

$$\begin{array}{r} 5 \\ \hline 2 \\ \hline 1 \\ \hline 0 \end{array} \quad \begin{array}{l} 2 \text{ sisa } 1 \\ 2 \text{ sisa } 0 \\ 2 \text{ sisa } 1 \end{array}$$

Hasil : 101<sub>2</sub>

Dengan sistem :

Arah Konversi

- Desimal → Biner
- Desimal → Octadesimal
- Desimal → Hexadesimal
- Biner → Desimal
- Octadesimal → Desimal
- Hexadesimal → Desimal

Konversi bilangan dari 5<sub>(10)</sub>

5	2 Sisa	1
2	2 Sisa	0
1	2 Sisa	1
0		

Nilai Desimal : 5

Hasil Konversi : 101<sub>2</sub>

Hitung

Gambar 6. Pengujian Validasi Konversi Desimal ke Biner, Oktal dan Hexa

Konversi bilangan adalah cara mengubah atau mengalihkan bilangan ke sistem bilangan lainnya. Mengubah bilangan dari desimal ke biner, oktal dan hexadesimal adalah dengan pembagian sisa. Bilangan desimal yang akan diubah secara berturut-turut dibagi bilangan basis konversi dengan memperhatikan sisa pembagiannya. Untuk bilangan biner sisa pembagian akan bernilai 0 atau 1 dengan sisa yang terakhir menunjukkan MSBnya. Dalam bilangan hexadesimal untuk nilai sisa yang menghasilkan sisa 10 sampai 15 diterjemahkan menjadi menjadi huruf A sampai F.

Sedangkan untuk konversi sebaliknya dari biner, oktal dan hexadesimal ke desimal dilakukan dengan cara perkalian masing-masing bit dalam bilangan basis dengan *place value* nya. Aplikasi konversi bilangan ini dijalankan dengan memilih salah satu arah konversi selanjutnya inputkan bilangan dan hasil konversi akan tampil pada kolom panel.

Dalam menginput bilangan sistem konversi ini terbatas hanya untuk bilangan asli atau bilangan yang berbentuk integer positif. Dari pengujian yang dilakukan dalam menginput bilangan, sistem tidak menerima bilangan dengan nilai-nilai yang ekstrim seperti bilangan negatif, bilangan float dan bilangan 0. Untuk input bilangan hexadesimal penulisan bilangan yang menggunakan huruf harus ditulis dengan huruf kapital.

3.5. Penggunaan Sistem

Untuk penggunaan sistem pada form menu utama yaitu dengan menginputkan data yang dikonversikan. Tentukan arah konversi bilangan tersebut kemudian klik tombol hitung secara otomatis hasil konversi akan tampil satu persatu pada kolom panel. Gunakan *clear canvas* untuk membersihkan tampilan panel, untuk melihat laporan klik tombol *print*. Lakukan hal yang sama untuk proses konversi bilangan lainnya. Pilih sub menu *close* untuk keluar dari aplikasi.

IV. KESIMPULAN DAN SARAN

4.1. Kesimpulan

Setelah melakukan proses perancangan, implementasi dan pengujian pada Aplikasi Konversi Bilangan dari Desimal ke Biner, Oktal, Hexadesimal dan sebaliknya maka didapatkan kesimpulan sebagai berikut:

1. Remainder Method (Metode Pembagian Sisa) merupakan metode yang paling umum digunakan untuk konversi Desimal ke biner, Oktal dan Hexadesimal.
2. Konversi Biner, Oktal dan Hexadesimal ke Desimal dilakukan dengan cara mengalikan place value dengan absolute digit bilangan awal basis.
3. Sistem konversi bilangan ini diimplementasikan dengan pemrograman Borland Delphi 7.0. Dengan adanya sistem konversi bilangan ini membuat proses konversi bilangan menjadi lebih mudah dan cepat.
4. Pengujian yang dilakukan pada aplikasi konversi bilangan Desimal ke Biner, Oktal, Hexadesimal dan sebaliknya didapat beberapa kekurangan dari sistem tersebut antara lain bahwa input bilangan yang dimasukkan ke dalam sistem hanya untuk bilangan berbentuk integer positif atau bilangan asli. Sehingga sistem tidak menerima bilangan dengan nilai yang ekstrim seperti bilangan 0, bilangan float dan bilangan negatif.

#### 4.2. Saran

Penggunaan sistem konversi Desimal ke Biner, Oktal, Hexadesimal dan sebaliknya ini terbatas hanya untuk bilangan berbentuk integer positif atau bilangan asli sehingga untuk bilangan berbentuk integer pecahan tidak dapat

dihitung dengan sistem konversi tersebut. Jika ingin mengubah nilai bilangan dengan jumlah digit yang lebih besar maka sebaiknya menggunakan algoritma dengan optimasi yang menerapkan Algoritma *Divide and Conquer*.

#### DAFTAR PUSTAKA

**Dwiantoro, Tino.** 2006. *Pengantar Teknologi Informasi*, Gramedia, Jakarta.

**Hartono, Jogyanto.** 2004. *Pengenalan Komputer*, Andi, Yogyakarta.

**Hendrayudi.** 2007. *Pemrograman Delphi 7.0*, Yrama Widya, Bandung.

<http://jkw1.files.wordpress.com/2007/11/microprocessor.pdf>, [ 25 Juli 2008 ]

**Munir, Rinaldi.** 2002. *Algoritma dan Pemrograman*, Informatika, Bandung.

**Nugroho.** 2005. *Komponen Visual Pada Delphi 7.0*, Graha Ilmu, Yogyakarta.

**Simarmata, Janner.** 2007. *Dasar-Dasar Pemrograman Dengan Delphi 7.0*, Gramedia, Jakarta.

**Sutedjo Budi,** 2005 *Algoritma & Teknik Pemrograman*, Penerbit Andi Yogyakarta

**[http://www.dwiantoro.com/document/s/PTI\\_TND.pdf](http://www.dwiantoro.com/document/s/PTI_TND.pdf)**, [ 24 Agustus  
2 0 0 8  
]**[http://www.kleq.web.ugm.ac.id/modul\\_4\\_PTI.pdf](http://www.kleq.web.ugm.ac.id/modul_4_PTI.pdf)**, [25 Juli  
2008]

**<http://www.materikuliaah.com>**, [19  
September 2008]

**Yuda, C.** 2004. *Panduan Objek Oriented Programming (Dasar-Dasar Pemrograman Delphi)*, Andi, Yogyakarta.