

PERMAINAN SOAL
DI
KOMPUTER

ditulis oleh
GUNAWAN SUSILO
SMP NEGERI 1 GANDUSARI
BLITAR

BLITAR - MEI - 2007

DAFTAR ISI

PERMAINAN SOAL DI KOMPUTER.....	i
DAFTAR ISI	ii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Tujuan.....	2
BAB II PERANGKAT LUNAK MENGUJI DIRI	4
2.1 Menjalankan SOAL.EXE	4
2.1.1 Menjalankan melalui Run.....	5
2.1.2 Menjalankan dengan Shortcut	5
2.2 Membuat Soal.....	6
2.3 Menyisipkan gambar pada soal	8
2.4 Menyisipkan audio pada soal.....	9
2.5 Menyisipkan video pada soal.....	9
BAB III ALAT BANTU MENYUSUN SOAL	11
3.1 Variabel.....	11
3.1.1 Variabel Numerik	11
3.1.2 Variabel String.....	12
3.2 Operator Aritmatika.....	13
3.3 Procedure dan Fungsi	13
3.3.1 Fungsi	13
3.3.2 Procedure.....	16
3.4 Pengarah @.....	17
BAB IV FORMAT PENULISAN SOAL	19
4.1 Mengatur warna Tulisan.....	19
4.1.1 Menggunakan [cl...].	19
4.1.2 Menggunakan [rgbr,g,b].	20
4.2 Mengatur Style penulisan	20
4.2.1 Menggunakan [fs..].	21

4.2.2	Menggunakan [=..] dan [>..]	21
4.2.3	Menggunakan [#..]	22
4.3	Menulis indek, pangkat, akar, pecahan dan elemen vector dua dimensi	22
4.4	Mengatur penulisan baris berikutnya	23
4.5	Menulis dengan hurub lain atau menggunakan karakter font Simbol	23
BAB V PENUTUP		25
5.1	Mendistribusikan Soal	25
5.2	Mengganti Logo Tutwuri dan Gambar Pengawas	25
LAMPIRAN DAFTAR NILAI ASCII DAN KARAKTER SIMBOL		27

BAB I

PENDAHULUAN

1.1 Latar Belakang

Tehnologi telah berkembang, memasuki dunia Telekomunikasi dan Informasi yang canggih. Sistem komputerisasi telah merambah ke segala aspek kehidupan, termasuk dunia pendidikan.

Kemampuan menggunakan tehnologi menjadi sangat besar pengaruhnya dalam usaha meningkatkan kualitas pendidikan, dan bukan mustahil sebuah bangsa akan menjadi tertinggal akibat kemampuan menguasai atau menggunakannya lambat.

Bagi pendidik menguasai tehnologi bukanlah hal yang mudah karena masih banyak factor yang harus diselesaikan. Sarana-prasarana yang kurang, pengetahuan yang sulit didapat, tingkat perekonomiannya kadang masih bermasalah, kesadaran belajar siswa rendah, dan kesadaran mendidik keluarga maupun masyarakat masih harus dipertanyakan.

Persoalan-persoalan klasik tersebut memang harus diselesaikan, tetapi tehnologi yang terus berkembang janganlah dibiarkan meninggalkan bangsa ini, termasuk dunia pendidikan karena semakin lambat kita mengikuti maka semakin jauh kita ditinggalkan, dan pada akhirnya akan semakin rendah kemampuan besaing dalam mempertahankan kehidupan bangsa ini di dunia.

Tuntutan dan kondisi tersebut membuat penulis berusaha agar dapat ikut serta mengembangkan penggunaan tehnologi di dalam pendidikan, terutama melalui tehnologi komputer.

Melalui kegiatan di computer penulis berusaha memacu motivasi siswa untuk belajar dan terus senang belajar. Beberapa factor yang dapat meningkatkan motivasi anak didik diantaranya menciptakan:

1. Kegiatan bersifat permainan, ada kebebasan yang diikat dengan aturan.
2. Kegiatan terbuka dan segera mendapat umpan balik untuk merasa bangga atau harus berusaha meningkatkan kemampuannya.
3. Kegiatan yang adil dengan aturan yang terbuka dan dapat diterima mereka.

Banyak sekali bentuk kegiatan yang dapat menghasilkan factor diatas. Tetapi disini penulis mengambil bentuk yang sering dilakukan oleh pendidik adalah melalui mengerjakan soal-soal.

Soal-soal dalam permainan computer ini mempunyai beberapa keunikan, diantaranya adalah:

1. Soal keluar secara acak, sesuai arah yang telah direncanakan penulis soal.
2. Soal dapat dibuat bervariasi sesuai kemampuan pendidik dalam merencanakannya.
3. Dapat memberi tahu hasilnya secara langsung setelah usai dikerjakan.
4. Jika dikehendaki penulis, kunci jawaban bisa ditampilkan sebagai bentuk keterbukaan dan pembelajaran yang bersifat informative.
5. Soal dapat mencakup materi audio dan visual.

Pembuatan kegiatan seperti yang penulis rencanakan tentu bukan tanpa masalah karena persoalan pada pendidik sendiri yang masih banyak, juga kemampuan rata-rata pendidik di bidang teknologi computer khususnya masih kurang. Oleh karena itu, melalui media ini penulis mencoba menawarkan perangkat lunak yang dapat mempermudah pendidik mewujudkan kegiatan tersebut.

1.2 Tujuan

Perangkat lunak yang berhasil dibuat penulis ditujukan untuk membantu pendidik pada khususnya, dan seluruh masyarakatan pecinta pendidikan secara umum, agar dapat meningkatkan motivasi belajar anak didik.

Nampaknya perangkat tersebut hanya mengarah pada sebuah permainan mengerjakan soal di hadapan computer, tetapi dengan kegiatan tersebut anak didik memperoleh:

1. kesempatan menguji diri,
2. informasi – informasi yang belum diketahui,
3. rasa senang karena mendapat perlakuan yang sama,
4. rasa puas karena ada bentuk keterbukaan penilaian.

Sedangkan keuntungan yang dapat diperoleh pendidik dari perangkat tersebut adalah:

1. menambah teknik penilaian,

2. dapat menyusun soal sendiri untuk kegiatan permainan tersebut yang umumnya sulit dilakukan,
3. dapat menyampaikan informasi atau pengetahuan pada anak didik sambil bermain,
4. mengurangi beban koreksi guru
5. dapat menyertakan materi audio-visual dalam permainan

BAB II

PERANGKAT LUNAK MENGUJI DIRI

2.1 Menjalankan SOAL.EXE



Perangkat ini hanyalah sebuah file aplikasi yang diberi nama SOAL.EXE. Icon file tersebut tampak pada gambar (1) . Pada saat diakses akan mencari sumber soal yang tempatnya disertakan melalui parameter pertama. Contoh:

Gambar (1)

SOAL.EXE IKASOAL.TXT

Contoh diatas akan menampilkan program menguji diri dengan soal diambil dari file IKASOAL.TXT. File IKASOAL.TXT saat itu berada pada direktori yang sama dengan file SOAL.EXE.

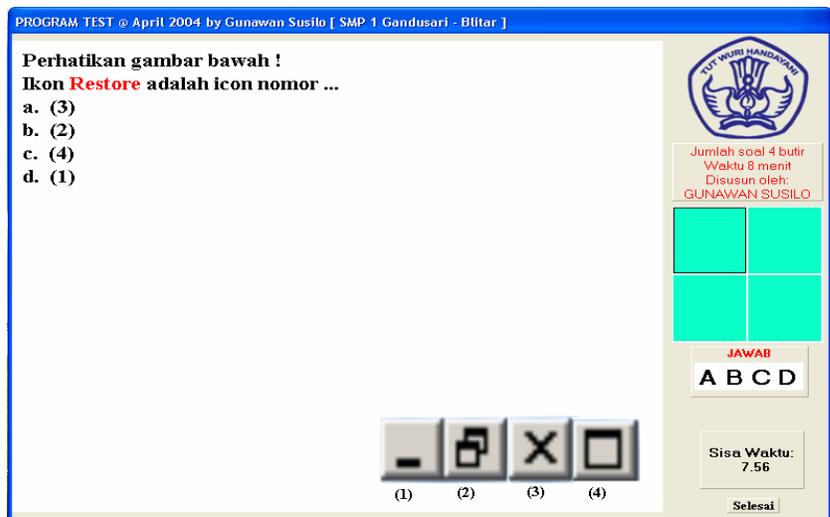
File soal yang akan diujikan juga dapat diletakkan didalam direktori yang berada di dalam direktori yang memuat file SOAL.EXE, misalnya:

SOAL.EXE MAT\IKASOAL.TXT

Maka file IKASOAL.TXT yang akan diujikan berada dalam direktori MAT\ dan direktori MAT\ terletak didirektori tempat file SOAL.EXE berada.

Jika pencarian sumber soal berhasil dan sumber tersebut memenuhi prasyarat sebagai soal maka akan tampil program uji diri seperti gambar dibawah ini.

Karena file SOAL.EXE membutuhkan parameter soal yang ingin ditampilkan maka cara menjalankannya tidak bisa langsung mengakses ikonnya, tetapi menjalankannya dapat menggunakan berbagai alternatif. Antara lain melalui:

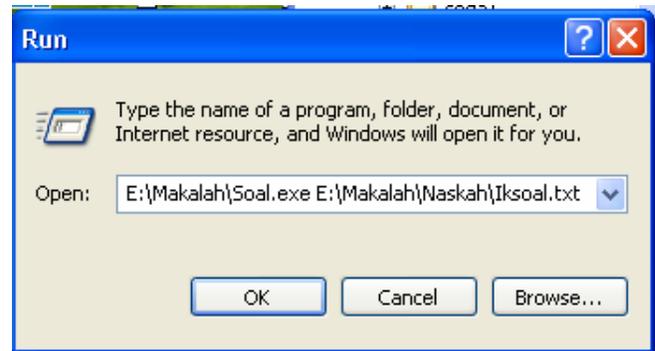


Gambar (2)

2.1.1 Menjalankan melalui Run

Untuk mengakses program dari RUN dapat menggunakan langkah – langkah berikut:

- Start
- Run hingga muncul dialog seperti gambar (3) .



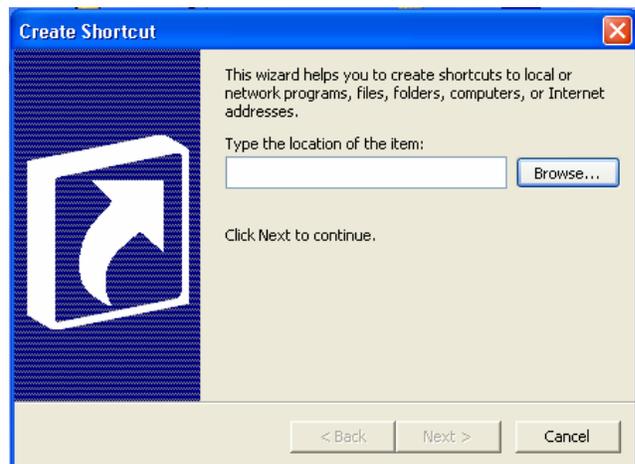
Gambar (3)

- Isi Open dengan posisi file SOAL.EXE diikuti tempat soal yang ingin di tampilkan. Pada contoh disamping file SOAL.EXE ada di direktori **E:\Makalah** sedangkan soal yang ditampilkan ada di direktori **E:\Makalah\Naskah** dengan nama file IKASOAL.TXT.
- Oke

2.1.2 Menjalankan dengan Shortcut

Cara ini lebih praktis dan mudah menjalankannya. Langkah-langkah untuk membuat adalah:

- Klik kanan pada desktop atau folder tempat shortcut akan dibuat
- Klik New
- Klik Shortcut hingga muncul dialog gambar (4)
- Isi Type the location of the items dengan lokasi file **SOAL.EXE** diikuti file yang akan ditampilkan seperti isian pada saat menggunakan **RUN**
- Tekan Next hingga berakhir finish dan terbentuk icon shortcut seperti gambar (5).



Gambar (4)

Setelah shortcut terbentuk, jika ingin menjalankan program **SOAL.EXE** dan isinya cukup membuka (Double Klik) shortcut tersebut.



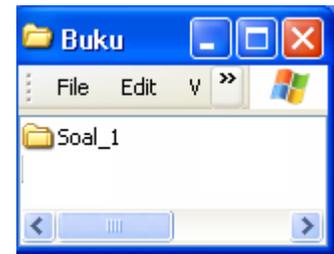
Gambar 5

Shortcut tersebut dapat kuta salin/ pindah ke folder lain tetapi eksekusinya tetap mengarah pada file-file yang telah ditunjuk.

2.2 Membuat Soal

Agar proses pembuatan soal berjalan mudah, maka sebelum memulai membuat soal lakukan langkah-langkah berikut:

1. Buatlah sebuah folder langsung dibawah drive tertentu. Misal disini kita buat folder **Buku** di bawah direktori **C:**
2. Masukkan CD Asli yang berisi file **SOALE.EXE**, **BUAT_SOALE.EXE**, **SANDI.EXE** ke CDRom computer anda.
3. Buat folder baru (misal: **Soal_1**) sebagai tempat menempatkan data soal kita yang pertama. Sehingga isi folder **C:\Buku** akan tampak seperti gambar (6).



Gambar (6)

Setelah folder Buku diatas telah tersedia kita buka program

NOTEPAD yang ada di computer (melalui:Start-Program-Assesories-Notepad) lalu kita ketikkan naskah soal seperti gambar (7).

Simpan naskah tersebut di folder **C:\Buku\Soal_1** dengan nama file

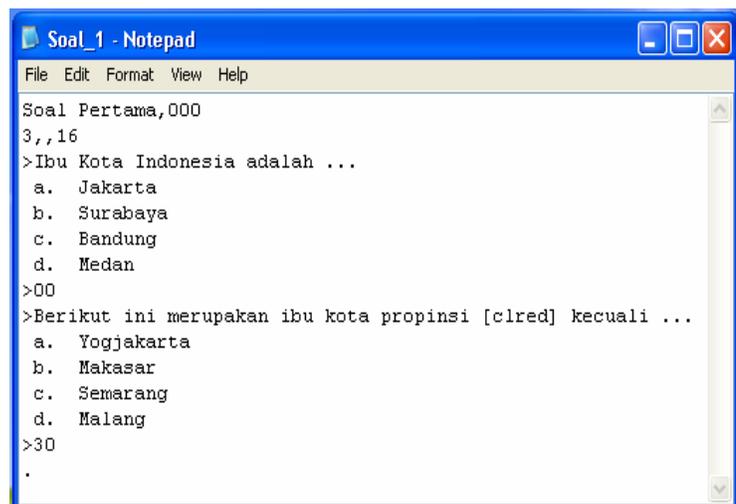
Soal_1.TXT, kemudian

bukalah drive CDRom untuk

menjalanka program

BUAT_SOALE.EXE langsung dari ikonnya.

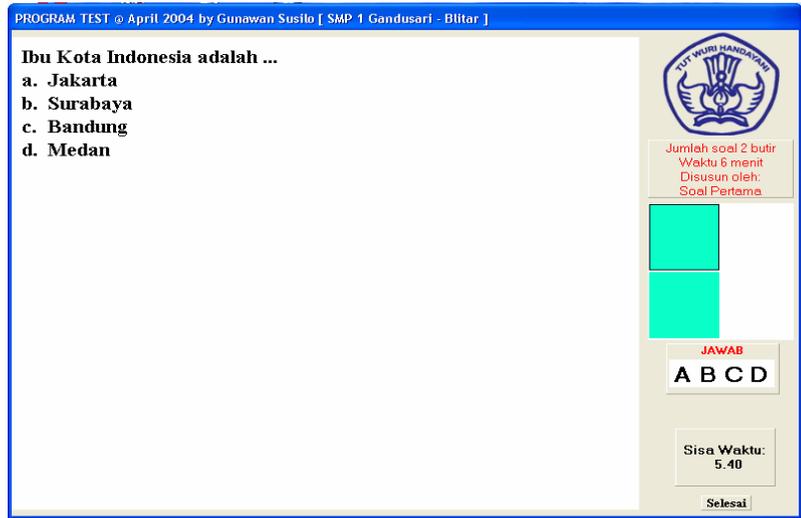
Pada saat **BUAT_SOALE.EXE** dijalankan akan meminta file soal yang akan ditampilkan, pilihlah file **Soal_1.TXT** di folder **C:\Buku\Soal_1** yang telah kita buat. Apa yang terjadi ! Jika penulisan anda benar akan tampil gambar (8), tampilan hasil penulisan naskah anda dijalankan dengan program Test (**SOALE.EXE**).



Gambar (7)

Baris pertama pada naskah **SOAL_1.TXT** mengandung informasi:

- Kalimat sebelum tanda koma (**Soal Pertama**) akan tampil sebagai identitas penyusun, lihat gambar (8).
- Angka pertama setelah koma sebagai tanda soal diacak tempatnya atau tidak. Jika berisi **1** maka soal akan diacak, tetapi apabila **0** pilihan soal akan ditampilkan sesuai urutan pembuatan.



Gambar (8)

- Angka kedua setelah koma sebagai tanda pilihan diacak atau tidak. Jika berisi **1** maka pilihan soal akan diacak, tetapi apabila **0** maka pilihan soal akan ditampilkan sesuai urutan pembuatan.
- Angka ketiga dan selanjutnya menyatakan jumlah file gambar yang disertakan. Misal ada **12** file gambar yang disertakan pada **SOAL_1.TXT** maka baris pertama tersebut akan tertulis:

Soal Pertama,1112

Baris kedua pada naskah **SOAL_1.TXT** mengandung informasi:

- Angka pertama sebelum koma menunjukkan jumlah waktu untuk setiap soal dalam satuan menit. Jadi di naskah tersebut setiap soal **3 menit**.
- Angka diantara koma menunjukkan jumlah soal yang diujikan (maksimal sama dengan jumlah soal), Dalam naskah tersebut **tidak diisi** dianggap **semua soal diujikan**.
- Angka diantara koma kedua adalah ukuran hurub yang digunakan dalam menulis soal. Dalam contoh diatas adalah **16**.
- Masih terdapat satu bilangan lagi dibelakang bilangan 16 tersebut yang dipisahkan dengan koma. Apabila diisi **1** maka kunci jawaban setelah selesai mengerjakan akan ditampilkan, tetapi jika diisi **0** atau tidak ada maka kunci jawaban tidak pernah ditampilkan.

Baris soal selalu diawali karakter “>” dan ditutup dengan “>”. Untuk karakter “>” penutup diikuti beberapa kode. Karakter **pertama** pada kode setelah karakter “>” penutup merupakan kunci jawaban soal di atasnya. Sedangkan karakter kedua pada kode setelah karakter “>” harus diisi 0. Ketentuan karakter kunci adalah:

- 0 jika jawaban soal adalah A
- 1 jika jawaban soal adalah B
- 2 jika jawaban soal adalah C
- 3 jika jawaban soal adalah D

Penulisan soal setelah baris pertama didahului dengan sebuah spasi.

Soal bagian **jawaban pilihan** harus tersusun mulai dari **a** hingga **d** dan selalui diikuti dengan titik.

2.3 Menyisipkan gambar pada soal

Sebelum kita menulis naskah soal yang mengandung gambar, kita harus menyiapkan file-file gambar yang berisi gambar-gambar yang ingin kita sisipkan. File tersebut disimpan dalam format **BMP** dan diberi nama berupa **nomor**, mulai nomor 1. File-file gambar tersebut kita simpan satu folder dengan naskah soalnya.

Untuk menyertakan gambar ke soal dilakukan dengan menulis “**Gb**” diikuti **nomer file** dan karakter “\” kemudian koordinat **pojok kiri atas** serta **pojok kanan bawah** yang berupa daerah persegi panjang dan memuat gambar tersebut. Penulisan tersebut dimulai dari karakter **ke 4** pada baris akhir dari soal (sebaris dengan kunci).

Misalnya kita ingin menyertakan gambar di file **1.BMP**, dengan koordinat pojok-pojok gambar **(10,200)** dan **(130,260)** maka dibagian akhir soal yang memuat gambar tersebut harus ditulis.

>20Gb1.bmp\10,200,130,260

contoh diatas kita menganggap kunci jawabannya **C**. Kenapa ?

Semua file yang ingin dilibatkan dalam soal harus didaftarkan pada **baris pertama** dari naskah tersebut.

2.4 Menyisipkan audio pada soal

File suara yang dapat disisipkan pada soal berupa file **MP3** dan seluruhnya dimasukkan ke dalam folder naskah soal tersebut. Disarankan file-file tersebut diberi nama bilangan. (misal: 1.MP3).

Untuk menyertakan suara ke soal dinyatakan dengan menulis `\a\nama_file` pada baris akhir soal, tepatnya mulai karakter ke **empat** (sebaris dengan kunci soal).

Misalnya:

```
>30\a\015005
```

Contoh diatas menunjukkan jawaban soal adalah **D**, dan saat soal ditampilkan akan diperdengarkan suara dari file **015005.MP3**.

Suara dan gambar juga dapat disisipkan sekaligus dalam sebuah soal. Untuk menyisipkannya dapat dilihat contoh berikut,

```
>00\a\015005\Gb11.bmp\0,0,270,110
```

Dari penulisan diatas, jawaban soal adalah **A**, suara yang diperdengarkan berasal dari file **0105005.MP3**, gambar yang ditampakkan berasal dari file **11.BMP** dengan pojok pojoknya **(0,0) – (270,110)**.

Jika hanya ingin memainkan sebagian dari suara maka setelah nama file audio diikuti empat parameter dengan pemisah koma sehingga:

```
>00\a\015005,1,4,1,2\Gb11.bmp\0,0,270,110
```

artinya audio dimainkan mulai $\frac{1}{4}$ sampai $\frac{1}{2}$ dari keseluruhan file.

2.5 Menyisipkan video pada soal

File vidio yang dapat disisipkan pada soal berupa file **MPG** dan seluruhnya dimasukkan ke dalam folder naskah soal tersebut. Disarankan file-file tersebut diberi nama bilangan. (misal: 1.MPG).

Untuk menyertakan suara ke soal dinyatakan dengan menulis `\v\nama_file` pada baris akhir soal, tepatnya mulai karakter ke **tiga** (sebaris dengan kunci soal).

Misalnya:

```
>10\v\015
```

Contoh diatas menunjukkan jawaban soal adalah **B**, dan saat soal ditampilkan akan dimainkan vidio dari file **015.MPG**.

Jika hanya ingin memainkan sebagian dari video maka setelah nama file video diikuti empat parameter dengan pemisah koma sehingga:

>10\015,1,4,1,2

artinya video dimainkan mulai $\frac{1}{4}$ sampai $\frac{1}{2}$ dari keseluruhan file.

BAB III

ALAT BANTU MENYUSUN SOAL

Agar soal-soal yang ditampilkan tidak mudah ditebak/dihafal maka soal tersebut harus cukup bervariasi. Untuk membentuk variasi pada proses pembuatan soal dapat digunakan beberapa operator, fungsi, procedure, dan variable.

3.1 Variabel

Variabel dalam menyusun soal sangat terbatas (hanya 26 buah). Terdapat dua jenis variable yaitu:

3.1.1 Variabel Numerik

Variabel ini namanya sudah ditetapkan, yaitu A sampai dengan Z, dan hanya bisa diisi dengan bilangan. Cara mengisi variable numeric dapat melalui operator **sama dengan** (=). Nama variable dan tanda sama dengan tidak boleh ada jarak. Contoh:

$$\begin{aligned} A &= 12 \\ B &= 25 \end{aligned}$$

arti dari pernyataan diatas adalah mengisi A dengan nilai 12, dan mengisi B dengan nilai 25.

Variabel dapat diisi dengan nilai variable yang lain melalui tanda pengenalan **&** yang diletakkan didepan variable pengisi. Contoh:

$$A = \&B$$

artinya nilai variable A diisi dengan nilai variable B.

Nilai variabel numeric dapat ditampilkan dalam soal dengan operator **&** yang diletakkan didepan nama variable diikuti sebuah angka yang menyatakan jumlah decimalnya. Contoh:

$$\begin{aligned} A &= 12 \\ B &= 18.569 \\ &> \text{Ali mempunyai } \&A^2 \text{ cm tali jerami,} \\ &\text{sedangkan Badu mempunyai } \&B^0 \\ &\text{bungkus kacang goreng.} \end{aligned}$$

Penggalan naskah soal diatas pada saat dijalankan akan menghasilkan bentuk:

*Ali mempunyai 12.00 cm tali jerami,
sedangkan Badu mempunyai 19
bungkus kacang goreng.*

3.1.2 Variabel String

Variabel String hanya bisa diisi dengan kalimat. Variabel ini tidak punya nama tetapi disediakan berdasarkan **nomor**. Ada 26 nomor yaitu nomor 0 hingga nomor 25.

Untuk mengisi variable ini hanya ada satu cara yaitu melalui procedure **SET\$**. Contoh:

```
SET$(A,Ani,Bayu,Jln. Imam Bonjol 12A,Gajah Mada Plaza)
```

arti pernyataan diatas mengisi variable string nomor **0** hingga nomor **3** sebagai berikut:

nomor **0** (0 diwakili A) diisi **Ani**
nomor **1** (1 diwakili B) diisi **Bayu**
nomor **2** (2 diwakili C) diisi **Jln. Imam Bonjol 12 A**
nomor **3** (3 diwakili D) diisi **Gajah Mada Plaza**

Perlu diingat A **bukanlah** nama variable tetapi hanyalah bentuk yang mewakili 0, begitu juga hurub-hurub berikutnya akan mewakili angka berikutnya.

Sehinga **SET\$(K,.....)** artinya akan mengisi variable string mulai nomor 10.

Penulisan isi variabel string dilakukan dengan tanda **&** didepan variable numeric yang berisi nomor variable string yang akan ditampilkan dan diikuti tand **\$**. Contoh

```
SET$(A,Ani,Bayu,Jln. Imam Bonjol  
12A,Gajah Mada Plaza)  
A= 1  
B= 2  
C= 3  
D= 4  
>&A$ dan &B$ pergi ke &D$  
yang ada di &C$
```

Rangkaian pernyataan pada naskah soal diatas akan menampilkan bentuk:

*Ani dan Bayu pergi ke Gajah Mada Plaza
yang ada di Jln. Imam Bonjol 12 A*

Disamping SET\$, juga terdapat procedure SETNO\$. Kedua procedure sama, hanya saja SETNO\$ mempunyai parameter pertama adalah bilangan.

3.2 Operator Aritmatika

Dalam penyusunan naskah soal kita dapat menggunakan beberapa operator aritmatika. Berdasarkan prioritas operasinya operator tersebut adalah:

- (.....) pengelompokan
- ^ perpangkatan bilangan ($a^b = a^b$ dengan **b bilangan cacah**)
- * dan / perkalian dan pembagian
- + dan - penjumlahan dan pengurangan

Operator aritmatika ini hanya dapat dikenakan pada bilangan (tidak pada variable), sehingga jika ingin mengoperasikan variable maka yang dioperasikan adalah isinya.

Contoh:

```
A= 4
B= 8
X= &A ^ 2 + &B ^ 2
Y= &B ^ 2 - &A ^ 2
Z= 2*(&A*&B)
```

>Sebuah segitiga siku-siku mempunyai sisi paling panjang &X0, sedangkan salah satu sisi lainnya adalah &Y0. Maka panjang sisi ketiganya adalah a. &Z0

Penggalan diatas jika akan memberikan:

Sebuah segitiga siku-siku mempunyai sisi Paling panjang 80, sedangkan salah satu Sisi lainnya adalah 48. Maka panjang sisi ketiganya adalah ...
a. 64

3.3 Procedure dan Fungsi

3.3.1 Fungsi

Merupakan proses yang menghasilkan nilai. Nilai yang dihasilkan dapat digunakan untuk mengisi variable tertentu. Sesuma fungsi disini mempunyai parameter bilangan. Beberapa macam fungsi diantaranya:

- RND(nilai)

Fungsi yang menghasilkan sebuah bilangan acak dari **0** sampai dengan **nilai - 1**. Contohnya:

$$A = \text{RND}(9) + 1$$

Pernyataan diatas akan mengisi A dengan sembarang bilangan yang diambil salah satu dari 1 sampai 9.

- **ABS(nilai)**

Fungsi yang menghasilkan nilai absolute (positif) dari **nilai**.

Contoh:

$$A = 2^3$$

$$B = \text{ABS}(\&A - 10)$$

Nilai B akan berisi 2.

- **INT(nilai)**

Fungsi yang menghasilkan bilangan bulat dari pembulatan **nilai**.

Contoh:

$$A = 25/4$$

$$B = \text{INT}(\&A)$$

Nilai dari B adalah 6 karena 6 merupakan hasil pembulatan hasil perhitungan dari $25/4 = 6.25$

- **COS(nilai)**

Fungsi yang menghasilkan nilai trigonometri **cosinus** dari sudut **nilai**. Nilai disini dianggap mempunyai satuan derajat.

- **SIN(nilai)**

Fungsi yang menghasilkan nilai trigonometri **sinus** dari sudut **nilai**. Nilai disini dianggap mempunyai satuan derajat.

- **TAN(nilai)**

Fungsi yang menghasilkan nilai trigonometri **tangen** dari sudut **nilai**. Nilai disini dianggap mempunyai satuan derajat.

- **ARCTAN(nilai)**

Fungsi yang menghasilkan invers nilai trigonometri **tangen** dari **nilai**.

- **ARCSIN(nilai)**

Fungsi yang menghasilkan invers nilai trigonometri **sinus** dari **nilai**.

- **ARCSIN(nilai)**

Fungsi yang menghasilkan invers nilai trigonometri **sinus** dari **nilai**.

- **EXP(nilai)**

Fungsi yang menghasilkan nilai e^{nilai} .

- **EXP10(nilai)**

Fungsi yang menghasilkan nilai 10^{nilai} .

Contoh:

$$A = \text{EXP}(5)$$

$$B = \text{EXP}(2)$$

$$C = \&A + \&B$$

Pernyataan diatas akan mengisi A dengan 10^5 dan B dengan 10^2 dan nilai didalam C adalah 100100.

- **LN(nilai)**

Fungsi yang menghasilkan nilai logaritma dari **nilai** dengan bilangan pokok e

$$\text{LN}(\text{nilai}) = \text{}^e\log(\text{nilai})$$

- **EXPXY(a,b)**

Fungsi ini mengembalikan nilai a^b dengan $a > 0$.

- **SQRT(nilai)**

Fungsi yang menghasilkan nilai akar kuadrat dari **nilai**. **Nilai** tidak boleh negatif.

- **PRIMA(nilai)**

Fungsi yang menghasilkan bilangan prima ke **nilai**. **Nilai** maksimal 100.

Contoh:

$$A = \text{EXPXY}(27, (1/3))$$

$$B = \text{SQRT}(10)$$

$$C = \text{PRIMA}(4)$$

$$> \text{Nilai } A = \&A0, B = \&B3, \text{ dan } C = \&C0$$

Penggalan pernyataan diatas akan menghasilkan bentuk:

$$\text{Nilai } A = 3, B = 2,162, \text{ dan } C = 7$$

- **MIN(nilai₁, nilai₂, nilai₃, ...)**

Fungsi ini menghasilkan nilai paling kecil dari nilai yang ditulis dalam kurung.

- **MAX(nilai₁, nilai₂, nilai₃, ...)**

Fungsi ini menghasilkan nilai paling besar dari nilai yang ditulis dalam kurung.

- **NOM(x, nilai₁, nilai₂, nilai₃, ...)**

Fungsi ini menghasilkan nilai yang berada pada urutan ke x setelah diurutkan dari besar ke kecil.

Contoh:

$$A = \text{MIN}(3, 8, 6, 5, -4)$$

$$B = \text{MAX}(3, 8, 6, 5, -4)$$

$$C = \text{NOM}(3, 3, 8, 6, 5, -4)$$

$$> \text{Nilai } A = \&A0, B = \&B3, \text{ dan } C = \&C0$$

Penggalan pernyataan diatas akan menghasilkan bentuk:

$$\text{Nilai } A = -4, B = 8, \text{ dan } C = 5$$

- **MOD(x,y)**
Fungsi ini menghasilkan sisa pembagian dari x oleh y. x dan y bilangan asli.

- **IF(a,b,x,y,z)**
Fungsi ini menghasilkan nilai x jika $a > b$, menghasilkan y jika $a = b$, dan menghasilkan z jika $a < b$.

Contoh:

A = 14

B = 3

C = MOD(&A,&B)

D = IF(&A,&B,0,1,2)

E = IF(&C,&B,0,1,2)

F = IF(&A,14,0,1,2)

SET\$(A,lebih besar,sama dengan,lebih kecil)

>A = &A0, B = &B3, dan C = &C0

A &D\$ B

C &E\$ B

A &F\$ 14

Penggalan pernyataan diatas akan menghasilkan bentuk:

A = 14, B = 3.000, dan C = 2

A lebih besar B

C lebih kecil B

A sama dengan 14

3.3.2 Procedure

Merupakan proses yang tidak menghasilkan nilai. Tetapi akibat proses tersebut tersebut nilai variable kemungkinan bisa berubah. Beberapa procedure yang disediakan program adalah:

- **RND(var,cacah,Fc())**
Merupakan procedure yang mengisi variable **var** dan variable berikutnya sebanyak **cacah** dengan nilai acak yang berbeda. Nilai acak tersebut berupa bilangan yang dihasilkan oleh fungsi **Fc()**.

Contoh:

RND(C,3, RND(8))

akan mengisi variable C, D, dan E dengan bilangan acak yang berbeda dari 0 hingga 7 yang dihasilkan fungsi **RND(8)**.

- **SORT(var,cacah)**
Merupakan procedure yang mengatur ulang isi **var** dan berikutnya sejumlah **cacah**, sedemikian sehingga nilainyaurut ke atas.

Contoh:

SQRT(C,3)

akan mengatur isi variable C, D, dan E sehingga nilai yang dikandungnya menjadi urut dari kecil ke besar.

- **LRIK(var,var_ke,cacah)**

Merupakan procedure yang digunakan mengisi variable sebanyak **cacah** variable, mulai **var** dengan isi variable nomor **var_ke** dan nomor berikutnya.

Contoh:

```
_RND(G,4,RND(8))
_LRIK(A,6, 4)
```

pernyataan diatas sama dengan A= &G, B=&H, C=&I, D=&J.

- **FILL(var,cacah,isian)**

Merupakan procedure yang digunakan mengisi variable sebanyak **cacah** variable, mulai **var** dengan isi sesuai isiannya. Jika dalam rumus isian terdapat **_x_** maka akan diisi sebagai nilai penghitung mulai **a** sampai **cacah**.

Contoh:

```
_FILL(A,4,2*_x_+1)
```

pernyataan diatas sama dengan A= 1, B= 3, C= 5, D= 7.

- **JWB(var,cacah,Range)**

Merupakan procedure yang digunakan mengisi variable sebanyak **cacah** variable, setelah **var** dengan isi disekitar nilai **var** dengan selisih setengah **Range**.

Contoh:

```
JWB(A,20)
```

pernyataan diatas akan mengisi B sampai D dengan nilai sekitar 4 ± 10 . Nilai A sampai D tidak ada yang sama.

3.4 Pengarah @

Pengarah @ berfungsi mengarahkan pembuatan soal. Bentuk pengarah ini adalah:

@Jum_soal,Jum_pakei

perintah diatas akan mengambil sebanyak **Jum_pakei** soal dari **Jum_soal** yang disediakan setelah perintah ini. Procedure ini dilakukan sebelum pengambilan soal yang diarahkan oleh perintah **baris dua**.

Contoh:

```
@2,1
_RND(A,2,RND(50)+50)
C=&A+&B
>[clred]&A0 + &B0 = ...
A=&C
JWB(A,40)
a. &A0
b. &B0
c. &C0
d. &D0
>00
```

```
A=RND(100)+70
B=RND(80)+50
C=&A+&B
E=&A+&C
>[clred]&C0 - &A0 = ....
A=&B
JWB(A,40)
a. &A0
b. &B0
c. &C0
d. &E0
>00
```

Pernyataan @2,1 pada soal tersebut akan mengakibatkan dua soal yang ditulis dibaris berikutnya akan diambil satu soal secara acak.

BAB IV

FORMAT PENULISAN SOAL

Format penulisan disini berisi petunjuk/ sandi atau kode yang mengatur tampilan soal yang pada saat dijalankan menggunakan SOAL.EXE. Ada beberapa format penulisan diantaranya:

4.1 Mengatur warna Tulisan

Jika tanpa diatur, penulisan setiap baris soal dilakukan menggunakan warna hitam. Warna tersebut dapat diubah menggunakan tanda-tanda berikut.

4.1.1 Menggunakan [cl...]

Setelah penulisan [cl...] akan mengubah hasil tampilan pada soal berwarna ... (sesuai isi dari titik-titik). Adapun kode yang diperbolehkan mengisi adalah sebagai berikut:

- Aqua
- black
- blue
- dkgray
- fuchsia
- gray
- green
- lime
- ltgray
- maroon
- navy
- olive
- purple
- red
- silver
- teal
- white
- yellow

Cotoh:

```
>The following things belong to household  
[clred]equipment [clblack]...  
a. Axe, hoe, hammer  
b. Blender, mixer, rice cooker  
c. computer, radio, TV  
d. Thermometer, stethoscope, scissors  
>10
```

Pernyataan soal tersebut akan mengubah warna tampilan **equipment** menjadi warna merah. Tampilannya sebagai berikut:

```
The following things belong to household equipment ...  
a. Axe, hoe, hammer  
b. Blender, mixer, rice cooker  
c. computer, radio, TV
```

d. Thermometer, stethoscope, scissors

4.1.2 Menggunakan [rgbr,g,b]

Format penulisan ini, kata **[rgb** adalah harus ada, sedangkan **r**, **g**, dan **b** merupakan bilangan 0 sampai 255. Bilangan **r** menggambarkan komposisi warna **merah**, bilangan **g** merupakan komposisi warna **hijau**, dan yang terakhir adalah komposisi warna **biru**.

Contoh:

```
>[rgb90,24,170](1) What about going to a picnic?  
[rgb90,24,170](2) Where will we go?  
[rgb90,24,170](3) That's great!  
[rgb90,24,170](4) Tommorrow is Sunday, isn't it?  
[rgb90,24,170](5) Yes, what's the matter?  
[rgb90,24,170](6) Serang Beach.  
The correct arrangement is ....  
a. (4)-(5)-(1)-(3)-(2)-(6)  
b. (4)-(5)-(1)-(2)-(3)-(6)  
c. (4)-(5)-(1)-(6)-(2)-(3)  
d. (2)-(6)-(4)-(5)-(1)-(3)
```

>00

Pernyataan **[rgb90,24,170]** pada penggalan soal diatas akan menyebabkan warna tulisan dibelakangnya berubah menjadi warna komposisi **merah 90**, **hijau 24**, dan **biru 170**. Hasilnya seperti berikut:

- (1) What about going to a picnic?
- (2) Where will we go?
- (3) That's great!
- (4) Tommorrow is Sunday, isn't it?
- (5) Yes, what's the matter?
- (6) Serang Beach.

The correct arrangement is

- a. (4)-(5)-(1)-(3)-(2)-(6)
- b. (4)-(5)-(1)-(2)-(3)-(6)
- c. (4)-(5)-(1)-(6)-(2)-(3)
- d. (2)-(6)-(4)-(5)-(1)-(3)

4.2 Mengatur Style penulisan

Disini hanya sedikit style penulisan yang dapat digunakan. Untuk menggunakannya digunaka kode kode berikut:

4.2.1 Menggunakan [fs..]

Titik-titik dapat diisi angka 0 sampai 7, dan karakter minus (-). Makna angka adalah sebagai berikut:

1 untuk penulisan huruf tebal

2 untuk penulisan huruf miring

4 untuk penulisan huruf bergaris bawah

sehingga untuk angka hasil penjumlahan ketiga bentuk akan menghasilkan penulisan huruf yang angka angkanya dijumlahkan. Jadi [fs6] untuk penulisan huruf miring yang bergaris bawah ($6=2+1$).

Penulisan [fs-] digunakan untuk penulisan huruf yang dicoret.

Untuk membuat penulisan normal gunakan [fs0].

```
SET$(A, fs1,fs2,fs3,fs-)
```

```
A=RND(4)
```

```
>[&A$]Tahukah anda, apa style penulisan ini
```

- a. huruf tebal
- b. huruf miring
- c. huruf bergaris bawah
- d. huruf yang dicoret

```
>&A00
```

Penggalan soal diatas akan menampilkan bentuk soal yang acak dari empat macam pertanyaan, yaitu:

Jika A=0 maka hurufnya tebal

A=1 maka hurufnya miring

A=2 maka hurufnya bergaris bawah

A=3 maka hurufnya dicoret

Kunci pada soal ditulis >&A00 karena nilai A ditentukan secara acak dan digunakan sebagai kunci jawaban.

Salah satu kemungkinan dari soal yang ditampilkan adalah:

~~Tahukah anda, apa style penulisan ini ...~~

- a. **huruf tebal**
- b. **huruf miring**
- c. **huruf bergaris bawah**
- d. **huruf yang dicoret**

4.2.2 Menggunakan [=..] dan [>..]

Titik-titik diisi sebuah karakter dan diikuti tulisan yang akan dikenai kode tersebut. [=..] akan menyebabkan adanya garis mendatar pada posisi **nilai ascii karakter** dikurangi 41 dari atas tulisan kebawah.

[>..] sama seperti diatas, tetapi garis yang dibuat berupa panah ke arah kanan.

Kode **ascii karakter** nilai huruf A = 41, B = 42, ... dan seterusnya (coba lihat lampiran buku ini).

4.2.3 Menggunakan [#..]

Titik-titik diisi dengan angka **dua digit** yang menunjukkan tulisan setelah kode tersebut akan mulai ditulis pada jarak nilai angka tersebut dalam **dot (titik)** dari posisi paling kiri.

Contoh:

```
>[cblblue]Tia[#80]: Look. The man always works hard there
[cblblue]Adi[#80]: Yes, I know. [=SWhat a diligent man he is].
The uperline words expression is about ....
a. Certainty
b. Afraid
c. Possibility
d. Admiration
>30
```

Penggalan soal diatas akan menampilkan bentuk penulisan sebagai berikut:

Tia : Look. The man always works hard there
Adi : Yes, I know. What a diligent man he is.
The uperline words expression is about
a. Certainty
b. Afraid
c. Possibility
d. Admiration

4.3 Menulis indek, pangkat, akar, pecahan dan elemen vector dua dimensi.

Menulis indek digunakan [**i..**], Titik-titik akan ditulis dalam format indek.

Menulis pangkat digunakan [**p..**], sedangkan penulisan akar digunakan [**v..**]. Titik-titik akan ditulis dalam format pangkat atau didalam tanda akar.

Contoh:

```
>a[_10p1] dan a[p2] adalah akar-akar persamaan
ku_10adrat. Maka bentuk bentuk persamaan
ku_10adrat tersebut kemungkinan adalah ...
a. _10 x[p2] - (a[i1]+a[i2])x + a[i1].a[i2] = 0
b. _10 x[p2] + (a[i1]+a[i2])x + a[i1].a[i2] = 0
c. _10 x[p2] - (a[i1]+a[i2])x - a[i1].a[i2] = 0
d. _10 x[p2] + (a[i1]+a[i2])x - a[i1].a[i2] = 0
```

penulisan diatas akan menghasilkan bentuk:

a_1 dan a_2 adalah akar-akar persamaan kuadrat. Maka bentuk bentuk persamaan kuadrat tersebut kemungkinan adalah ...

- a. $x^2 - (a_1+a_2)x + a_1.a_2 = 0$
- b. $x^2 + (a_1+a_2)x + a_1.a_2 = 0$

$$c. \mathbf{x}^2 - (\mathbf{a}_1 + \mathbf{a}_2)\mathbf{x} - \mathbf{a}_1 \cdot \mathbf{a}_2 = 0$$

$$d. \mathbf{x}^2 + (\mathbf{a}_1 + \mathbf{a}_2)\mathbf{x} - \mathbf{a}_1 \cdot \mathbf{a}_2 = 0$$

Untuk penulisan pecahan digunakan [/pembilang/penyebut], sedangkan untuk vector dua dimensi digunakan [(vx(vy)]. **Pembilang** atau **vx** merupakan kalimat bagian atas, dan **penyebut** atau **vy** adalah kalimat yang ada dibagian bawah.

Contoh:

>Titik A([/ -1 /3], 2) didilatasikan [(-3 (4]
me_20nghasilkan

bentuk diatas akan menghasilkan

Titik A($\frac{-1}{3}, 2$) didilatasikan [$\begin{bmatrix} -3 \\ 4 \end{bmatrix}$

menghasilkan

Dari dua contoh diatas tampak tanda _**20** artinya jarak baris berikutnya adal 20 titik (dot). Yang harus diperhatikan adalah penulisan tanda itu berada mulai karakter ke empat (**4**) dan diikuti angka dua digit.

4.4 Mengatur penulisan baris berikutnya

Digunakan karakter _ (**garis bawah**) diikuti angka dua digit yang akan digunakan untuk menentukan penulisan baris berikutnya. Tanda tersebut harus ditulis pada karekter keempat (**4**).

4.5 Menulis dengan hurub lain atau menggunakan karakter font Simbol

Untuk memilih hurub lain pada suatu baris penulisan digunakan [fn...] , dengan titik-titik diisi nama font yang akan digunakan. Standart font pada program ini adalah **Time Roman**. Pengaturan nama font tersebut hanya berlaku pada karakter setelah tanda tanda itu ada dalam satu baris.

Sedangkan untuk menulis sebuah karakter dari font **Simbol** digunakan [\$.] dengan titik titik diisi karakter sebuah symbol atau [@..] dengan titik titik diisi nilai **ascii** dari karakter symbol tersebut. Daftar karakter symbol dapat dilihat pada lampiran.

Contoh:

>Jika [\$p] = 3.14 [@222] luas lingkaran
adalah

- a.[fnArial Black] ini arial black
- b. [fnArial] ini arial saja
- c. [fnArial narrow] kalau ini arial narrow
- d. ini asli times roman

Bentuk diatas akan menghasilakan penulisan sebagai berikut:

**Jika $\pi = 3.14 \Rightarrow$ luas lingkaran
adalah**

- a. ini arial black**
- b. ini arial saja**

- c. kalau ini arial narrow
- d. ini asli times roman

BAB V

PENUTUP

5.1 Mendistribusikan Soal

Setelah soal (contohnya **Soal_1.TXT**) ditulis dan berhasil dijalankan dengan **BUAT_SOAL.EXE** maka pada folder tempat soal_1.TXT akan muncul file:

- ISoal_1.TXT
- 1.TXT, 2.TXT, (sesuai jumlah gambar yang disertakan).

Diantara file tersebut yang siap dieksekusi program **SOAL.EXE** adalah file **ISoal_1.TXT**.

Untuk mendistribusikan hasil pembuatan soal tersebut ke sebuah folder baru yang harus disalin adalah:

- ISoal_1.TXT
- 1.TXT, 2.TXT, (sesuai jumlah gambar yang disertakan).
- File-file audio (*.MP3) yang ikut diperdengarkan.
- File-file video (*.MPG) yang ikut dimainkan.

5.2 Mengganti Logo Tutwuri dan Gambar Pengawas

Gambar Logo Tutwuri maupun gambar Pengawas yang muncul setiap menit selama 10 detik dapat dilakukan dengan menambah kode pada baris pertama.

Contoh:

Gunawan Susilo,111,0,105,120,225\157,167,275,288

Akan mengganti gambar pengawas dengan gambar pada daerah persegi (0,105,120,225) yang ada pada file 1.BMP. Sedang gambar Tut Wuri akan diganti dengan gambar pada daerah persegi (157,167,275,288) dari file 1.BMP.

Gambar tersebut dapat diganti salah satu saja, yaitu dengan membiarkan yang lain kosong. Contoh kita ingin mengganti pengawas saja maka dapat ditulis:

Gunawan Susilo,111,0,105,120,225

Atau mengganti logonya saja, ditulis:

Gunawan Susilo,111,157,167,275,288

Program ini masih banyak menyimpan kekurangan, sehingga dalam menggunakan harus hati-hati dan penuh kesabaran. Kekurangan akan sangat terasa adalah:

- Variabelnya terbatas
- Sistem operasi tidak lengkap
- Fung serta prosedurnya sangatlah minim
- Tidak disertai system pelacakan kesalahan sehingga kemungkinan computer bingung akan terjadi jika kita melakukan kesalahan menulis kode soal.
- Format penulisan masih jauh dari baik dan sangat tidak lengkap.

Penulis sangat berharap adanya saran maupun kritik dari pengguna yang dapat membagi rasa serta membangkitkan usaha memperbaiki program ini.

Akhirnya, kami ucapkan terima kasih kepada semua pihak yang langsung maupun tidak langsung telah mendorong terwujudnya program ini. Semoga program ini dapat berarti bagi dunia pendidikan khususnya. Amin.

LAMPIRAN

DAFTAR NILAI ASCII DAN KARAKTER SIMBOL

Desimal	Time Roman	Simbol	Desimal	Time Roman	Simbol
32			77	M	M
33	!	!	78	N	N
34	"	∇	79	O	O
35	#	#	80	P	Π
36	\$	∃	81	Q	⊙
37	%	%	82	R	P
38	&	&	83	S	Σ
39	'	ə	84	T	T
40	((85	U	Υ
41))	86	V	ς
42	*	*	87	W	Ω
43	+	+	88	X	Ξ
44	,	,	89	Y	Ψ
45	-	-	90	Z	Z
46	.	.	91	[[
47	/	/	92	\	∴
48	0	0	93]]
49	1	1	94	^	⊥
50	2	2	95	~	—
51	3	3	96		
52	4	4	97	a	α
53	5	5	98	b	β
54	6	6	99	c	χ
55	7	7	100	d	δ
56	8	8	101	e	ε
57	9	9	102	f	φ
58	:	:	103	g	γ
59	;	;	104	h	η
60	<	<	105	i	ι
61	=	=	106	j	φ
62	>	>	107	k	κ
63	?	?	108	l	λ
64	@	≡	109	m	μ
65	A	A	110	n	ν
66	B	B	111	o	ο
67	C	X	112	p	π
68	D	Δ	113	q	θ
69	E	E	114	r	ρ
70	F	Φ	115	s	σ
71	G	Γ	116	t	τ
72	H	H	117	u	υ
73	I	I	118	v	ϖ
74	J	ϑ	119	w	ω
75	K	K	120	x	ξ
76	L	Λ	121	y	ψ

Desimal	Time Roman	Simbol	Desimal	Time Roman	Simbol
122	z	ζ	173		
123	{	{	174	®	→
124			175	—	↓
125	}	}	176	°	°
126	~	~	177	±	±
127	□	□	178	²	"
128	€	□	179	³	≥
129	□	□	180	'	×
130	,	□	181	μ	∞
131	f	□	182	¶	∂
132	”	□	183	·	•
133	...	□	184	˙	÷
134	†	□	185	¹	≠
135	‡	□	186	°	≡
136	^	□	187	»	≈
137	%	□	188	¼	∴
138	Š	□	189	½	
139	<	□	190	¾	—
140	Œ	□	191	¿	⌋
141	□	□	192	À	℥
142	Ž	□	193	Á	₪
143	□	□	194	Â	₹
144	□	□	195	Ã	¢
145	‘	□	196	Ä	⊗
146	’	□	197	Å	⊕
147	“	□	198	Æ	⊖
148	”	□	199	Ç	⊔
149	•	□	200	È	∩
150	—	□	201	É	∪
151	—	□	202	Ê	⊃
152	~	□	203	Ë	⊂
153	™	□	204	Ì	⊆
154	š	□	205	Í	⊇
155	›	□	206	Î	∈
156	œ	□	207	Ï	∉
157	□	□	208	Ð	∠
158	ž	□	209	Ñ	∇
159	ÿ	□	210	Ò	®
160		□	211	Ó	©
161	ı	Υ	212	Ô	™
162	ϕ	’	213	Ö	∏
163	£	≤	214	Ö	√
164	₣	/	215	×	·
165	¥	∞	216	∅	⌊
166	ı	f	217	Ù	^
167	§	♣	218	Ú	v
168	¨	♦	219	Û	↔
169	©	♥	220	Ü	←
170	ª	♠	221	Ý	↑
171	«	↔	222	ƒ	⇒
172	¬	←	223	ß	⇓

Desimal	Time Roman	Simbol	Desimal	Time Roman	Simbol
224	à	◊	240	ð	◻
225	á	<	241	ñ	>
226	â	®	242	ò	¡
227	ã	©	243	ó	í
228	ä	™	244	ô	¦
229	å	∑	245	õ	
230	æ	/	246	ö	\
231	ç		247	÷	
232	è	\	248	ø)
233	é	┌	249	ù	┐
234	ê		250	ú	
235	ë	└	251	û	┘
236	ì	/	252	ü	\
237	í	}	253	ý	}
238	î		254	þ)
239	ï		255	ÿ	◻

