Sistem Bilangan dan Kesalahan

Penyajian Bilangan Bulat

Bilangan bulat yang sering digunakan adalah bilangan bulat dalam sistem bilangan desimal yang didefinisikan sbb:

$$N = \left(a_n a_{n-1} a_{n-2} \ldots a_0\right)$$

$$= a_n 10^n + a_{n-1} 10^{n-1} + a_{n-2} 10^{n-2} + \ldots + a_0 10^0$$

Contoh :

$$2673_{10} = 2*10^3 + 6*10^2 + 7*10^1 + 3*10^0$$

Bilangan bulat dengan bilangan dasar **c** didefinisikan sebagai :

$$N = \left(a_n a_{n-1} a_{n-2} \ldots a_0\right)_c$$
$$= a_n c^n + a_{n-1} c^{n-1} + a_{n-2} c^{c-2} + \ldots + a_0 c^0$$

ALGORITMA KONVERSI BILANGAN

Bila diketahui koefisien-koefisien $a_1, a_2, a_3, \ldots, a_n$ dari polinomial

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + a_{n-2} x^{n-2} + \ldots + a_0$$

dan suatu bilangan β

Maka dapat dihitung $b_n, b_{n-1}, .., b_0$ dari β sebagai berikut :

$$\boxed{\begin{aligned} b_n &= a_n \\ b_{n-1} &= a_{n-1} + b_n \beta \\ b_{n-2} &= a_{n-2} + b_{n-1} \beta \\ &\ldots\ldots\ldots\ldots\ldots\ldots \\ b_0 &= a_0 + b_1 \beta \end{aligned}}$$

Algoritma 1.1

Contoh:

Bilangan biner $(1101)_2$ dapat dihitung sebagai :

$$b_3 = a_3 = 1$$

$$b_2 = a_2 + b_3\beta = 1 + 1.2 = 3$$

$$b_1 = a_1 + b_2\beta = 0 + 3.2 = 6$$

$$b_0 = a_0 + b_1\beta = 1 + 6.2 = 13$$

sehingga $(1101)_2 = 13$

Penyajian Bilangan Pecahan

- Bilangan pecahan **x** antara 0 s/d 1 dalam sistem bilangan desimal didefinisikan :

$$x = (a_1 a_2 a_3 .. a_n) = a_1 10^{-1} + a_2 10^{-2} + a_3 10^{-3} + ... + a_n 10^{-n}$$

- Bilangan pecahan **x** secara umum dalam sistem bilangan dengan bilangan dasar ***k*** didefinisikan :

$$(a_1 a_2 a_3 ... a_n)_k = \sum_{i=1}^{n} a_i k^{-i}$$

Contoh:

a) $0{,}625_{10} = 6*10^{-1}+2*10^{-2}+5*10^{-3}$

b) $(0{,}101)_2 = 1*2^{-1}+0*2^{-2}+1*2^{-3}$

$= 0{,}5 + 0{,}125$

$= 0{,}625$

Algoritma 1.2

Jika diketahui x di antara 0 dan 1 sebuah bilangan β yang lebih besar dari 1. Carilah b_1, b_2, b_3,…

$$c_0 = x$$

$$b_1 = (\beta c_0)_I \qquad c_1 = (\beta c_0)_F$$

$$b_2 = (\beta c_1)_I \qquad c_2 = (\beta c_1)_F$$

…………………………..

Maka :

$$x = (b_1 b_2 b_3 \ldots)_\beta = \sum_{k=1}^{\infty} b_k \beta^{-k}$$

Contoh:

Jika x = $(0{,}101)_2$, dengan β = 10 (atau 1010_2), carilah b_1, b_2, b_3

$$c_0 = x = 0{,}101_2$$

$$\beta c_0 = (1010)(0{,}101)_2 = (110{,}010)_2 \text{ sehingga } b_1 = (110)_2 \quad c_1 = (0{,}01)_2$$

$$\beta c_1 = (1010)(0{,}01)_2 = (10{,}10)_2 \text{ sehingga } b_2 = (10)_2 \quad c_2 = (0{,}1)_2$$

$$\beta c_2 = (1010)(0{,}1)_2 = (101{,}0)_2 \text{ sehingga } b_3 = (101)_2 \quad c_1 = (0{,}01)_2$$

Karena b_k dst sama dengan nol, dan $b_1 = 6, b_2 = 2, b_3 = 5$
Maka :

$$x = (a_n a_{n-1} \ldots a_0, b_1 b_2 b_3 \ldots)_2 \quad = \quad x = (a_n a_{n-1} \ldots a_0, b_1 b_2 b_3 \ldots)_{10}$$

$$x = (0{,}101)_2 = (0{,}625)_{10}$$

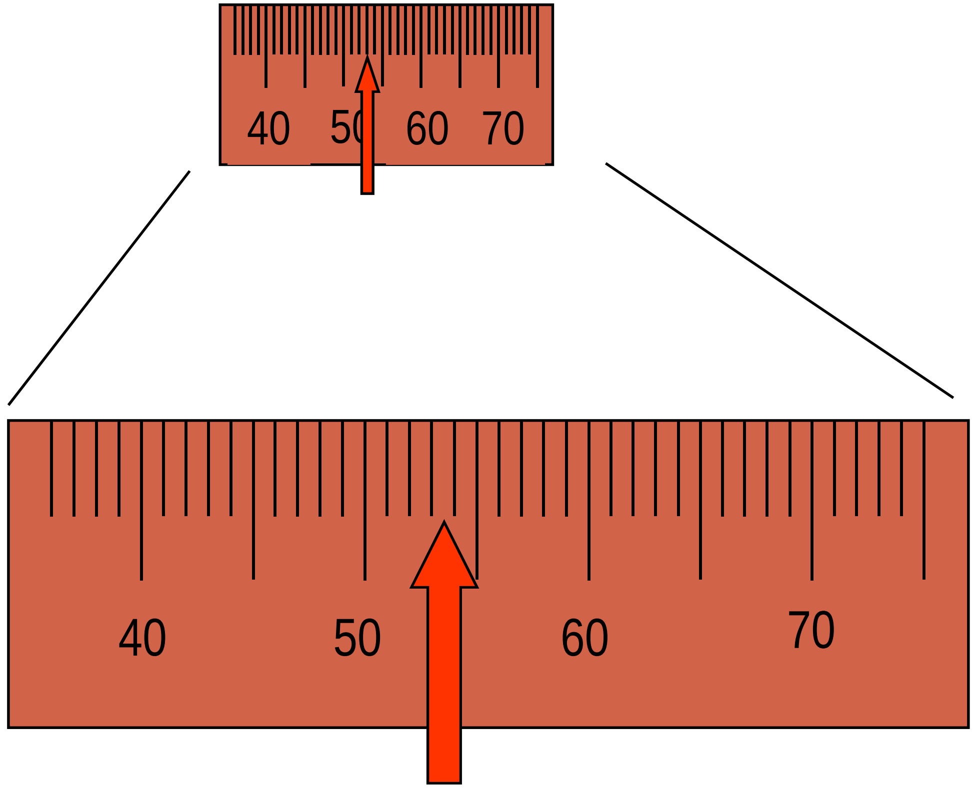
Nilai Signifikan / Angka Penting

Nilai signifikan ⇒ suatu nilai dimana *jumlah angka ditentukan sebagai batas nilai tersebut diterima atau tidak.*

Terdiri dari digit 1,2 3,4,5,6,7,8,9 dan 0

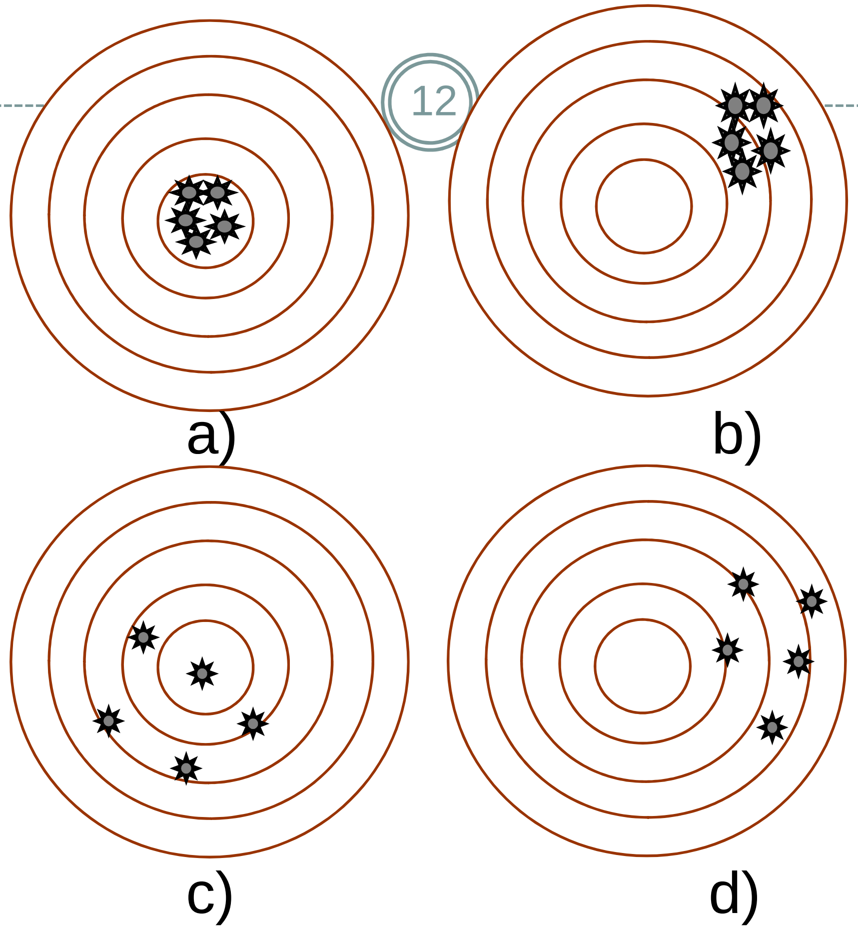
untuk 0 tidak termasuk angka signifikan jika digunakan untuk menentukan titik desimal atau untuk mengisi tempat-tempat dari digit yang tidak diketahui/dibuang.

Perhatikan nilai pada penggaris



Dengan nilai signifikan = 1, maka nilai adalah 53 atau 54
Dengan nilai signifikan = 0,1, maka nilai adalah 53 atau 53,5

Akurasi dan Presisi



(a) menunjukkan hasil yang akurat dan presisi.
(b) menunjukkan hasil yang presisi tetapi tidak akurat.
(c) menunjukkan hasil yang sebenarnya akurat tetapi tidak presisi.
(d) menunjukkan hasil yang tidak akurat dan tidak presisi.

Akurasi dan Presisi

Nilai presisi mengacu pada jumlah angka signifikan yang digunakan dan sebaran bacaan berulang pada alat ukur.

Pemakaian alat ukur penggaris dan jangka sorong akan mempunyai perbedaan nilai presisi.Pemakaian jangka sorong mempunyai presisi yang lebih tinggi.

Nilai akurat atau akurasi mengacu pada dekatnya nilai pendekatan yang dihasilkan dengan nilai acuan atau nilai eksak.

Misalkan nilai eksak diketahui ½, sedangkan hasil pendekatan adalah 0.500001 maka hasil ini dikatakan akurat bila torelansinya $=10^{-4}$.

Aturan Pembulatan

1. Jika bilangan yang dibuang kurang dari ½ satuan dari tempat yang ke –n, maka angka yang ke-n tetap tidak dirubah
2. Jika bilangan yang dibuang lebih dari ½ satuan dari tempat yang ke –n, maka angka yang ke-n ditambah dengan 1
3. Jika bilangan yang dibuang tepat ½ satuan dalam tempat yang ke-n, maka angka yang ke-n tidak dirubah, jika angka yang ke-n adalah genap atau angka yang ke-n ditambah dengan 1 (satu) jika angka yang ke-n gasal, dengan kata lain perkataan membulatkan sedemikian hingga angka yang ke-n adalah genap.

Jika suatu bilangan sudah dibulatkan menurut aturan di atas bilangan itu dikatakan betul (*correct*) sampai n angka yang berarti

Pendekatan dan Kesalahan

Kesalahan dalam Metode Numerik :

- Kesalahan pembulatan (*round of error*)

  Kesalahan pembulatan adalah kesalahan yang disebabkan oleh pembulatan.

  Contoh : 0,4 → 0 0,5 → 1

- Kesalahan pemotongan (*truncation error*)

  Kesalahan yang ditimbulkan pada saat dilakukan pengurangan jumlah angka signifikan.

  Contoh :

  0,4123568 → 0,41236

  (7 angka signifikan) (5 angka signifikan)

- Kesalahan dari perhitungan Numerik
  kesalahan yang timbul karena adanya proses pendekatan.

Hubungan kesalahan dan penyelesaian adalah :

$$\hat{x} = x + e$$

dimana:

$\hat{x}$ = nilai yang sebenarnya (nilai eksak)
x = nilai pendekatan yang dihasilkan dari metode numerik
e = kesalahan numerik.

Kesalahan fraksional adalah prosentase antara kesalahan dan nilai sebenarnya.

$$\in=\left(\frac{e}{\hat{x}}\right)x100\%$$

Kesalahan fraksional berdasarkan nilai pendekatan yang diperoleh: dimana e pada waktu ke n adalah selisih nilai pendekatan ke n dan ke n-1

$$\in=\left(\frac{x_n - x_{n-1}}{x_n}\right)\times 100\%$$

Perhitungan kesalahan semacam ini dilakukan untuk mencapai keadaan konvergensi pada suatu proses iterasi.

Definisi Konvergensi.

Suatu barisan $a_1, a_2, \ldots$ dikatakan konvergen ke α jika dan hanya jika untuk semua e>0 terdapat bilangan bulat $\eta_0(\in)$.

Sedemikian hingga untuk semua n≥ η_0 terdapat $|\alpha - \alpha_n| < \in$

Sehingga penyelesaian dalam metode numerik dicari berdasarkan <u>selisih hasil saat ini dengan hasil sebelumnya</u>.
Kriteria konvergensi ini dapat dipakai untuk mengurangi jumlah iterasi yang besar tetapi terkadang tidak akurat

Latihan soal :

1. Konversikan pecahan biner berikut ini menjadi pecahan desimal :
   $(0{,}110101)_2$ $(0{,}111111)_2$

2. Konversikan pecahan desimal berikut ini menjadi pecahan oktal
   $(0{,}1)_{10}$ $(0{,}86)_{10}$

3. Carilah angka biner yang mendekati π sampai 10^{-3}