Prikaz podataka u računalu

nastavak

Prikaz podataka u računalu

- Nakon odslušanog bit ćete u stanju:
  - objasniti prikaz realnih brojeva prema standardu IEEE
  - demonstrirati na konkretnim primjerima
  - objasniti prikaz realnih brojeva dvostruke preciznosti
  - ukazati na razlike u aritmetici između realnih brojeva u matematici i njihovog prikaza u računalu
  - predvidjeti moguće probleme.

Prikaz realnih brojeva u računalu – standard IEEE

- Binarni zapis realnog broja – normalizirani oblik:

$$broj = mantisa * 2^{eksponent}$$

Primjer:

$1010.11 = 1.01011 * 2^{3}$

$0.00010101 1 = 1.01011 * 2^{-4}$

- Realni brojevi u računalu se pohranjuju u normaliziranom obliku.
- Normalizirani oblik omogućava prikaz vrlo velikih i vrlo malih brojeva bez korištenja velikog broja nula.

Prikaz realnih brojeva u računalu (2)

31	30 … 23	22 … 0
p	karakteristika	mantisa bez skrivenog bita – 1

p – predznak:

- 0 – pozitivni broj
- 1 – negativni broj

k - karakteristika:

- binarni eksponent + 127
- $e \in \{-126, \ldots, 127\}$!
- $k \in \{0, \ldots, 255\}$!

Prikaz realnih brojeva u računalu (3)

- `k=0` i `k=255` – posebni slučajevi

<u>mantisa:</u>

- skriveni bit (prva jedinica) se ne pamti

- <u>Primjer:</u> Prikažite broj 10.25 kao realni broj.

$10.25_{10}=1010.01_{2}=1.01001*2^{3}$

$e=3,\ k=e+127_{10}=3_{10}+127_{10}=130_{10}=10000010_{2}$

0 10000010 01001000000000000000000

41240000_{16}

Posebni slučajevi

- Realni tip jednostruke preciznosti u programskom jeziku C: `float`

<u>Posebni slučajevi:</u>

- k = 0, svi bitovi mantise jednaki nula:

```
0000 0000 0000 0000 0000 0000 0000 0000 → +0
1000 0000 0000 0000 0000 0000 0000 0000 → -0
```

- Smatra se da su te vrijednosti jednake.

Posebni slučajevi (2)

- k = 0, postoji bit mantise različit od nule:

0000 0000 0101 1000 0000 0000 0000 0000 →

0.1011*2^{-126} (ne podrazumijeva se skriveni bit)

0000 0000 0000 0000 0000 0000 0000 1011 →

0.000 0000 0000 0000 0000 1011*2^{-126}

- k = 255, svi bitovi mantise jednaki nula:

0111 1111 1000 0000 0000 0000 0000 0000 → +∞

1111 1111 1000 0000 0000 0000 0000 0000 → −∞

Posebni slučajevi (3)

Primjer:

```
float x, y;
x = 2.f/0.f;  /* inf */
y = -2.f/0.f  /* -inf */
/* 2.f, 0.f – realne konstante */
```

Posebni slučajevi (4)

- k = 255, postoji bit mantise različit od nule:

`0111 1111 1000 0000 0000 0000 0100 1100` →

not **a** **n**umber

<u>Primjer:</u>

`float` x;

x = 0.f/0.f; /* nan */

Raspon realnih brojeva – tip `float`

- najmanji prikazivi pozitivni broj:

  $0.00000000000000000000001_2 * 2^{-126} \approx 1.4013e-45$

  $FLT\_MIN = 2^{-126} \approx 1.17549e-38$

- najveći prikazivi pozitivni broj:

  $1.11111111111111111111111_2 * 2^{127}$

  $FLT\_MAX \approx 3.40282e+38$

Realni brojevi dvostruke preciznosti

63	62 … 52	51 … 0
p	karakteristika	mantisa bez skrivenog bita – 1

- binarni eksponent `+ 1023`
- `e∈{-1022,…,1023}!`
- `k∈{0,…,2047}!`

- Realni tip dvostruke preciznosti u programskom jeziku C: `double`

Aritmetika realnih brojeva

- Aritmetika računala nije egzaktna.
- U njoj ne vrijede uobičajeni zakoni za operacije.

Na primjer, za zbrajanje i množenje u računalu **ne** vrijede

- asocijativnost,
- distributivnost.

Dakle, poredak izvršavanja operacija je bitan.

Aritmetika realnih brojeva (2)

Jedino standardno pravilo koje vrijedi je

- komutativnost za zbrajanje i množenje.

Primjer: neasocijativnost zbrajanja.

Parcijalna suma harmonijskog reda:

$$S_n := 1+1/2+1/3+\ldots+1/n.$$

$$S_{n,1} := (\ldots((1+1/2)+1/3)+\ldots+1/(n-1))+1/n.$$

$$S_{n,2} := 1+(1/2+(1/3+\ldots(1/(n-1)+1/n)\ldots)).$$

Aritmetika realnih brojeva (3)

- Zadatak: Za $n=1\ 000\ 000$, izračunajte u dvije standardne točnosti, $S_{n,1}$ (zbrajanje unaprijed) i $S_{n,2}$ (zbrajanje unatrag) te usporedite dobivene rezultate.

  Primjer: Uzmimo realnu aritmetiku „računala” u bazi 10. Pretpostavimo da za mantisu imamo $t=4$ znamenke, a za eksponent $s=2$ znamenke. Neka je

$$x=8.8866=8.8866\times 10^0,$$

$$y=8.8844=8.8844\times 10^0,$$

Aritmetika realnih brojeva (4)

Umjesto brojeva x i y (koji nisu prikazivi), u „memoriju" spremamo brojeve $fl(x)$ i $fl(y)$, pravilno zaokružene na $t=4$ znamenke

$$fl(x)=8.887\times 10^0,$$

$$fl(y)=8.884\times 10^0.$$

Razliku računamo tako da izjednačimo eksponente (već jesu), oduzmemo značajne dijelove (mantise), pa normaliziramo.

$$fl(x)-fl(y)=8.887\times 10^0-8.884\times 10^0$$

$$=0.003\times 10^0=3.???\times 10^{-3}.$$

Aritmetika realnih brojeva (5)

- ? = znamenke koje više ne možemo restaurirati (ta informacija se izgubila)

Konačni rezultat je $\mathtt{fl(x)-fl(y)}=3.000\times10^{-3}$.
Pravi rezultat je

$$x-y=8.8866\times10^{0}-8.8844\times10^{0}$$
$$=0.0022\times10^{0}=2.2\times10^{-3}.$$

Već prva značajna znamenka je pogrešna, a relativna greška je ogromna! Uočite da je oduzimanje bilo $\mathtt{fl(x)-fl(y)}$ egzaktno (i u aritmetici računala), ali rezultat je pogrešan. Uzrok su polazne greške u operandima.