

Otázka: Zálohování dat

Předmět: Informatika

Přidal(a): vlčák

Michal Vlček

Zobrazení údajů v počítači

ASCII (American Standard Code for Information Interchange)

-kódová tabulka, která definuje znaky anglické abecedy, a znaky používané v informatice.

-nejúspěšnější znaková sada, z které vychází většina současných standardů pro kódování textu

-Tabulka obsahuje znaky:

*písmena

*číslice

*jiné znaky (závorky, matematické znaky (+ - * / % ,...), interpunkční znaménka (, . : ; atd.)

*speciální znaky (@ \$ ~ atd.))

*řídící (netisknutelné) kódy(prvních 32 znaků), které byly původně určeny pro řízení periferních zařízení (např. tiskárny nebo dálnopisu).

Kód ASCII je sedmibitový (obsahuje 128 platných znaků).

Rozšířená verze je 8bitová.

Zabezpečení

Chybové kódy: Ochrana dat při přenosu, nejjednodušší s paritním bitem. **Paritní kódy:** využívají 1 paritní bit pro kontrolu dat při přenosu. **Existuje je i podélná a příčná parita** pro lepší zabezpečení.

-**Sudá parita:** celkový počet jedniček je sudý

Data	Parita
------	--------

00	0
----	---

01	1
----	---

10	1
----	---

11	0
----	---

-**Lichá parita** celkový počet jedniček je lichý

Data	Parita
------	--------

00	1
----	---

01	0
----	---

10	0
----	---

Unicode

tabulka znaků všech existujících abeced

obsahuje více než 110 000 znaků

základní způsoby kódování Unicodu:

UTF-32 každý znak reprezentován přímo 32bitovým číslem

UTF-16 znaky BMP reprezentují jedním 16bitovým číslem, znaky mimo BMP jsou reprezentovány

párem 16bitových čísel

UTF-8 znaky se kódují různě dlouhou (1–6 bajty) posloupností bajtů podle jejich pozice v Unicode

UCS-2 původní kódování Unicodu, pro uložení využívá posloupnosti dvoubajtových částí.

Nejpoužívanější číselné soustavy: (váhy začínají od 0)

Dekadická(desítková)

Značí se indexem D nebo 10

Poziční čís. soustava se základem 10

Pro zápis používá symboly 0,1,...,9

Používaná v běžném životě

čísla zapsaná v dekadické soustavě např: 9,15,35,256

mocninový rozvoj čísla v dané soustavě:

$$2546_{10} = 2 \cdot 10^3 + 5 \cdot 10^2 + 4 \cdot 10^1 + 6 \cdot 10^0$$

$$0.125_{10} = 0 \cdot 10^0 + 1 \cdot 10^{-1} + 5 \cdot 10^{-3}$$

Binární(dvojková)

Značí se indexem B nebo 2

využívá se v moderních dig. Počítačích

základem je 2

má pouze dva symboly 0 a 1.

poziční číselná soustava mocnin čísla 2

číslo zapsané v binární soustavě 1101101

mocninový rozvoj čísla v dané soustavě:

$$100111110010_2 = 1 \cdot 2^{11} + 0 \cdot 2^{10} + 0 \cdot 2^9 + 1 \cdot 2^8 + 1 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0$$

Hexadecimální(šestnáctková)

Jednoduchý vzájemný převod mezi šestnáctkovou a dvojkovou soustavou a proto se používá v informatice

Značí se indexem H nebo 16

Základem je H neboli 16

Př. pro adresy v operační paměti počítače.

Pro zápis používá symboly 0,1,...,9,A,B,C,D,E,F

číslo zapsané v hexadecimální soustavě 5F0A

mocninový rozvoj čísla v dané soustavě:

$$9F2_{16} = 9 \cdot 16^2 + 15 \cdot 16^1 + 2 \cdot 16^0$$

BCD kód: binárně zakódované dekadické číslo, čtyřbitový binární kód (váhy 8,4,2,1)

D	0	1	2	3	4	5	6	7	8	9
BCD	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

Další kombinace na 4 bitech (10-15) jsou mimokódové (Chyby). Výhoda je snadný převod mezi BCD a Dekadickou soustavou.

Pevná řádová čárka: vyjadřuje absolutní přesnost. Odděluje desetinná místa od celých. Při velkém počtu cifer nebo míst je velmi nevýhodná

Plovoucí řádová čárka: vyjadřuje relativní přesnost. S čárkou lze hýbat podle potřeb (omezeno normami). Při zobrazování a převádění se využívá semilogaritmického tvaru (0,0015 v semilogaritmickém tvaru $15 \cdot 10^{-4}$).

Formáty s pohyblivou řádovou čárkou

Short Real – Pro většinu běžně používaných čísel. Přesnost 24b

Long Real – Pro technické a vědecké účely, vysoká přesnost (53b)

Temporary Real: pracovní formát FPU, velmi velký rozsah a přesnost (64b).

Znaménko

Osmibitový přímý binární kód. Maximální zobrazené číslo je $11111111 = \pm 127D$.

1. Bit zleva vyhrazen jako znaménko (0= +, 1= -)

Př: $+45D = 0\ 0101101B$

- $45D = 1\ 0101101B$

Pokud chci kladné číslo převést na záporné, tak ho nejdříve zneguji a pak přičtu 1.