

10. Číselné soustavy

Využití číselných soustav

Číselná soustava je prostý způsob reprezentace čísel, soustavy rozlišujeme podle toho, kolik můžeme použít znaků pro jednu cifru (př. Desítková- deset, Dvojková – dva).

Mezi hlavní využití patří zejména využití v IT

Dvojková soustava – nejvhodnější, počítači stačí znát pouze dva stavy(0 a 1;vypnuto, zapnuto)

Dále se v IT využívá např. šestnáctková (MAC adresy, IPv6)

Další využití např. Šedesátková (úhly, čas)

Funkčnost

Každá soustava, pokud překročí svůj nejvyšší znak přesouvá se první znak do vyššího řádu.

Hodnota na každém řádu je rovna($c \cdot \text{základ}^{\text{číslo řádu}}$) – polyadický zápis

U všech soustav fungují všechny operace a převody na stejném principu

Převody mezi číselnými soustavami

1. způsob - Pomocí polyadického zápisu čísla

Vhodné pro převod do desítkové

2. způsob – Dělení základem soustavy (Pro převod z desítkové soustavy)

Dělíme základem soustavy a sepisujeme zbytek po dělení, výsledkem jsou zbytky seřazeny odzadu.

U čísel desetinných musíme násobit základem soustavy

Pro člověka zdlouhavé, vhodné pro výpočetní techniku

3. způsob - Pomocí postupného dělení čísla mocninou základu (Převod z desítkové)

Hledáme nejvyšší mocninu, která se vejde do čísla a poté dělíme postupně mocninou základu až když je mocnina 0.

4. způsob – Bitová mřížka

Pokud převádíme ze soustav, kde jsou jejich základy navzájem mocniny můžeme si číslo rozdělit na části, podle toho jak velký jsou mocniny (mocnina 4=po 4 bitech...)

Aritmetické operace v různých číselných soustavách

Sčítání

Provádí se ve všech soustavách stejně, sčítají se číslice v každém řádu, pokud součet je roven nebo větší než základ soustavy, napíše se o kolik číslo přesáhlo soustavu a k dalšímu sčítání se přičte zbytek, který vyšel(kolikrát přeskočil základ soustavy).

Odčítání

1. způsob

Každé číslice v jednotlivých řádech se odečtou, pokud odečítáme větší číslo od menšího => sepíšeme výsledek, a při dalším odečtu musíme odečíst ještě zbytek.

2. Způsob sčítáním pomocí dvojkového doplňku

Nejprve inverze odčítaného čísla ($0=1$ a $1=0$), poté přičtení jedničky k odčítanému číslu a následně sečteme s číslem, od kterého odečítáme.

Násobení

1. Způsob

Vynásobíme každou číslici každou, pokud přesáhneme základ soustavy přičítáme k dalšímu násobení zbytek, po každém vynásobení prvního čísla posouváme další násobení o jednu číslici doleva, nakonec všechny vynásobená čísla sečteme.

2. Způsob

Sčítáme stejné číslo tolikrát kolikrát ho chceme vynásobit, pro člověka zdlouhavé, vhodnější pro počítač

Dělení

Označíme si největší číslo, do kterého se vejde dělitel, sepíšeme kolikrát se do čísla vejde a toto číslo vynásobíme s dělitelem a odečteme od označeného čísla a pokračujeme, dokud nejsme na konci čísla a zbytek je 0.

polyadický zápis čísla

Zápis čísla, kde každá hodnota na jednotlivém řádu je rovna $c \cdot \text{základ}^{\text{číslo řádu}}$, a mezi všemi těmito výrazy je sčítání, pokud čísla sečteme měli by jsme dostat původní číslo. Využívá se při převodu čísel do desítkové soustavy.

$$a = a_n z^n + a_{n-1} z^{n-1} + \dots + a_2 z^2 + a_1 z^1 + a_0 z^0$$

Definice kódu

Kód je předpis pro zobrazení dat do jiné reprezentace, využívá se pro zrychlení, zkrácení zápisu a čtení informací a ke kompresi dat.

$$Z = \Phi(I)$$

I – Vstupní množina informací, které máme zobrazit

Z – Výstupní množina informací

Φ – Předpis podle kterého přiřazujeme prvky

příklady a využití kódování

BCD kód – kód, který desítkové číslo rozdělí na číslice 0 až 9 a převádí každou číslici zvlášť podle dvojkové soustavy

Příklad: 3458987 = 0011 0100 0101 1000 1001 1000 0111

ASCII kód – Definuje velké množství znaků jako binární číslo

Grayův kód

Kód spočívá v tom, že každý následující číslo se smí lišit pouze v jednom bitu.

Původně navržen pro zabránění hazardu spínacím relé, dnes se používá pro opravu chyb v digitální komunikaci, pro některé snímače poloh, pro Kargnauhovu mapu.

	x ₄	x ₃	x ₂	x ₁
0	0	0	1	1
1	0	1	0	0
2	0	1	0	1
3	0	1	1	0
4	0	1	1	1
5	1	0	0	0
6	1	0	0	1
7	1	0	1	0
8	1	0	1	1
9	1	1	0	0
10	1	1	0	1
11	1	1	1	0
12	1	1	1	1
13	0	0	0	0
14	0	0	0	1
15	0	0	1	0

zobrazení dat v počítači

Zobrazení celých čísel

Bez znaménka – nemusíme řešit záporná čísla, klasické zobrazení celých čísel(128(10))

Se znaménkem – musíme řešit znaménko 3 způsoby

1. způsob – přímý kód

První bit určuje znaménko (0=kladné, 1=záporné)

2. způsob – inverzní kód

Inverze všech bitů

3. způsob – dvojkový doplněk

Inverze všech bitů a přičtení k číslu 1, tím se odstraní kladná a záporná nula.

Zobrazení desetinných čísel

S pevnou desetinnou čárkou – klasické zobrazení (1,54564; 2,254545)

Nevýhodou je, že musíme vědět kolik řádů chceme zobrazit

S plovoucí desetinnou čárkou

$C=m \cdot Z^e$ C=číslo, m= mantisa , Z=základ soustavy e=exponent

Příklad: $1,602 \cdot 10^{-19}$

Normalizovaný tvar – Vždy před desetinnou čárkou je jednička

Sčítání, odčítání - upravíme na společný exponent a sčítáme/odčítáme mantisy

Násobení, dělení – vydělíme/vynásobíme mantisy a sečteme/odečteme exponenty

První bit mantisy je vždy jednička – u nenulových čísel není potřeba

Zobrazení textu

Text je zobrazen pomocí různých znakových sad

ASCII – 8 bitová sada zobrazuje nejzákladnější znaky

Unicode – Spousta verzí, zahrnuje většinu znaků

Používají se různá kódování UTF-8 , UTF-16