

Pozor na čísla

celá čísla

reálná čísla

jejich zobrazení v počítači

Reálná a celá čísla

- zpracovávané informace v počítači
 - bitové
 - znakové
 - textové
 - **číselné**
 - reálná čísla
 - celá čísla
- v programech rozlišujeme reálná a celá čísla
- běžné programovací jazyky nabízejí pro tyto dvě kategorie čísel předdefinované datové typy
 - celočíselný datový typ
 - reálný datový typ (s pohyblivou desetinnou čárkou)

Reálná a celá čísla

- důvody:
 - odlišný způsob vnitřní reprezentace v počítači
 - odlišný způsob provádění výpočtů
 - celá čísla se ukládají **přesně**
 - práce s celými čísly je jednodušší a **rychlejší**
 - u reálných čísel dejte vždy pozor na **zaokrouhlovací chyby**
- vnitřní reprezentace v počítači
 - záleží na konkrétním počítači
 - záleží na použitém překladači
 - **ALE obecné zásady jsou vždy stejné**

Celá čísla

přesný tvar reprezentace čísel v počítači závisí na dvou věcech

1. na velikosti paměťového prostoru, který se pro jedno číslo vyhradí
 2. na způsobu využití přiděleného paměťového prostoru
- kladná celá čísla jsou v počítači zobrazena ve dvojkové soustavě
 - záporná celá čísla jsou v počítači obvykle zobrazena v tzv. doplňkovém kódu
 - nejvyšší bit slouží k uložení znaménka, 0 číslo kladné, 1 číslo záporné
 - vlastní hodnota čísla je uložena ve zbývajících bitech
 - díky paměťovému omezení není možné uložit v počítači celé číslo libovolné velikosti, ale vždy jen celé číslo z určitého, omezeného intervalu

Celá čísla

- pokud programovací jazyk používá pro uložení kladného celého čísla např. 16 bitů, pak do tak velkého paměťového prostoru lze uložit číslo o dané maximální velikosti
- to může být omezující
 - pak je potřeba použít jiný datový typ
 - nebo si programově vytvořit vlastní datový typ
 - číslo je možné rozdělit po určitém počtu cifer a rozdělené uložit
- při výpočtech s celými čísly je potřeba dát **pozor na dělení**
 - **použít prostředky pro celočíselné dělení**
- pokud provádíme výpočty s celými čísly, je dobré je neopouštět
 - přechod na reálná čísla znamená zpomalení výpočtu
 - a také hrozí nebezpečí vzniku zaokrouhlovací chyby

Reálná čísla

- reálný datový typ (datový typ pro zobrazení čísel v pohyblivé řádové čárce) slouží pro práci s racionálními čísli
- omezení:
 - předem omezený počet platných cifer
 - omezený rozsah přípustných hodnot
- paměťový prostor pro uložení číselné hodnoty reálného čísla je rozdělen
 - jeden bit na znaménko
 - pevně daný počet bitů pro uložení hodnoty exponentu (exponent udává o kolik míst je potřeba posunout desetinnou čárku), je-li hodnota exponentu kladná, pak je zobrazované číslo v absolutní hodnotě větší než 1
 - pevně daný počet bitů pro uložení mantisy, mantisa je uložena v tzv. normalizovaném tvaru, kdy desetinná čárka je umístěna před řádem mantisy, který je nenulový

Reálná čísla

- souhrnně
 - je třeba si pamatovat, že omezený počet bitů pro zobrazení mantisy určuje **maximální počet platných cifer**
 - a že omezený prostor pro zobrazení exponentu určuje **rozsah hodnot**, které lze uložit

Reálná čísla

- příklad
 - minimální velikost paměti pro uložení čísla v pohyblivé desetinné čárce bývá 4 byty
 - z toho je obvykle 24 bitů určeno pro mantisu, včetně znaménka
 - 8 bitů je určeno pro exponent
- mantisa
 - hodnota mantisy se uloží do 23 bitů, dvě nejbližší, po sobě jdoucí čísla se liší v hodnotě nejnižšího bitu, který má oproti nejvyššímu bitu váhu o 2^{-22} nižší, desítkově asi tak $2,38 \cdot 10^{-7}$, z toho plyne přesnost zobrazení na 6 až 7 platných cifer
- exponent
 - pokud je pro exponent vyhrazeno 8 bitů, pak to představuje posun desetinné čárky v rozmezí -2^7 až 2^7-1 , tj. -128 až 127, zobrazitelná čísla leží v rozmezí 2^{-128} až 2^{127} , desítkově 10^{-38} až 10^{38} .

Reálná a celá čísla

- poučení
 - pokud provádíme výpočty s celými čísly, neopouštět obor celých čísel
 - při výpočtech s reálnými čísly musíme mít neustále na paměti možnost vzniku a kumulace zaokrouhlovacích chyb, které mohou znehodnotit i správné algoritmy
 - některé jevy, které mohou nastat, lze ošetřit pomocí např. **OverflowException**