

Datové typy

Celočíselné typy
Typy s plovoucí řádovou čárkou
Booleovský typ
Znakový typ

Základní pojmy

- **příkaz** je povel, který provede určitou akci
- příkazy jsou uvedeny uvnitř metod, např. **Main**
- příkazy musí splňovat sadu pravidel, tzv. **syntaxi** (např. v C# musí být příkazy ukončeny středníkem)
- **sémantika** jazyka vysvětluje význam příkazů, říká, co dělají
- **identifikátory** se používají pro pojmenování jednotlivých částí programu
- **klíčová slova** jsou identifikátory, které má jazyk C# vyhrazeny, např. **class**
- **proměnná** je místo, kde je uložena hodnota, každá proměnná má jedinečné jméno, tímto jménem se odkazujeme na její hodnotu

Datové typy

- datový typ proměnné určuje, jak je proměnná uložena v paměti a jaké operace lze s proměnnou provádět
- datové typy můžeme rozdělit na:
 - hodnotové typy (**value types**)
 - odkazové (referenční) typy (**reference types**)
- proměnná hodnotového typu má svou hodnotu uloženu přímo v paměti, zásobníku (**stack**):
 - typickým příkladem hodnotového typu je typ **int** pro celočíselné proměnné
- proměnná odkazového (referenčního) typu má uložen pouze odkaz (referenci) na skutečnou hodnotu
 - Abychom mohli s takovou proměnnou pracovat, musí existovat i odkazovaná hodnota. Tyto hodnoty jsou ukládány do jiné oblasti paměti, která se nazývá halda (**heap**)
 - Naprostou většinu datových typů tvoří odkazové typy.

ÚPŘT VŠCHT Praha, Jana Finkeová

Celočíselné typy

Název	Typ	Popis	Rozsah	Velikost
sbyte	System.SByte	8-bitové číslo se znaménkem	-128 až 127 (-2^7 až 2^7-1)	1 B
short	System.Int16	16-bitové číslo se znaménkem	-32 768 až 32 767 (-2^{15} až $2^{15}-1$)	2 B
int	System.Int32	32-bitové číslo se znaménkem	-2 147 483 648 až 2 147 483 647 (-2^{31} až $2^{31}-1$)	4 B
long	System.Int64	64-bitové číslo se znaménkem	-9 223 372 036 854 775 808 až 9 223 372 036 854 775 807 (-2^{63} až $2^{63}-1$)	8 B

ÚPŘT VŠCHT Praha, Jana Finkeová

Celočíselné typy

Název	Typ CTS	Popis	Rozsah	Velikost
byte	System.Byte	8-bitové číslo bez znaménka	0 až 255 (0 až 2^8-1)	1 B
ushort	System.UInt16	16-bitové číslo bez znaménka	0 až 65 535 (0 až $2^{16}-1$)	2 B
uint	System.UInt32	32-bitové číslo bez znaménka	0 až 4 294 967 295 (0 až $2^{32}-1$)	4 B
ulong	System.UInt64	64-bitové číslo bez znaménka	0 až 18 446 744 073 709 551 615 (0 až $2^{64}-1$)	8 B

ÚPŘT VŠCHT Praha, Jana Finkeová

Celočíselné typy

- celočíselný datový typ volte vhodně s ohledem na velikost čísel, která budete v proměnné ukládat
- celočíselná proměnná se chová jako **počítadlo** (např. kilometrů)
- jestliže je v proměnné již maximální hodnota a přičtete jedničku, rázem se dostanete na zcela nejnižší hodnotu,
- podobně odečtení jedničky od nejnižší hodnoty...

ÚPŘT VŠCHT Praha, Jana Finkeová

Celočíselné typy

- datový typ **int** je nejvhodnější pro výpočty, je 32-bitový a na 32-bitových systémech jsou operace s typem **int** nejefektivnější
- důvodem pro použití jiného typu pro celá čísla než typu **int** může být
 - malý rozsah (v tom případě použijeme typ **long**)
 - zbytečně velký rozsah (chceme šetřit paměť např. v případě mnoha celočíselných proměnných v poli – lze užít menší celočíselný typ **short**)

Příklady deklarací:

```
int i = 123;
int cislo;
short z = 5;
long velkeCislo;
```

Konstanta:

```
const int x = 0;
```

ÚPŘT VŠCHT Praha, Jana Finkeová

Typy s plovoucí řádovou čárkou

- datový typ s plovoucí řádovou (desetinnou) čárkou je v paměti realizován odlišně než celočíselné datové typy
- ukládá se zvlášť
 - mantisa
 - exponent
- např. číslo 213,456543 a číslo 213456,543 jsou v podstatě stejná až na řád
 - první můžeme zapsat jako $2,13456543 \cdot 10^2$, druhé $2,13456543 \cdot 10^5$.
 - obě čísla mají stejnou přesnost (9 platných číslic), u prvního je řád 2 (stovky), u druhého 5 (statisíce)

ÚPŘT VŠCHT Praha, Jana Finkeová

Typy s plovoucí řádovou čárkou

Název	Typ	Popis	Počet platných číslic	Rozsah (zaokrouhleno)	Velikost
float	System.Single	32-bitové číslo s jednoduchou (single) přesností s plovoucí řádovou čárkou	7	$+(-) 1.5 \cdot 10^{-45}$ až $+(-) 3.4 \cdot 10^{38}$	4 B
double	System.Double	64-bitové číslo s dvojitou (double) přesností s plovoucí řádovou čárkou	15 - 16	$+(-) 5.0 \cdot 10^{-324}$ až $+(-) 1.7 \cdot 10^{308}$	8 B

```
double w = 1.7E+3;  
const double gravitationalConstant = 6.673e-11;  
float y ;
```

Desítkový typ

- typ v plovoucí řádové čárce
- **desítkový typ (decimal)** větší přesnost a menší rozsah (finanční výpočty)

Desítkový typ

Název	Typ	Popis	Počet platných číslic	Rozsah (zaokrouhleno)	Velikost
decimal	System.Decimal	128-bitové číslo s jednoduchou přesností s plovoucí řádovou čárkou	28	$+(-) 1.0 \cdot 10^{-28}$ až $+(-) 7.9 \cdot 10^{28}$	16 B

ÚPŘT VŠCHT Praha, Jana Finkeová

Booleovský typ

- booleovský typ (**boolean**) slouží pro reprezentaci hodnot ve dvouhodnotové logice
- může nabývat dvou hodnot:
 - **true** a **false**
 - tyto hodnoty obvykle reprezentují, zda je určitá podmínka (nebo stav) pravdivá (true) či nepravdivá (false)

ÚPŘT VŠCHT Praha, Jana Finkeová

Booleovský typ

Název	Typ	Popis	Hodnoty
bool	System.Boolean	Reprezentuje boolevskou hodnotu	true nebo false

ÚPŘT VŠCHT Praha, Jana Finkeová

Znakový typ

- **znakový typ** (**char**) slouží pro ukládání jedné znakové hodnoty, např. písmena
- pracuje v kódování Unicode, může tedy obsahovat i znaky národních abeced
- literální (doslovné) hodnoty udáváme v apostrofech, např. 'A'.

ÚPŘT VŠCHT Praha, Jana Finkeová

Znakový typ

Název	Typ CTS	Popis	Hodnoty
char	System.Char	Reprezentuje 16-bitový znak v Unicode	Libovolný znak Unicode

ÚPŘT VŠCHT Praha, Jana Finkeová

Vyhledávání v nápovědě

- Value Types (C# Reference)
- Default Values Table (C# Reference)
- Integral Types Table (C# Reference)
- Floating-Point Types Table (C# Reference)
- decimal (C# Reference)
- bool (C# Reference)

ÚPŘT VŠCHT Praha, Jana Finkeová

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**

ÚPŘT VŠCHT Praha, Jana Finkeová

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**

ÚPŘT VŠCHT Praha, Jana Finkeová

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

ÚPŘT VŠCHT Praha, Jana Finkeová

Reálná čísla

- reálný datový typ (datový typ pro zobrazení čísel v pohyblivé řádové čárce) slouží pro práci s racionálními čísly
- 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

ÚPŘT VŠCHT Praha, Jana Finkeová

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

ÚPŘT VŠCHT Praha, Jana Finkeová

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} .

ÚPŘT VŠCHT Praha, Jana Finkeová

Reálná a celá čísla

- poučení
 - pokud provádíme výpočty s celými čísly, neopouštět obor celých čísel, jinak hrozí
 - zpomalení výpočtů
 - zaokrouhlovací chyby
 - chyby při dělení
 - 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**

ÚPŘT VŠCHT Praha, Jana Finkeová

**Příště: klíčová slova,
identifikátory, operátory**