

# 1- Základní programové konstrukce

## Proměnné

- Proměnná je místo v paměti, kam se ukládají data
- Je to označení pro identifikátor, který uchovává určitou hodnotu při běhu programu
- `int x` - pouze deklarace
- `x = 2000` - inicializace
- `int y = 5000` - deklarace s inicializací

## Datové typy, reference a hodnota

- Zásobník (stack) – systémem přidělená rychlá malá paměť, která je využívána k ukládání lokálních proměnných hodnotového typu a referencí
- Halda – pomalejší paměť, který obsahuje referenční datové typy

## Hodnotové datové typy

- Hodnota je uložena v zásobníku a program s ní pracuje přímo
- Mají pevnou velikost
- Lze dělit na základní typy, struktury (struct) a výčtové typy (enum)
- `int`, `float`, `char`, `bool`, `struct`, `enum`

## Referenční datové typy

- V zásobníku je odkaz na haldu
- Na jeden objekt může být více referencí a můžeme vytvářet objekty s dynamickou velikostí
- Garbage collector – maže data z haldy, které nemají žádnou referenci na zásobníku
- `object`, `class`, `interface`, `array`, `delegate`, `List<>`
- `string` – `immutable`, mohou být velké => je lepší je uložit na haldu

## Programy a podprogramy

- Program: posloupnost instrukcí popisující realizaci dané úlohy počítačem
- Podprogram:
  - Část programu
  - Lze opakovaně použít kdekoli v programu (i v podprogramu)
  - Může mít vstupní parametry
  - Může mít návratovou hodnotu
  - V C# metoda
  - Obecně funkce vrací hodnotu

## Operátory + priorita operátorů

Kategorie	Operátor
primární	x.y f(x) a[i] x?.y new typeof checked
unární	+x -x !x ~x ++x --x ^x await true false
rozsah	x..y
switch a with	switch with
multiplikativní	x*y x/y x%y
aditivní	x+y x-y
bitové posuny	x<<y x>>y
relační	x<y x>y x<=y x>=y is as
rovnost	x==y x!=y
bit AND	&
bit XOR	^
bit OR	
AND	&&
OR	
ternární	?
přiřazení a lambda	= += -= *= /= =>

## Převody

- Implicitní – hodnota se převádí na typ s větším datovým rozsahem (int na double)  

```
int myInt = 9;  
double myDouble = myInt;
```
- Explicitní – hodnota se převádí na typ s menším datovým rozsahem (double na int), může dojít ke ztrátě dat  

```
int myInt = (int) myDouble;
```