# 13 - Testování softwaru

* Kompilovaný jazyk – Zdrojový kód nejprve přeložen pomocí překladače (kompilátoru) do strojového kódu a až poté je možné ho spustit (C#)
* Interpretovaný jazyk – Program zvaný interpreter přímo spouští kód step-by-step bez nutnosti předběžného překladu (Python)
* Hardwarové chyby – Chyby způsobené selháním hardwaru
* Uživatelské chyby – Chyby způsobené nekorektními vstupy od uživatele
  + Některé může programátor snadno ošetřit, např. když uživatel do kolonky hmotnost zadá textový řetězec namísto čísla
  + Na jiné se reaguje složitěji, např. když uživatel do kolonky hmotnost zadá hodnotu v librách namísto kilogramů
* Programátorské chyby – Chyby způsobené nesprávně napsaným kódem programu

# Programátorské chyby

## Syntaktická chyba

* Syntax – Symbolová reprezentace
* Syntaktická chyba – Kód neodpovídá pravidlům zápisu daného jazyka
* U kompilovaných jazyků je hlášena během překladu, většina IDE na chybu upozorní, popřípadě nabídne možné opravy
* Chyby v zápisu klíčového slova (překlepy) se občas nazývají lexikální

## Sémantická chyba

* Program dělá něco, co vlastně nechceme
* Na první pohled často neviditelná chyba
* Nemusí se vždy projevit
* Kód nemusí ukazovat žádné chyby při kompilaci

## Run-time chyba

* Objevují se po spuštění kódu
* Například nelze nalézt uvedený soubor

# Debugging

* Hledání a oprava chyb
* Oprava překlepů (syntax) či chybných výsledků (sémantika)
* Sémantické chyby nemusí být na první pohled zjevné, s nalezením jejich příčin mohou pomoci nástroje pro ladění
* Post-Mortem: zkoumání stavu aplikace po tom, co spadla

## Breakpoint

* Zastavení běhu aplikace na daném řádku

## Krokovací nástroje

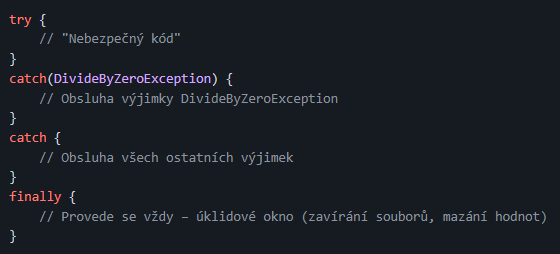
* Lze pomocí nich postupovat v kódu řádek po řádku

## Logování

* Logování buď výpisem do souboru nebo do IDE logovací konzole

## Výjimka

* Softwarové přerušení, nastává při běhu programu již po kompilaci a přeložení
* Standardní způsob předávání informací o chybě nebo nestandardním stavu
* Objekt nesoucí informaci o chybovém stavu
* V dobrém programu by měly být všechny výjimky ošetřeny
* Výjimku lze vyvolat operátorem throw, např. pro upozornění testera když část programu není doimplementována
* throw new DivideByZeroException("vlastní zpráva");
* Je možné si definovat i vlastní výjimky, všechny dědí z System.Exception



## Jednotkové testy

* Rozdělení kódu na nezávislé jednotky (třída, metoda, proměnná)
* Testování funkčnosti jednotky