14. Teoretické modely počítačů

# Generace

* Nedostatky prvních počítačů (počítačů 0. generace):
  + Malý výkon
  + Složitost
  + Programování pomocí drátových propojek naprogramování a spuštění jedné úlohy trvalo několik hodin i několik dní)
* 1. generace 1940-1956
  + Elektromechanické nebo elektronkové
  + byly velké, těžké, spotřebovávaly velké množství energie a byly programovány pomocí zapojení nebo ovládacích panelů.
* 2. generace 1956-1963
  + Tranzistory místo elektronek => menší, rychlejší a energeticky efektivnějším počítače.
  + programování bylo prováděno pomocí symbolických jazyků a assembleru.
* 3. generace 1964-1971
  + Integrované obvody => menší, levnější
  + Používaly se hlavně vysokoúrovňové programovací jazyky (například COBOL, Fortran) a bylo zavedeno sériové zpracování.
* 4. generace 1971-dnes
  + charakterizována mikroprocesory
  + Dochází k rozmachu síťových technologií, a začínají se objevovat mobilní a vestavěné počítače.

Obsah obrázku text, snímek obrazovky, diagram, Písmo

Popis byl vytvořen automatickyVon Neumanova architektura

* počítač je řízen programem uloženým v paměti
* Von Neumanova architektura počítače nebyla do dnes překonána.
* Její základní principy používá naprostá většina všech počítačů.
* Bylo sice navrženo a částečně též realizováno mnoho alternativních řešení, avšak žádné z nich se nesetkalo s takovým komerčním úspěchem, aby von Neumannovu koncepci zastínilo nebo dokonce zcela vytlačilo z trhu počítačů.

1. Počítač se skládá z těchto jednotek

* **paměť**: slouží k uchování zpracovávaného programu, zpracovávaných dat a výsledků výpočtů
* **řadič**: řídící jednotka, která řídí činnost všech částí počítače
* **aritmetická jednotka**: jednotka provádějící veškeré aritmetické výpočty a logické operace
* **vstupní jednotky**: zařízení určená pro vstup programu a dat
* **výstupní jednotky**: zařízení určená pro výstup výsledků, které program zpracoval

1. Struktura počítače je nezávislá na typu řešené úlohy. Počítač se programuje obsahem paměti.
2. Instrukce a operandy (data) jsou v téže paměti.
3. Paměť je rozdělena do buněk stejné velikosti, jejich pořadová čísla se používají jako adresy sloužící k jejich identifikaci.
4. Program je tvořen posloupností elementárních příkazů (instrukcí), v nichž zpravidla není uložena hodnota operandu (uvádí se adresa v paměti, kde je uložen) - program se při změně dat nemění. Instrukce se provádějí jednotlivě v pořadí, v němž jsou do paměti zapsány.
5. Změna pořadí provádění instrukcí se vyvolá instrukcí podmíněného nebo nepodmíněného skoku.
6. Pro reprezentaci instrukcí i čísel (operandů, výsledků, adres) se používají dvojkové signály a dvojková číselná soustava.

# Jak to celé funguje:

* Pomocí vstupní jednotky se do paměti zavede program a zpracovávaná data
* Řadič čte postupně z paměti instrukce a řídí (dává pokyny) aritmetickou jednotku, která provede výpočet.
* Výsledek výpočtu se uloží do paměti k dalšímu zpracování, nebo může být odeslán na výstupní jednotku.
* Za nejdůležitější rysy von Neumannovy architektury lze považovat body 1 a 2, protože ty definují základní strukturu počítače a způsob jeho řízení.
* Tyto body zůstaly v platnosti zcela bez výhrad do dnes a používá ho většina všech sériově vyráběných počítačů na celém světě.
* Uložení programu v paměti představuje největší přínos ve vývoji počítačů, protože otevřel cestu k univerzálnosti počítačů a tím k jejich využití v takovém měřítku v jakém je známe dnes.
* Poměrně brzy si uživatelé počítačů uvědomili nebezpečí, vyplývající z uložení programu a dat v téže paměti - bod 3 (programátorskou chybou lze snadno zničit program).
* V některých systémech dává přednost oddělení paměti dat od paměti programu   
  Tato struktura byla použita v počítačích MARK III, MARK IV navržených ve 40. letech na Harvardově univerzitě v Cambridgi.
* Za nejdůležitější rysy von Neumannovy architektury lze považovat body 1 a 2, protože ty definují základní strukturu počítače a způsob jeho řízení.
* Tyto body zůstaly v platnosti zcela bez výhrad do dnes a používá ho většina všech sériově vyráběných počítačů na celém světě.
* Uložení programu v paměti představuje největší přínos ve vývoji počítačů, protože otevřel cestu k univerzálnosti počítačů a tím k jejich využití v takovém měřítku v jakém je známe dnes.

# Harvardská architekturaObsah obrázku text, snímek obrazovky, diagram, Písmo Popis byl vytvořen automaticky

* Zpravidla se používá u j**ednočipových počítačů -** mikrořadičů
  + Program těchto počítačů je uložen v paměti typu ROM (EPROM, EEPROM), čímž je fyzicky oddělen od paměti dat, která je typu RAM.
* Reprezentace čísel i instrukcí dvojhodnotovými signály je omezení, které se dodnes nepodařilo odstranit, protože technické parametry používaných součástek neumožňují spolehlivě odlišit více než dva fyzikální stavy.
* Dnešní počítače disponují vstupně-výstupními jednotkami:umožňují výstup dat i vstup dat či programu (typickým příkladem je pevný disk)

# Netradiční počítače

* **Netradiční:**
  + **počítače řízení tokem dat**
  + **počítače řízení tokem požadavků**
  + **systolické systémy**
  + **neuronové počítače**

## Počítače řízení tokem dat

* Instrukce je spuštěna v okamžiku kdy má   
  k dispozici všechny operandy.
* Pro okamžik provedení není důležité, zda instrukce byla zapsaná na konci či začátku programu.
* Hlavní odlišnost je ve způsobu programování.

Nevýhoda**:**

* přílišná hardwarová složitost pro sestavení univerzálního počítače.

Použití**:**

* jednoduché – jednoúčelové počítače využívané pro předzpracovávání dat (například přepočítání vstupních údajů snímaných čidly).

## Počítače řízení tokem požadavků

* Nevýhodou počítačů řízených tokem dat je „vytváření výsledků do zásoby“.
* Počítače řízené tokem požadavků provedou výpočet právě v okamžiku, kdy je požadován výsledek pro další výpočet

### Výhoda:

* není potřeba operace rozhodování.

### Nevýhoda:

* pomalý

## Systolické systémy

* Systolický systém je pevně propojená síť funkčních jednotek určená pro řešení jedné úlohy (nebo skupiny podobných úloh)
* Jednotky jsou schopné provádět pouze jednoduché operace. Například: **y = y + w \* x**
* **Výhoda:** snížení požadavků na rychlost operační paměti.

## Neuronové sítě

* Inspirovány poznatky   
  o neuronech a nervových sítích živých organizmů
* Schopnost řešit silně nelineární úlohy
* Schopnost učit se
* Schopnost zevšeobecňovat
* Využití pro klasifikaci, regresi a predikci časových řad

### Umělá neuronová síť

* je distribuovaný výpočetní systém sestávající z dílčích podsystémů (neuronů)
* je inspirován neurofyziologickými poznatky o struktuře a činnosti neuronů a nervových systémů živých organizmů
* **neuron** je základní výpočetní jednotkou neuronových sítí
* obsahuje několik vstupů, které jsou ohodnoceny vahami a jeden výstup
* v neuronu pobíhají dva procesy:
  + výpočet (post-synaptického) potenciálu
  + výpočet hodnoty výstupu pomocí aktivační funkce, nejčastěji tzv. sigmoidy
* pro větší výpočetní sílu se neurony uspořádávají do sítí neuronů

### Typy neuronových sítí

* Existuje celá řada typů neuronových sítí
  + Vícevrstvá perceptonová síť (MLP)
  + Hopfieldova síť
  + Kohonenovy samoorganizující se mapy
  + Síť RBF
* Každý typ se hodí pro jinou třídu úloh
* Základními úlohami neuronových sítí jsou klasifikace a regrese (aproximace)
* Podle přítomnosti „učitele“ můžeme neuronové sítě dělit na sítě s učitelem a bez učitele
* Proces učení neuronových sítí
* Pro učení (trénování NS) je třeba mít dostatek reprezentativních příkladů:
  + trénovací množina
  + výběrová množina
  + testovací množina
* Proces učení neuronových sítí
* Na začátku učení bývají váhy nejčastěji nastaveny na náhodná čísla
* Proces učení se snaží minimalizovat odchylku (chybu) mezi skutečným (aktuálním) a požadovaným výstupem
* Každá neuronová síť má jiný algoritmus učení, vesměs jsou to „iterační procesy“.
* **Použití:**
  + rozpoznávání a komprese obrazu, zvuku
  + předvídání vývoje časových řad   
    (např. burzovních indexů)
  + k filtrování spamu