

Základní pojmy informatiky a výpočetní techniky

[Historie výpočetní techniky](#)

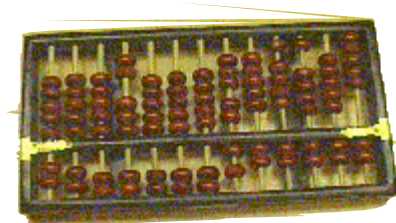
[Informace](#)

[Bit, byte a kódy](#)

[Počítač](#)

Historie výpočetní techniky

Již ve **starověku** vznikl **abakus** (viz obrázek), který měl různou podobu: počítacího prkénka s oblázkou ve žlábkách či korálků navlečených na tyčkách. Tato pomůcka pro počítání se zachovala téměř do současné doby a zejména v Japonsku a Rusku byla vytlačena až nástupem elektronických kalkulaček.



V **novověku** přispěl k rozvoji výpočetní techniky matematik a filozof **John Napier**, který v roce 1614 zveřejnil své **logaritmické tabulky**. Logaritmy, logaritmické tabulky a později logaritmické pravítko umožňují převést poměrně komplikované násobení a dělení na snazší sčítání a odčítání.



Jen o několik let později **Wilhelm Schickard** vynalezl **mechanickou kalkulačku** (1623), o které uvažoval už dávno před ním génius **Leonardo da Vinci**. K rozvoji mechanické kalkulačky přispěl ve svých 19 letech francouzský matematik a filozof **Blaise Pascal** (na obrázku). V roce 1642 vyrobil z kovu svoji **pascalinu** – mechanickou kalkulačku pro sčítání a odčítání, u které se mechanické číselníky ovládaly speciální jehlou.

K rozvoji mechanické kalkulačky přispěl mj. i německý učenec **Gottfried Wilhelm von Leibniz**. Mechanické kalkulačky tvořily hlavní prostředek výpočetní techniky až do 60. let 20. stol., dokud je nevytlačily kalkulačky elektronické a počítače.

Roku 1889 **Hermann Hollerith** vyvinul v USA **stroj na děrné štítky** a použil ho k vyhodnocení sčítání lidu USA, které proběhlo v roce 1890. Oproti předchozímu sčítání lidu za rok 1880 vyhodnocovanému ručně, získal výsledky za značně kratší dobu s mnohem méně pracovníky. Děrné štítky používal již začátkem 19. stol. francouzský tkadlec hedvábí **Joseph-Marie Jacquard** k řízení tkalcovského stavu při vytváření složitých vzorů. Stroje na děrné se používaly v institucích a firmách pro zpracování dat až do 60. let 20. stol.



V 19. století značně předběhl dobu **první pokus o počítač**, který učinil anglický matematik **Charles Babbage**. Nejprve se pokusil o vytvoření **diferenciálního stroje**, který měl pracovat podle pevného programu. Později tento nápad opustil a věnoval se myšlence **analytického stroje**, který by měl být už skutečným univerzálním počítačem, který provádí různé operace podle předem připraveného progra-



mu. Diferenciální stroj by ještě mohl být sestrojen, byť by to bylo monstrum velké asi jako parní lokomotiva a stejně jako ona by to bylo mechanické zařízení poháněné parou. Analytický stroj nepostavil ani Babbage a ani později jeho syn, v 19. stol. to byl pro tehdejší techniku ještě neřešitelný úkol. Na obrázcích je Charles Babbage a jeho nejbližší spolupracovnice baronka **Ada Augusta** von Lovelace, nadaná matematická, dcera anglického romantického básníka lorda Byrona, kterou lze označit jako první **programátorku**.

Skutečný počítač vznikl až v období 2. světové války – v období snad nejstrašnější tragédie lidstva, v níž zahynuly desítky milionů vojáků a civilistů. Na vývoji výpočetní techniky se tehdy často podílely armádní štáby, které potřebovaly šifrovací a dešifrovací techniku a výpočetní techniku pro vývoj nových zbraní. Za tvůrce prvního skutečného počítače se dnes považují:

- a) **Konrad Zuse**, který v Německu od roku 1938 postupně vytvořil několik mechanických kalkulátorů a prvních reléových počítačů s poměrně malou kapacitou paměti a s malým početním výkonem. Zuse byl na jedné straně dobře teoreticky vybaven, na straně druhé se na jeho vývoji armádní výzkum podílel jen minimálně, takže měl jen minimum prostředků a jeho prototypy naštěstí pro lidstvo nacistickému Německu nikterak významně neprospěly.
- b) **Howard H. Aiken** a jeho spolupracovníci z Harvardské univerzity s podporou IBM vytvořili v USA reléový počítač MARK 1. Tým H. Aikena měl dostatek podpory i finančních prostředků.
- c) **John V. Atanasoff** v USA již dříve před Aikenem vytvořil speciální kalkulátor. Po vstupu USA do války opustil Iowa State University, která vývoj podporovala a vývoji digitálních počítačů se už nevěnoval.

Ve vývoji **počítačů** se často hovoří o **generacích**, které jsou založeny na základním stavebním prvku počítačů. Dosud se hovoří o generacích (uvedená období jsou přibližná):

0. generace (1940 – 1945) – založená na **elektromechanických relé**. Patří sem především výše uvedené prototypy počítačů.

1. generace (1945 – 1955) – je založena na **elektronkách**. Počítače jsou opět pouze prototypy s malou kapacitou operační paměti a malým početním výkonem. Vstup je hlavně pomocí děrných štítků (či děrné pásky). Jako paměťové prvky se používají elektronky a magnetické bubny. Spolehlivost počítače je nízká, spotřeba elektrické energie velká. Programovací jazyky ještě neexistují, programuje se ve **strojovém kódu**. Typickým představitelem je první skutečný počítač **ENIAC**. Bylo to neuvěřitelné monstrum s téměř 18 tisíci elektronkami, zabírající plochu basketbalového hřiště, napájené menší elektrárnou a chlazené dvěma leteckými motory.

2. generace (1955 – 1965) – je spojena s **tranzistory a diodami** – jsou to tedy již skutečně elektronické počítače. Roste výkon počítačů, kapacita jejich vnitřní a vnější paměti. Vznikají první programovací jazyky. Počítače se začínají vyrábět v (zatím malých) sériích a používají je státní orgány a velké firmy.

3. generace (1965 – 1975) je založena na **integrovaných obvodech**. Počítače se začínají dělit na velké **sálové**, menší **kancelářské** a postupně i na **osobní**. Značně roste jejich výkon, umožňují už práci více uživatelů najednou a mohou současně pracovat na více úlohách. V oblasti sálových počítačů je nejvýznamnější řada počítačů **IBM 360**. Vznikají i **superpočítače** – např. pro předpověď počasí.

4. generace (od roku 1975 až dosud) – roste počet součástek v integrovaných obvodech, vzniká **mikroprocesor**. Roste výkon počítačů, zvyšuje se kapacita paměti, počítače se zmenšují a zlevňují. Rozšiřuje se jejich nasazení, vznikají nové programovací jazyky a balíky uživatelských programů. Hlavním představitelem je řada počítačů **IBM PC** (1981) založená na řadě procesorů firmy **Intel**.

Informace

V teorii informací je informace charakterizována:

Informace je každá zpráva, která snižuje entropii (neurčitost chování) příjemce.

Zjednodušeně řečeno, **informace je každá zpráva, která ovlivňuje rozhodování a chování příjemce informace.**

Teorie informace užívá též **jednotku velikosti informace**, kterou je jeden **bit**.

Informace o velikosti jednoho bitu je informace, že nastal či nestane jeden ze dvou stejně pravděpodobných jevů. Dva stejně pravděpodobné jevy jsou např. výsledek házení jednou mincí, kdy buď padne lící, nebo rubová strana mince. Podobně je to výsledek házení hrací kostkou, kdy buď padne lichý, nebo sudý počet ok. Jednotka informace vznikla pro potřeby telegrafie při úvahách o počtu dvojic signálů potřebných pro vyslání jednoho znaku mezinárodní abecedy.

Informatiku lze chápat jako vědní obor o získávání, přenosu a zpracování informací. Pro naši potřebu jí budeme chápat spíše jako praktickou činnost spojenou s uvedenými činnostmi (získání, přenos a zpracování informací) prováděnou za pomoci počítačů.

Pozn.: slovo *informatika* má původ francouzský. (V angličtině se vědní obor s tímto obsahem nazývá většinou *computer science* - věda o počítačích.) Francouzština je zajímavá tím, že na rozdíl od prakticky všech ostatních jazyků nepřebírá chaoticky anglické termíny z oblasti výpočetní techniky, ale důsledně si buduje vlastní terminologii, která má (přes pochopitelný poněkud umělý charakter) svoji vnitřní logiku a větší výstižnost slov - např. slovo počítač (angl. *computer*) se neodvozuje od dnes již zastaralého pouhého počítání, protože francouzsky počítač se řekne *l'ordinateur* - pořadač.

Bit, byte a kódy

Bit byl vysvětlen výše. Slovo bit v angličtině existuje a označuje též kousek, chvilku nebo drobnou minci. V teorii informace se později začalo říkat, že je to zkratka z anglických slov „binary digit“ - dvojková číslice. Zde je zvláštní případ, neboť obvykle nějaký *akronym* (zkratka) vznikne a začne se ho užívat jako slovo, zde slovo existovalo, teprve pak se začalo tvrdit, že je to akronym.

Protože počítače jsou vybudovány z bistabilních prvků (z prvků, které mají dva stabilní stavy - např. proud prochází x neprochází, zmagnetizováno jedním x opačným směrem) jsou veškeré údaje v něm zachycené vyjádřeny číselně ve dvojkové soustavě.

Kolik bitů je potřeba k vyjádření jedné desítkové číslice?

Dvě na třetí dává osm stavů, dvě na čtvrtou pak 16 stavů. **Pro vyjádření jedné desítkové číslice je zapotřebí čtveřice bitů** (s redundancí - nadbytečností - šesti stavů).

Jako skupina bitů se později vžila osmice bitů zvaná **byte** (čti bajt) - česky zvaná též slabika, vžil se i zápis bajt. Tato skupina osmi bitů - **byte** - se postupně stala jednotka velikosti operační i vnější paměti jakožto (viz dále) paměť potřebná pro zobrazení jednoho abecedně číslicového znaku.

Na rozdíl od bitu je původ slova byte nejasný.

Větší skupiny bitů a bajtů jsou:

- 1 bit (b),
- 1 byte (B) = 8 bitů,
- 1 Kilobyte (Kilobajt - KB) = 2^{10} bajtů = 1 024 B = 8 192 b
- 1 Megabyte (Megabajt - MB) = 2^{10} KB = 1 024 KB = 2^{20} B = 1 048 576 B = 8 388 608 b
- 1 Gigabyte (Gigabajt - GB) = 2^{10} MB = 1 024 MB = 2^{20} KB = 1 048 576 KB = 2^{30} B = 1 073 741 824 B = 8 589 934 592 b
- atd.

Všimněte si, že předpony (Kilo, Mega a Giga) se zde píšou s velkým písmenem, protože Kilo není 1000 apod., ale že Kilo je 1024, neboť jde o mocniny dvou jakožto základu dvojkové soustavy.

Kolik bitů je třeba k zobrazení jednoho abecedně číslicového znaku?

Uvážíme-li, že číslic je 10, písmen (jen velkých nebo jen malých) mezinárodní abecedy je 26, pak alespoň šest bitů ($2^6 = 64$), kterých je zapotřebí pro vyjádření těchto 36 znaků, neboť 5 nestačí ($2^5 = 32$).

Přesto existoval **mezinárodní telegrafní kód CCITT 2** (zkratka označuje *Mezinárodní telegrafní a telefonní poradní výbor*), který užíval jen 5 bitů, byl to **kód pětistopé děrné pásky**. Obsahoval dva přepínače řad a dvě řady - číselnou a textovou. Mohl tak vyjádřit 10 číslic, 26 písmen mezinárodní abecedy (jen velká nebo jen malá) a ještě 26 různých - tisknutelných znaků (např. pro operátory aritmetických operací aj.) i netisknutelných (řídících - např. zvonek) znaků.

Dnes je základním kódem užívaných u počítačů třídy PC **kód ASCII** [aski]. ASCII je zkratka pro *American Standard Code for Information Interchange* - Americký standardní kód pro výměnu informací. Původně byl sedmibitový a mohl tak vyjádřit 128 různých tisknutelných a netisknutelných znaků. Osmý bit byl užíván jako paritní, pro kontrolu správnosti při přenosu informací. Dnes se užívá osmý bit pro vyjádření dalších 128 znaků, především národních abeced. Kód obsahuje 32 netisknutelných řídících znaků a 224 znaků tisknutelných - písmen velkých i malých, číslic, aritmetických operátorů, znaků národních abeced, znaků pro jednoduché a dvojitě rámečky zobrazované v textovém režimu a dalších.

Kód ASCII má dnes různé **národní mutace**. I když problémů se správným zobrazování češtiny je v prostředí dnešních grafických systémů již méně než v textovém prostředí (např. DOS), přesto je třeba správnému zobrazování češtiny stále věnovat pozornost – zejména na internetových stránkách v e-mailech apod. Situace vypadá zjednodušeně takto:

- **ISO-8859-2 (ISO Latin 2, podle IBM CP912)** – je norma pro zobrazování angličtiny a řady středoevropských jazyků užívajících latinku včetně češtiny. Tento kód se užívá v systémech UNIX a jim příbuzných (např. Linux). Tato norma je specifikací MIME jako **iso-8859-2**.
- **CP852 (IBM852, PC Latin2)**. Toto kódování se běžně označuje jako PC Latin 2 nebo jako kódová stránka 852. Podporuje ho Microsoft i IBM. Obsahuje všechny tisknutelné znaky normy ISO-8859-2, avšak diakritická písmenka jsou na jiných pozicích. Tento kód není specifikován v MIME.
- **Windows-1250 (CP1250, WinCS, WinEE)** je kódování používání v MS Windows od verze 3.1 až do současnosti. Tento kód se u některých znaků (např. „š“) liší od normy ISO-8859-2. Tato znaková sada je specifikovaná podle MIME jako **windows-1250**.

Osmibitové kódování se postupně opouští a přechází se k šestnáctibitovému kódování Unicode.

Počítač

Pod pojmem **počítač** budeme rozumět **číslicový počítač**.

Číslicový počítač je stroj na zpracování dat, pracující samostatně podle programu předem uloženého do paměti.

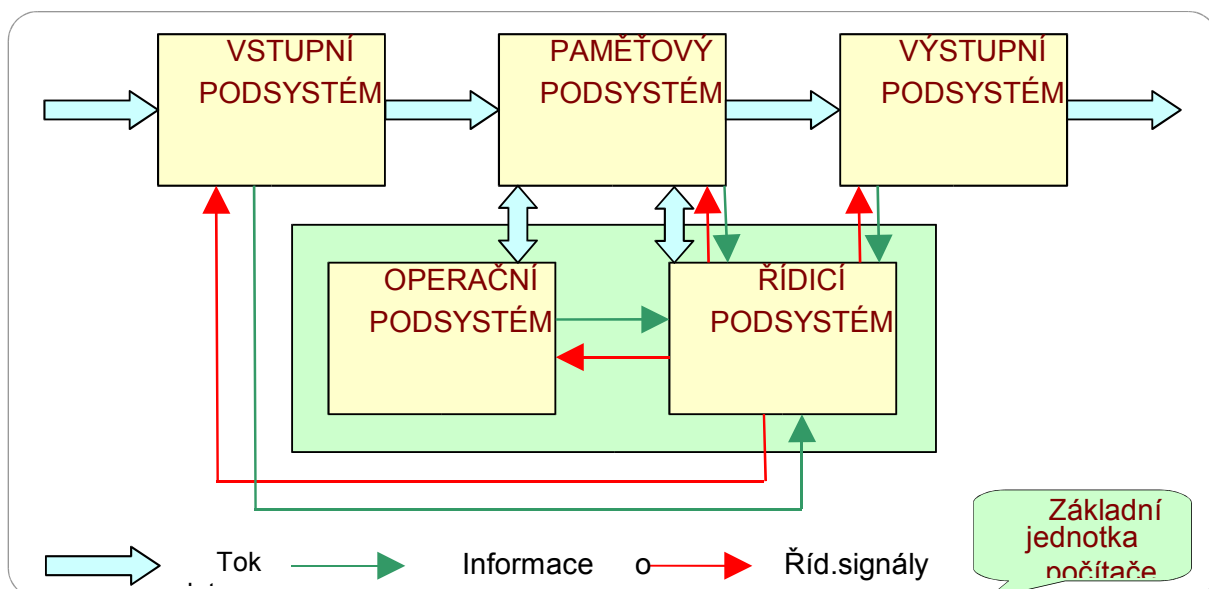


Model číslicového počítače navrhl již v roce 1945 americký matematik maďarského původu **John von Neumann** (1903 – 1957). Jeho modelu počítače, který s jistými výjimkami zůstal zachován dodnes, říkáme **von Neumannova koncepce počítače**.

Části číslicového počítače

Podle von Neumannovy koncepce se číslicový počítač skládá z pěti základních částí (viz obr. 1):

- **Paměťový podsystém** (operační paměť) sloužící k uchování veškerých informací, tj. programů (systémových, aplikačních, uživatelských) a dat (vstupních, pracovních a výsledků).
- **Operační podsystém**, neboli aritmeticko-logická jednotka (**ALU - Arithmetic Logic Unit**), provádějící veškeré aritmetické a logické operace s daty.
- **Řídicí podsystém** řídící a koordinující (pomocí řídicích signálů zasílaných do ostatních podsystémů počítače) činnost spolupracujících podsystémů. Činí tak na základě dekodování instrukcí získaných z paměťového podsystému. Stavů těchto podsystémů jsou naopak zasílány do řídicího podsystému jako informace o stavu (stavová hlášení).
- **Vstupní podsystém** řídící vstup dat z periferních zařízení do paměťového podsystému počítače.



Obr. 1 Blokové schéma číslicového počítače – von Neumannova koncepce

- **Výstupní podsystém** řídící výstup dat (výsledků) z paměťového podsystému počítače na periferní zařízení.

Řídícímu a operačnímu podsystému počítače říkáme **základní** nebo **centrální jednotka (CPU - Central Processing Unit)** neboli **procesor** počítače.

Mikroprocesor je tedy základní jednotka počítače (řídící a paměťový podsystém) v mikroprovedení, v jednom pouzdře integrovaného obvodu.

Základní odlišnosti dnešních počítačů od von Neumannovy koncepce:

- Počítač může pracovat i s **více než jedním procesorem**.
- Von Neumannův počítač pracuje vždy jen s jedním programem, což vede k neefektivnímu využití strojového času. Je tedy obvyklé, že počítač zpracovává paralelně více programů zároveň (tzv. **multitasking**).
- Vstupní/výstupní podsystémy dnešních počítačů umožňují jak vstup, tak výstup dat, a jsou (hlavně) schopny vstupní/výstupní operaci samostatně řídit. Pro tyto podsystémy se dnes používají názvy **řadič/adaptér/kontrolér** apod..
- Program se do paměti nemusí zavádět celý, ale je možné zavést jen jeho část a ostatní části zavádět až v případě jejich potřeby.
- Již od svého vzniku v roce 1974 mají mikropočítače většinu elektroniky zahrnutou na jedné desce plošných spojů zvané **základní/systémová deska (motherboard)**, na které je mikropočítač vystavěn. Na této desce jsou umístěny mikroprocesor, paměti, generátor hodinových signálů, obvody pro připojení klávesnice, sběrnice pro přenos dat mezi jednotlivými podsystémy a patice pro připojení V/V modulů.
- V základní konfiguraci mikropočítače pak bývají kromě vnitřní (operační) paměti i tzv. **vnější paměti**, kterými jsou nejčastěji pevné disky.

Další koncepce číslicových počítačů

- Kromě von Neumannovy koncepce se můžeme setkat např. s tzv. **Harvardskou koncepcí**, která se odlišuje tím, že předpokládá existenci dvou oddělených pamětí. V jedné paměti jsou uloženy programy, ve druhé jsou uložena data. Tuto koncepci mají zejména některé kapesní kalkulátory a jednoúčelové programové automaty.
- Základním nedostatkem jednoprocessorových počítačů je jejich pomalost, podmíněná sériovým způsobem činnosti, kdy procesor čte instrukce z paměti, tyto dekóduje, potom načítá operandy a pak vykonává požadovanou operaci a ukládá výsledek. Teprve potom čte další instrukci. Odstranění tohoto nedostatku nabízejí různé architektury využívající více procesorů pro zpracování jedné úlohy. Takovými architekturami jsou: **řetězené procesory**, **multiprocessory** (paralelní soustava procesorů), nebo **maticové procesory** (procesorová pole).
- Dalšími nástroji pro zvýšení výkonu procesorů mohou být **přesahování (overlapping)**, kdy vykonávání některých instrukcí se dokončuje ještě v době náležející další instrukci, a **řetězení (pipelining)**, které lze chápat jako „přesahování v procesoru“, který se skládá z řady sériově řazených subprocesorů (jde o pokročilejší architekturu řetězených procesorů).