## Základní počítačové komponenty

## Case – počítačová skříň

* Slouží k upevnění všech komponent.
* Většinou je celokovová ve standardní velikosti, obsahuje úložné šachty (bay).
* Velikost se určuje základní deskou.

## PSU, UPS

**PSU = Power Supply Unit (napájecí zdroj)**

* Slouží ke zpracování střídavého proudu dodávaného ze sítě na nízké napětí, které je potřebné k napájení komponentů v PC.
* Většina napájecích zdrojů spadá do standartu ATX.

**UPS = Uninterruptible Power Supply/Source (zdroj nepřerušovaného napájení)**

* Dodává a udržuje energii pro případný výpadek proudu, aby stroje nebyli bez napájení.
* Toto zařízení se používá hlavně u počítačů nebo serverů, které musí být stále spuštěné.

## MotherBoard – základní deska

* Propojuje všechny komponenty do jednoho celku.
* Klasická základní deska poskytuje zapojení procesoru pomocí **socketu** (CPU) a **operační paměti** (RAM), dále se dá pomocí rozšiřujících slotů připojit např. grafická/zvuková karta
* Je na ní umístěna ROM (energeticky nezávislá pamět) a BIOS .
* Obsahuje Chip-set, který se dělí na **Southbridge a Northbridge.**
* **Southbridge**
  + Čip realizuje pomalejší funkce základní desky v počítačové architektuře se severním a jižním můstkem.
  + Jižní můstek odlišíme od severního snadno tak, že není přímo spojen s procesorem.
* **Northbridge** 
  + Severní most zajišťuje komunikaci mezi CPU, pamětí RAM (řadič paměti), AGP portem nebo PCI Express sběrnicí a také zajišťuje spojení s jižním mostem
  + Základní deska obsahuje několik druhu konektorů pro připojení dalších komponent jako například – PCI-Express (univerzální konektor pro přídavné karty), AGP (konektor hlavně pro grafické karty), ...

## CPU - procesor

* Vykonává strojové instrukce, ze kterých je tvořen počítačový program a obsluhuje jeho vstupy a výstupy.
* Mezi hlavní součásti procesoru patří aritmeticko-logická jednotka, registry a řadič, který řídí činnost procesoru (načítání strojových instrukcí z paměti, jejich dekódování, provedení a uložení výsledků).
* Má-li v sobě více procesorových jednotek, je označována jako vícejádrový procesor.
* Co se týče výkonu, je potřeba sledovat jeho **taktovací frekvenci** (GHz) a počet jader.

## RAM - operační paměť

* Volatilní paměť (Energeticky závislá paměť)
* **K udržení dat potřebuje neustálé napájení.**
* Z pravidla se označuje jako RAM a dá se dělit na statickou a dynamickou RAM.
* Ukládá se do ní většinou běžící programy počítače a jejich data.
* Paměti současných počítačů, tabletů a chytrých telefonů mají velikost řádu GB.
* Jaké paměti použijeme se vždy řídí použitou čipovou sadou na základní desce a základní deskou samotnou – SIMM, DIMM, DDR - DDR4, ...

## GPU - grafický procesor



* Zajišťuje rychlé grafické výpočty a změny obsahu videopaměti, které zobrazí na monitoru.
* Dnes se dají použít i k jiným než grafickým výpočtům.
* V počítači je umístěn buď na samostatné grafické kartě, nebo je na základní desce integrován s mikroprocesorem do jednoho čipu tzv. APU (např. Intel HD Graphic).
* **Vyžaduje intenzivní chlazení a výkonný zdroj energie.**
* Dělí se na dedikované, integrované nebo externí

**Dedikované grafické karty**

* + Většinou jsou připojené pomocí PCI-Express nebo APG.
  + Jsou výkonnější než integrované grafické karty, protože obsahuje svoji vlastní paměť RAM
  + Mají snadnější možnost výměny, ale jsou náročnější na výkon.
* **Integrované grafické karty**
  + Úzce vázané sdílí operační paměť a mohou spolupracovat s procesorem, anebo tu jsou ty, které sice můžou sdílet paměť s procesorem, ale nejsou schopny úzce spolupracovat.
  + Výhodou úzce vázaných je, že jsou rychlejší – pracuje se společnou pamětí a se stejnými daty, které se nemusí přetahovat z jedné paměti do druhé.
  + Nemají tak velkou spotřebu energie
* **Externí GPU**
* Je umístěna mimo počítač.
* Používají se většinou u notebooků, jelikož jim často schází výkonná grafická karta
* Nemají moc velkou podporu od oficiálních distributorů graf. karet

## Úložiště

* Hlavním účelem je udržet data v počítači, buď pro aktuální, nebo pozdější potřebu.
* **Nevolatilní paměť (Energeticky nezávislá paměť)**
* Slouží k uložení dat, které musí být v počítači i mimo něj uloženy bez jakéhokoliv napájení.
* **Pevný disk**
  + Ukládá se na něj data, která se v něm musí uchovat i po přerušení napájení.
  + Například programy, ovladače, filmy, ...
  + Jsou připojeny pomoci SATA nebo PATA kabelu k základní desce.
  + **Dva hlavní typy HDD a SSD**
* **HDD**
* Uvnitř pevného disku jsou kruhové desky, které se otáčejí velmi rychle (běžně 7 200 otáček za minutu) a pomocí zápisové hlavy se data ukládají na plotny.
* Používá metodu magnetického zápisu.
* Kapacita těchto disků se pohybuje v řádech GB až TB.
* Nevýhodou těchto disků je, že jsou náchylné na otřesy a tím se dokážou data na nich uložená poškodit.
* **SSD**
* Je menší, rychlejší a odolnější než HDD, jelikož neobsahuje žádné pohyblivé části.
* Většinou má ale menší kapacitu než HDD.

## Chlazení

**Pasivní chlazení**

* Je to nepohyblivá součástka, která má žebra k zajištění co největší plochy k lepšímu předávání tepla okolnímu vzduchu.
* Nejsou tak kvalitní jako aktivní chladiče, ale jsou u některých komponent dostačující.

**Aktivní chlazení**

* Používá se ventilátor, který vhání studený vzduch do skříně a odvádí teplý vzduch ven.
* Jsou sice hlučnější, ale za to efektivnější a mají nižší nároky na velikost.

**Vodní chlazení**

* Je to soustava, kde protéká chladící kapalina (destilovaná voda)
* Vodní chlazení pouze pomáhá přesouvat velká kvanta tepla z uzavřených, a pro vzduch těžko dostupných, míst na místa pro chlazení vhodnější

## Vstupní a výstupní periferie

**Dělení periferií**

Periferie se dělí podle toho, jestli přenáší informace do počítače nebo počítač předává informace periferiím

* **Vstupní periferie**
  + Slouží k předávání dat do počítače.
  + Mezi tyto zařízení patří hlavně klávesnice, myš, skener, webkamera,…
  + Většinou jsou uživatelské, ale mohou být i specializované (čtečka čárových kódů)
* **Výstupní periferie**
  + Přenáší data z počítače k uživateli.
  + Monitor, tiskárny, reproduktor, …

**Charakteristika a důležité parametry periferií**

**Monitor**

* + Základní výstupní elektronické zařízení.
  + Slouží k zobrazování textových a grafických informací.
  + Používají barevný model RGB.
  + Když je připojen k počítači, je propojen s grafickou kartou.
  + Dělí se na CRT, LCD, LED, OLED, PLAZMA, ...
    - * + **LCD monitor**

U klasických LCD panelů je na zadní stěně monitoru zdroj světla (CCFL)

Odsud světlo putuje do speciální rozptylovací vrstvy, která se pokusí světlo co nejrovnoměrněji rozvést po celé ploše monitoru.

Dále světlo prochází přes první polarizační filtr do vrstvy s tekutými krystaly, které jsou řízeny elektronikou monitoru dle vstupního signálu - zde se určuje intenzita jasu jednotlivých pixelů.

Nyní stále ještě bílé světlo zamíří do vrstvy s barevným RGB filtrem, odkud se dále již v barvě přenáší na druhý polarizační filtr.

Jako ochranná vrstva slouží tenké sklo, na němž jsou dále ještě nasazeny tři vrstvy, jež se snaží o co nejlepší rozptýlení světla a další vylepšení promítaného obrazu.

Výhodou je, že je kompatibilní, lehký, malá energetická spotřeba, ale má omezené pozorovací úhly a malý kontrastní poměr

* + - * + **LED monitor**

Podsvícení pomocí LED.

Také je to LCD monitor, ale místo podsvícení CCFL, používá LED.

Vysoký jas a dynamický kontrast, ale je dražší a je tu možnost přesvětlení.

* + - * + **Parametry**

Obrazovka - úhlopříčka, rozlišení, typ panelu, odezva, FPS, jas, ...

Vstupy/Výstupy - konektory k připojení externích úložišť, HDMI, ....

Spotřeba a funkce

**Tiskárna**

* + Je to výstupní zařízení.
  + Převádí informace v elektronické podobě na papír.
  + Používají hlavně barevný model CMYK, jelikož má oddělenou černou barvu.
  + Můžeme jí připojit drátově (USB, LPT, RJ-45) nebo bezdrátově.
  + Kontaktní tiskárny
  + Využívají proklepávání pásky pomocí celých znaků či jehliček.
  + Na tiskové pásce je naneseno barvivo a to se při úderu obtiskne na papír.
  + Mají nízké náklady, jsou vcelku hlučné, kvalita tisku není dobrá, v omezené míře umožňují barevný tisk.
  + Řádkové, Jehličkové, Bubnové nebo Znakové tiskárny.
* **Bezkontaktní tiskárny**
  + - K tisku jsou využity technologie, které nanášejí barvivo na papír bez použití mechanického úderu.
    - Inkoustové tiskárny

K tisku používají tekutou barvu – inkoust, který je umístěný v nádobce, které se říká cartridge.

* + - Drop on Demand

Jednotlivé kapky inkoustu jsou z tiskové hlavy vypuzovány jen tehdy, mají-li dopadnout na potiskované médium.

Využívá dvě základní technologie pro vypuzení kapky inkoustu (Bubble jet a Ink jet)

* + - Kontinuální inkoustový tisk

Tiskárny vytvářejí nepřetržitý proud velkého množství kapiček inkoustu, kde vybrané kapky jsou vychýleny tak, aby dopadly na potiskované médium, a ostatní kapky jsou odváděny sběrným systémem zpět do zásobníku.

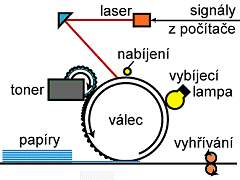
* + - Bublinkové tiskárny (Bubble Jet)

Používají k vytváření kapek tlaku, který v tiskové komůrce vytvářen bublinkou vypařujícího se inkoustu.

* + - Piezoelektrické tryskové tiskárny (Ink jet)

Využívají k tisku piezoelektrickou deformaci tryskové komůrky

* + - Laserové tiskárny

Využívají princip elektrofotografického tisku.

Základním prvkem tiskové jednotky je tiskový válec vyrobený z elektricky vodivého materiálu a na povrchu válce je nanesena vrstva polovodivého materiálu.

Průběh

tiskový válec se nabije záporným elektrickým nábojem

v místě dopadu laseru se změní náboj na kladný

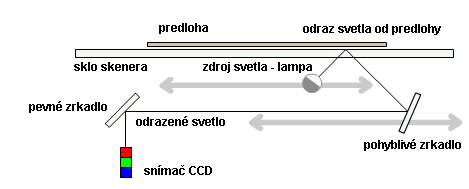
záporně nabitý toner přilne na kladně nabitém místě válce

válec se obtiskne na papír a toner na něm ulpí

zažehlovací válec trvale přichytne (zapeče) toner na papír (cca 200°C)

* **Parametry**
  + - Barevnost tisku - jaký používá barevný model (RGB - aditivní, CMYK - subtraktivní)
    - Tisková média - na co může tisknout (A4, A3, obálky,…)
    - Hluk - hlučné jsou jehličkové, přijatelný hluk mají laserové a LED tiskárny, inkoustové jsou nejtiššími.
    - Kapacita disku - udává se v počtech stran na měsíc
    - Připojení tiskárny, Náklady,…

**Skener**

* + Je to vstupní periferie, která převádí 2D nebo 3D předlohu do digitální podoby většinou pomocí počítače.
  + Čtečka čárových kódu
    - Využívá paprsek laseru nebi laserové paprsky.
    - Většinou jsou zabudované nebo ruční.
* **Ruční skener**
  + - Přejíždí se s nim přes předlohu.
    - Jeho nevýhodou je velmi malá kvalita, jelikož se musí předjíždět a to nedělá přesný pohody po předloze.
* **Stolní skener**
  + - U stolního skeneru se předloha pokládá na sklo a poté se nasnímá.
    - Princip je podobný jako u tisku.
    - Dnes jsou jedny z nejrozšířenějších a nejpoužívanější.
    - Jejich nevýhoda je ta, že jsou omezené velikostí.
    - Jsou dnes už v celku levné od 1 000,-
* **Parametry**
  + - Barevná hloubka

Udává maximum barev, které je skener schopen nasnímat.

Dnes to bývá většinou 48 bitů (281 474 976 710 655 odstínů).

* + - Rozlišení obrazu

Udává se v DPI (počet tiskových bodů na palec.

Znamená, jak moc bude výsledný obraz jemný a nerozmazaný.

* + - Velikost nasnímané plochy

Udává, jakou maximální velikost předlohy může nasnímat.

* + - Denzita

Schopnost skeneru odlišit tmavé body. Většinou se nachází na kvalitnějších skenerech. V praxi to znamená, že když nám horší skener někde nasnímá černou plochu, tak ten s lepší denzitou má na to místě kresbu.

**Způsoby připojení periferií k PC**

**Drátově**

* + Připojení je vedeno přes kabel, který vede z periferie do PC
  + Jsou určené různé standarty pro různé periferie.
  + Výhodou je kvalitnější a rychlejší přenos dat.
  + Nevýhodou však potřeba kabeláže.
    - * + **USB**

Dá se díky němu připojit snad každé zařízení.

Nejčastěji ovšem klávesnice nebo myš.

Jsou aktuálně dvě verze USB 2.0 a USB 3.0/3.1.

USB 3.1 je nejnovější a poskytuje nejrychlejší přenos.

* + - * + **VGA**

Je to grafický standart, pro připojování hlavně monitoru.

Dokáže přenášet pouze video.

Používá graf. model RGB.

* + - * + **DVI**

Jedná se o zobrazovací standart.

Je částečně kompatibilní s HDMI.

Existují 3 základní typy

DVI-D - pouze digitální přenos

DVI-A - pouze analogový přenos

DVI-I -analogový a digitální přenos

* + - * + **HDMI**

Standart pro nekomprimovaný obrazový a zvukový signál v digitálním formátu

Jedním kabelem není možné komunikovat obousměrně.

Dosahuje vysokých rychlostí a zároveň přenáší jak video tak i zvuk.

**Bezdrátově**

* + V dnešní době už se dají periferie připojit i bez potřeby dlouhých kabelů, které překážejí.
  + Ovšem nemůžeme poté dosahovat takových rychlostí.
    - * + **Bluetooth**

Tato technologie je definována standartem IEEE 802.15

Umožňuje bezdrátově připojit dvě zařízení - počítač, mobil, PDA,…

V dnešních mobilních zařízení se používá verze 2.0, která dosahuje rychlosti až 3 Mbit/s

Výhodou je vysoká kompatibilita s ostatními zařízeními, skoro vše má dnes zabudováno Bluetooth

Funguje na docela malé vzdálenosti v řádech metrů.

* + - * + **WiFi**

Definována standartem IEEE 802.11, který popisuje bezdrátovou komunikaci v síti.

Ke komunikaci používá buď 2.4 GHz frekvenci nebo 5 GHz.

Používá 16 pásem, které se navzájem překrývají.

Tři pásma, která se neruší jsou 1, 6, 11.

Zabezpečí

WEP - pouze síť bez jakéhokoliv hesla

WPA - je založen na WEP, ovšem pro připojení do sítě je potřeba, zadat heslo

WPA2 - používá bezpečnější šifru AES, která je ovšem více náročná na výkon, proto se nedá používat na starších zařízeních

* + **Firmware**

Kombinace hardwarového zařízení a počítačových instrukcí a dat, které jsou v zařízení umístěny jako software jen pro čtení.

Bez tohoto softwaru by nemohli součástí fungovat.

BIOS u PC

Firmware kalkulaček, tiskáren, reproduktorů,…

* + **Driver**

Je to software, který pomáhá operačnímu systému komunikovat a pracovat s hardwarem

Některé ovladače obsahuje operační systém, ale nějaké jsou distribuovány s hardwarem

Hlavně se ovladače používají u tiskárny, grafiky, síťové karty, ...

* + **Plug and play**

Jde o technologii, pomocí které dokáže operační systém rozpoznat připojený hardware a přiřadit k němu příslušné ovladače,…

# Úložná zařízení, autorský zákon

**Úložná zařízení**

**Druhy a použití**

**CD**

* Kompaktní disk, určený pro ukládání digitálních dat.
* Data jsou uložena ve stopách na jedné dlouhé spirále začínající ve středu média, která se postupně rozvíjí až k jeho okraji.
* **Kapacita: 700 MB**

**DVD**

* Digitální optický datového nosič, který může obsahovat filmy ve vysoké obrazové a zvukové kvalitě nebo jiná data.
* Má větší kapacitu než CD.
* **Kapacita: 4,7 GB**

**Blue-ray**

* Třetí generace optických disků.
* Data se ukládají ve stopě tvaru spirály 0,1 mm pod povrch disku.
* Ke čtení disku používá modrý paprsek.
* **Kapacita: 25–100 GB**

**Flashdisk**

* USB flash paměť – paměťové zařízení, používané převážně jako náhrada diskety.
* Je vybaveno pamětí typu flash, která umožňuje uchování dat i při odpojení napájení.
* **Kapacita: až do pár TB**

**Externí disk**

* Externí disk je pevný disk, který se používá pro přenos větších objemů dat. Standardně je napájen pomocí USB portu.
* Někdy je potřeba externí napájení.

**HDD**

* Uvnitř pevného disku jsou kruhové desky, které se otáčejí velmi rychle (běžně 7 200 otáček za minutu) a pomocí zápisové hlavy se data ukládají na plotny.
* **Používá metodu magnetického zápisu.**
* Kapacita těchto disků se pohybuje v řádech GB až TB.
* Nevýhodou těchto disků je, že jsou náchylné na otřesy a tím se dokážou data na nich uložená poškodit.

**SSD**

* Je menší, rychlejší a odolnější než HDD, jelikož neobsahuje žádné pohyblivé části.
* Většinou má ale menší kapacitu než HDD.

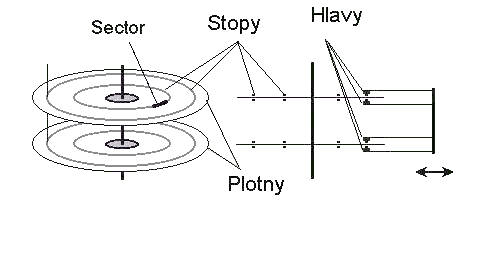
**Paměťová karta**

* Paměťová karta je elektronické zařízení, sloužící k ukládání dat.
* Používá se v digitálních fotoaparátech, PDA, laptopech, mobilních telefonech, přehrávačích a jiných elektronických zařízeních.
* Je to malé, kompaktní zařízení s relativně vysokou kapacitou, odolné vůči magnetickým a elektrickým polím.

**Způsoby zápisu**

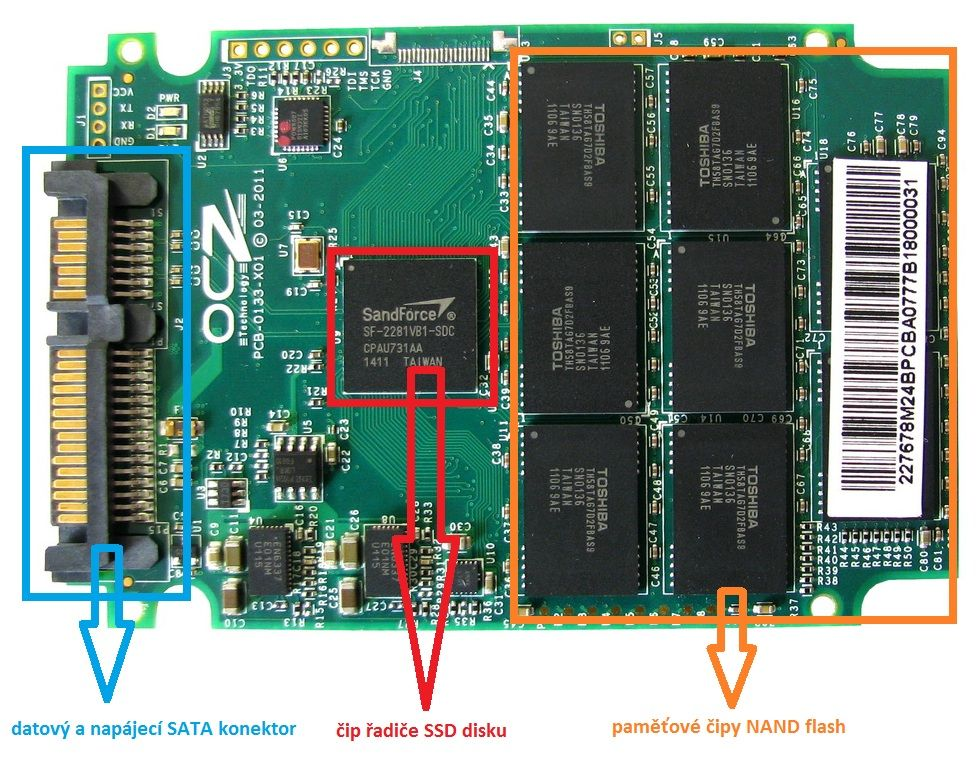
**HDD**

* Zápis probíhá pomocí zapisovacích hlav, které se těsně pohybují nad magnetickými plotnami
* Data se zapisují do sektorů



**SSD**

* Zápis probíhá do flash paměti pomocí elektronických obvodů



**USB**

* Stejný princip jako u SSD

**Mechaniky, čtečky karet**

**CD/DVD mechanika**

* Slouží k čtení či zápisu dat na optické nosiče – CD a DVD
* Připojení k základní desce **pomocí SATA kabelu** (novější typy) či PATA kabelu (starší typy)

**Čtečky karet**

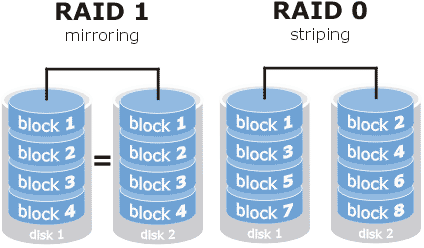
* Interní či externí
* Slouží k čtení a zápisu dat z/na různé paměťové karty (například karty do telefonů či fotoaparátů)

**Správa disku, diskové oddíly, disková pole**

**Diskové oddíly**

* Typicky se rozděluje disk na oddíly, například v případě přítomnosti jednoho disku v počítači, na kterém je nahraný operační systém, a ještě slouží pro úschovu dat
* Nebo je lze vytvořit ve chvíli, kdy chceme mít na jednom disku **více operačních systémů najednou**
* Možnost tvorby oddílu pro systém, pro data, pro swap prostor (dočasný úložný prostor, využívaný zejména v Linuxu)

**Disková pole**

* **RAID** - redundant array of independent disks
* Forma zálohy a zabezpečení dat proti selhání disku
* Několik typů:
  + Raid 0/1 - zrcadlení – stejná informace uložená na dvou discích; nevýhoda – přijde se o polovinu kapacity.
  + Výhoda – ochrana proti selhání či poškození jednoho z disků; RAID 5, 6, 10, Hotspare disk  
      
    

**Archivace, zálohování**

**Archivace**

* Archivace dat je proces, který slouží k dlouhodobému uchování dat, přičemž data jsou obvykle vhodně zabalena v archivu tak, aby byla uchována bez poškození.

**Zálohování**

* Záloha nebo záložní kopie (anglicky backup) je kopie dat uložená na jiném datovém nosiči (nebo i místě).
* Záložní data jsou **využívána v případě ztráty, poškození nebo jiné potřeby** práce s daty uloženými v minulosti.
* Zálohování probíhá nepravidelně (např. v domácnostech) nebo pravidelně podle rozvrhu (např. ve firmách).
* Zálohování může probíhat v těchto základních režimech:
  + **Online** – Proces tvorby zálohy počítače za jeho běžného chodu.
  + **Offline** - Zálohování je prováděno mimo běžný provoz počítače; obvykle se provádí za pomoci zavedení speciálního média.
* Pro různé podmínky se používají různé strategie zálohování.
* Volba správné strategie je závislá na tom, jestli je potřeba se zálohami pracovat velmi často nebo je naopak požadována maximální délka archivace zálohovaných dat.

**Další typy zálohování**

* **Nestrukturovaná**
  + Nestrukturovaným úložištěm může být větší množství disket, CD, DVD medií s minimem informací o záloze.
  + Tento způsob je nejjednodušší, ale není příliš oblíben u větších firem
* **Úplná + Inkrementální**
  + Má za cíl vytvořit více kopií zálohovaných dat vhodnějším způsobem.
  + Nejdříve je provedena úplná záloha všech dat. Posléze je prováděna inkrementální záloha (ukládány jsou pouze soubory, které se změnily od předešlé úplné nebo inkrementální zálohy).
  + Hlavní nevýhodou je, že při obnovení zálohy je potřeba pracovat s úplnou zálohou a následně se všemi inkrementálními zálohami až k požadovanému okamžiku zálohy, což může být velmi náročné na pracovní prostor.
* **Úplná + Rozdílová**
  + Rozdíl oproti předešlé metodě je v tom, že po úplné záloze se každá částečná záloha zachytí na všechny soubory vytvořené nebo změněné od vytvoření úplné zálohy, třebaže některé už jsou obsaženy v předešlé částečné záloze. Výhodou je, že obnova zahrnuje obnovení pouze poslední úplné zálohy.
* **Zrcadlová + Reverzně přírůstková**
  + Obsahuje zrcadlo reflektující stav systému po poslední záloze a historii přírůstkových záloh.
  + Výhodou je, že máme neustále k dispozici aktuální plnou zálohu a ukládáme pouze historii změn. Každé zálohování se automaticky promítá do zrcadla a soubory, které byly změněny, jsou přesunuty do přírůstkové zálohy.,
  + Tato metoda **se nehodí pro přenosná media**, protože každá záloha musí být provedena pomocí srovnání se zrcadlem.
* **Průběžná ochrana dat**
  + Využívá místo plánovaných periodických záloh okamžitý zápis každé změny do žurnálu změn (logu).
  + Provádí se ukládáním změněných bajtů nebo celých bloků dat místo ukládání celých změněných souborů. Průběžný záznam změn v žurnálu umožňuje získat obraz dat v minulosti.
  + Naproti tomu prosté zrcadlení dat na druhý disk (např. RAID 1) stav v minulosti nezachycuje.
* **Úplná záloha systému**
  + Metoda zálohuje obvykle celý počítač včetně operačního systému, vytváří obraz disku.
  + specializovaný software, jako je např. **Acronis True Image**

**Autorský zákon**

**121 / 2000 Sb.**

**Souvislost s informačními technologiemi a ukládáním dat**

* Autorský zákon upravuje tzv. autorská práva -> práva autorů k jejich dílům.
* Dílo je zákonem definováno jako literární a jiné dílo umělecké a dílo vědecké, které současně je jedinečným výsledkem tvůrčí činnosti autora a je vyjádřeno v jakékoli objektivně vnímatelné podobě.
* Dílem je např. dílo slovesné (např. **román**), grafické (např. **kresba**), hudební (např. **znělka**), choreografické (např. baletní choreografie), fotografické, audiovizuální (např. **film**), architektonické (stavba) nebo počítačový program.
* Autorským dílem není pouhý nápad nebo myšlenka, dílo musí být vyjádřeno tak, aby jej někdo jiný mohl vnímat.
* Od tohoto okamžiku je dílo chráněno autorským právem, není tedy nutná žádná registrace, jako např. u patentů.

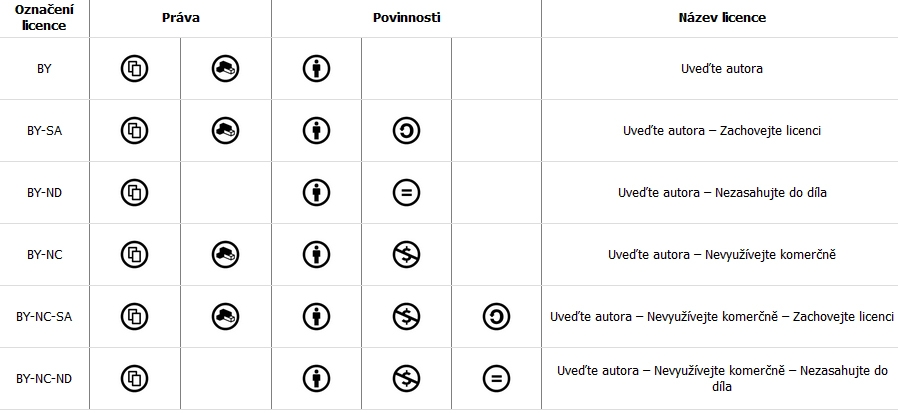
**Autorská práva se dělí na osobnostní a majetková.**

* + Osobnostní zahrnují především právo osobovat si autorství, rozhodnout o zveřejnění díla, právo na nedotknutelnost díla, zejména právo udělit souhlas ke změně nebo jinému zásahu do díla.
  + Majetková práva zahrnují hlavně právo dílo užít a udělit souhlas k užití.
* Autorský zákon stanoví, že nikdo nesmí užívat autorská díla bez souhlasu držitele autorských práv, není-li zákonem stanovena výjimka.
* Pojem „užívání“ znamená:
  + **rozmnožování** – zhotovování dočasných nebo trvalých, přímých nebo nepřímých rozmnoženin díla nebo jeho části,
  + **rozšiřování** – zpřístupňování díla v hmotné podobě převodem vlastnického práva,
  + **pronájem** – zpřístupňování díla za účelem hospodářského nebo obchodního prospěchu poskytnutím originálu nebo rozmnoženiny díla,
  + **půjčování** – zpřístupňování nikoli za účelem zisku
  + **vystavování** – umožnit dílo zhlédnout nebo jinak vnímat
  + **sdělování díla veřejnosti** – zpřístupňování díla v nehmotné podobě, živě nebo ze záznamu.
* Autorský zákon upravuje udělování licencí k užití díla a stanovuje také případy, kdy licence není třeba.
* Protože není možné, aby si každý autor ohlídal, zda jeho dílo není užíváno bez jeho souhlasu, ustanovuje zákon tzv. **kolektivní správu autorských práv.**
* Pověřená sdružení pak vybírají poplatky za užívání děl (např. od rozhlasových stanic) a zisk rozdělují mezi autory.

**Softwarové licence (příklady, charakteristika)**

**Licence**

* Softwarová licence je v informatice právní nástroj, který umožňuje používat nebo redistribuovat software, který je chráněn zákonem.
* Softwarové licence jsou obvykle velmi komplikované, takže jejich tvorbu přenechávají autoři softwarovým právníkům.
* Některé licence jsou předem vytvářeny tak, aby je mohli autoři snadno použít.
* Volbu licence provádí obvykle autor softwaru, někdy k ní doplňuje vlastní dodatky, případně používá více licencí, přičemž volba vhodné licence je ponechána na uživateli.



* **Často se používá licence ve formě EULA.** 
  + Autor může licenci měnit. V případě kolektivního díla se obvykle vyžaduje souhlas všech autorů, a proto se často autoři záměrně vzdávají autorství ve prospěch třetí strany, která pak stanovuje licenci.
* **EULA** (End-User-License-Agreement) je licence pro koncového uživatele softwaru určující, co uživatel smí a nesmí dělat.
  + Je možné, aby byl zdrojový kód open source, ale výsledný produkt už spadá pod EULA, kde se hovoří o zákazu editace a šíření tohoto programu (např. Mozilla Firefox).
  + Některé programy v závěru EULA při instalaci zmiňují instalaci dodatečného škodlivého programu, jedná se o tzv. grayware.
  + **Grayware** je označení pro software, který záměrně obtěžuje, například spyware a adware.

**Příklady licencí**

* **BSD licence** – licence pro svobodný software, mezi kterými je jednou z nejsvobodnějších.
  + Umožňuje volné šíření licencovaného obsahu, přičemž vyžaduje pouze uvedení autora a informace o licenci, spolu s upozorněním na zřeknutí se odpovědnosti za dílo.
* **GNU General Public** **License** – licence pro svobodný software, původně napsaná Richardem Stallmanem pro projekt GNU.
  + GPL je nejpopulárnějším a dobře známým příkladem silně copyleftové licence, která vyžaduje, aby byla odvozená díla dostupná pod toutéž licencí.
  + V rámci této filosofie je řečeno, že poskytuje uživatelům počítačového programu práva svobodného softwaru a používá copyleft k zajištění, aby byly tyto svobody ochráněny, i když je dílo změněno nebo k něčemu přidáno.
  + Toto je rozdíl oproti permisivním licencím svobodného softwaru, jejímž typickým případem jsou BSD licence

**Způsoby distribuce**

* **Adware** – programy, které mají integrovanou nepříjemnou reklamu
* **Donationware** – autor žádá uživatele o příspěvek na své konto nebo na dobročinné účely
* **Demo** – funkčně nebo časově (trial) omezená verze komerčního softwaru, která se šíří zdarma
* **Freeware** – software, jehož užívání je naprosto zdarma
* **Shareware** – software, který lze volně distribuovat a zdarma vyzkoušet, pro další používání je třeba zaplatit

## Přenosné technologie

## Typy a charakteristika přenosných paměťových médií, mechaniky, čtečky

**CD**

* + Kompaktní disk, určený pro ukládání digitálních dat.
  + Data jsou uložena ve stopách na jedné dlouhé spirále začínající ve středu média, která se postupně rozvíjí až k jeho okraji.
  + Kapacita: 700 MB

**DVD**

* + Digitální optický datový nosič, který může obsahovat filmy ve vysoké obrazové a zvukové kvalitě nebo jiná data.
  + Má větší kapacitu než CD.
  + Kapacita: 4,7 GB

**Blue-ray**

* + Třetí generace optických disků.
  + Data se ukládají ve stopě tvaru spirály 0,1 mm pod povrch disku. Ke čtení disku používá modrý paprsek.
  + Kapacita: 25 - 100 GB

**Flashdisk**

* + USB flash paměť, někdy též USB flash disk, paměťové zařízení, používané převážně jako náhrada diskety.
  + Je vybaveno pamětí typu flash, která umožňuje uchování dat i při odpojení napájení.
  + Kapacita: až do pár TB

**Externí disk**

* + Externí disk je pevný disk, který se používá pro přenos větších objemů dat. Standardně je napájen pomocí USB portu.
  + Někdy je potřeba externí napájení.

**Paměťová karta**

* + Paměťová karta je elektronické zařízení, sloužící k ukládání dat.
  + Používá se v digitálních fotoaparátech, PDA, laptopech, mobilních telefonech, přehrávačích a jiných elektronických zařízeních.
  + Je to malé, kompaktní zařízení s relativně vysokou kapacitou, je odolné vůči magnetickým a elektrickým polím.

## Typy mobilních zařízení

**Mobilní telefon**

* + Mobilní telefony slouží k navázání a přijmutí hovoru po telefonní síti
  + Dnes se teflon využívá mnohem více funkcí jako třeba připojení k internetu, SMS zprávy nebo různé aplikace.
  + Rozdělujeme je na Celulární, Satelitní a bezdrátové telefony.

Celulární

Jedná se o nejrozšířenější skupinu mobilních telefonů

Využívá GSM síť, aby dokázaly posílat SMS, WAP …

Satelitní

Dá se pomocí něj uskutečnit hlasový hovor, pomocí telekomunikačního satelitu. Funguje i v prostorech, kde by Celulární telefon nefungoval.

Bezdrátové

Umožňují malý pohyb po malém prostoru, komunikuje se svojí stanicí, která je připojená telefonní linkou.

3 Největší prodejci telefonů jsou Samsung, Apple a Huawei.

**SmartWatch**

* + Chytré hodinky jsou pomocí bluetooth připojeny k mobilnímu telefonu, mohou zobrazovat notifikace, zprávy, přístup ke kalendáři, odmítání a přijímání hovoru nebo ovládat další aplikace.
  + Výhody jsou, rychlejší přístup, delší výdrž baterie, lepší manipulace, využití při sportu …
  + Existují i dětské hodinky, které například kontrolují GPS polohu dítěte.

**GPS navigace**

* + Zařízení schopné určit polohu a čas, pomocí družic na oběžné dráze.
  + Využívá se v automobilech, pro turismus, ve vojenství, letectví, námořnictví…
  + Navigace dokáže vypočítat optimální trasu – nejrychlejší, nejkratší, placené trasy… Umí trasu přepočítávat
  + Důležité je mít stažené aktuální mapy

## Srovnání notebooku a stolního PC

* + Notebooky používáme ke stejným činnostem jako stolní počítače
  + Jejich výhodou je jejich přenosnost, velikost a hmotnost. Notebooky v sobě mají zabudované komponenty, LCD displej, klávesnici i myš (touch-pad).
  + Samozřejmě myš a klávesnici lze připojit externě jak u notebooku, tak u stolního PC.
  + Nevýhodou notebooků je to, že nedosahují stejných výkonů jako stolní počítač za stejnou cenu. Na vině je to že nové komponenty spotřebují velké množství energie, se kterou se v noteboocích šetří.
  + Notebook i stolní PC používají stejné vnější porty.

## Dokovací stanice, nabíječky

**Dokovací stanice**

* + Je stálým prostředníkem mezi počítačem a všemi připojenými periferiemi.
  + Je tak možné použít více USB konektorů, případně další rozhraní, která výrobce na notebook neumístil, protože na to nebyl prostor.
  + Dělení:

1. Univerzální – připojení pomocí USB

2. Specifické – pouze pro určité typy notebooků

* + Typy:

Port replikátory

Možnost snadného připojení k řadě externích zařízení.

Uživatel, který je většinou na cestách, se tak nemusí zvlášť připojovat k monitoru, klávesnici, myši, tiskárně, ale jen k jedinému replikátoru.

Specifická dokovací stanice

Musí přesně pasovat k určitému typu notebooku.

Výhoda – zaručená kompatibilita.

Nevýhoda – vyšší cena.

Univerzální dokovací stanice

Zařízení od různých dodavatelů, kteří jsou nezávislí na jednotlivých výrobcích notebooků.

Připojíte nejčastěji pomocí USB.

Výhody – širší spektrum podporovaných notebooků a nižší cena.

Nevýhody – může dojít k neúplné kompatibilitě.

**Nabíječky a baterie**

* + Je technické zařízení na opakované uchovávání energie, obvykle elektrické.
  + Akumulátor je sekundární článek, který je potřeba nejdříve nabít a teprve potom je možné jej použít jako zdroj energie.
  + Na rozdíl od sekundárních článků (akumulátorů) primární články dodávají energii ihned po svém sestavení a zpravidla je není možné dobíjet, například zinkouhlíkové baterie.
  + Dají se dělit hodně způsoby

## Synchronizace, bezdrátové komunikační technologie

**Synchronizace**

* + Obecně se jedná o propojení určitých věcí v jeden celek
  + Se synchronizací se můžeme setkat prakticky v jakémkoliv zařízení
  + V IT zaměření se synchronizace myslí jako propojení dvou zařízení v jeden celek

**Bezdrátové komunikační technologie**

* + - * + **Bluetooth**

Standard který propojuje 2 zařízení

Vytvořen v roce 1994 firmou Ericsson

Název podle dánského krále Haralda Modrozuba

Je součástí spousty zařízení (tiskárny, telefony, notebooky….)

Rozdělení do tříd podle dosahu

Pracuje na stejné frekvenci jako WI-FI 2,4 GHz, nově 5 GHZ

* + - * + **Wi-Fi**

Název původně neměl znamenat nic (wireless fidelity)

Jedná se o bezdrátové připojení do sítě

Pracuje v pásmu 2,4GHz nebo 5 GHz

V dnešní době se Wi-Fi nachází téměř všude

Routery parametry (dosah, pásma, přenosová rychlost)

**Operační systémy pro mobilní zařízení**

Nejpoužívanější operační systémy na mobilních telefonech jsou Android, který byl v roce 2016 na více jak 86% zařízení a iOS, jež byl druhý s necelými 13%, zbytek tvořili ostatní operační systémy jako například Windows Phone nebo Blackberry.

* + - * + **Android**

vychází z Linuxu a je opernsource.

Nejnovější verze je Android 9.0 Pie, velká přizpůsobitelnost, velké množství aplikací.

* + - * + **iOS**

Nedá se tolik přizpůsobit potřebám uživatele, je lépe zabezpečený, má jednotný design na všech zařízení, je synchronizovaný s ostatními zařízeními, aplikace se dají stahovat jen z Appstoru.

## Audiotechnologie a videotechnologie

## Audiotechnologie

* Definujte zvuk a popište jeho digitalizaci

**Zvuk**

* Je to mechanické kmitání hmotných částic, které se šíří v prostředí.
* Zdrojem je kmitající těleso - ladička, struna nebo sloupec kmitajícího vzduchu (hlasivky).
* Frekvence vlnění, které může vnímat člověk je 16 Hz - 20 000 Hz.
* Frekvence pod 16 Hz --> infrazvuk.
* Frekvence nad 20 kHz --> ultrazvuk

**Digitalizace**

* Převod analogového signálu na digitální signál.
* Na signál se použije vzorkování a hned poté kvantování.
* V digitalizace vždy dochází ke ztrátě informací (můžou být ovšem velmi malé).
* Vzorkování (Sampling)

Časová osa se rozdělí na rovnoměrné úseky, odkud se vezme vždy jeden vzorek.

Když se děje vzorkování, ztrácíme vlastně nějaké informace o grafu, jelikož v úseku kde nemáme vzorek, mohlo dojít ke kolísání.

**Kvantování**

* Protože počítač může vyjádřit čísla jen s omezenou přesností je třeba vzorky upravit i na svislé ose napětí. Hodnota vzorku se dá vyjádřit pouze v kvantech.
* Veličina na svislé ose může nabývat pouze celočíselných hodnot. Aby se zjistilo, kam která hodnota patří je třeba rozdělit prostor kolem hodnot na toleranční pásy.
* Když vzorek padne do daného pásu, je mu přiřazena daná hodnota.
* Většinou se vzorky liší od vzorků po kvantování.
* Velikost chyb se pohybuje mezi +1/2 až -1/2. kvantizační úrovně.

**Zvuková karta – princip (A/D a D/A převodníky), konektory**

* Zvuková karta je přídavná karta pro zvukový vstup a výstup.
* **A/D převodník**

Slouží pro převod analogového signál na digitální.

Využívá k tomu metody vzorkování a kvantování.

* **D/A převodník**

Slouží k převodu digitálního signálu na analogový.

Používá se např. v přehrávačích (CD, MP3,…), zvukových kartách v počítačích.

V každém telefonu je A/D i D/A převodník.

* **Konektory (obrázek na konci)**

Růžový = Analogový vstup na mikrofon

Modrý = Analogový vstup

Zelený = Analogový výstup pro přední reproduktor/sluchátka

Černý = Analogový výstup pro zadní reproduktory

Šedý = Analogový výstup pro boční reproduktory

Oranžový = Digitální výstup

**Příklady zařízení pro práci se zvukem a jejich charakteristika**

* **Sluchátka**

Specifické malé reproduktory, které se umisťují přímo na hlavu nebo do uší.

Většinou se připojují přes JACK 3.5 mm, ale můžou i bezdrátově.

Circumaurální sluchátka

Obepínají celý ušní boltec.

Vytváří velmi dobrou mikroakustiku.

Dokážou mít velmi dobrý a kvalitní zvuk.

Supraaurální sluchátka

Spočívají přímo na uchu.

Používají se převážně u přenosných nebo domácích modelů.

Intramurální sluchátka

Pecky

Spočívají na vyústění zvukovodu.

Nevýhodou je nekvalitní zvuk.

Slabé výšky a basy.

Špunty

Strká se do zvukovodu.

Kvalitnější zvuk než u pecek.

Za kvalitu je třeba platit.

**Parametry**

Harmonické zkreslení - udává se v % a větší hodnota je horší. Indikuje přítomnost nežádoucích frekvencí ve zvuku, které měniče sluchátek vytvoří.

Akustický tlak - udává se v dB a větší hodnota je lepší. Akustický výkon sluchátek. Práh slyšitelnosti = 0 dB.

Impendance - udává se v ohmech. Určuje, jak moc budou sluchátka brát energii ze zdroje. Větší hodnota je horší, ale hraje velkou roli ve kvalitě zvuku.

* **Reproduktory**

Mění elektrický signál na mechanický pohyb membrány.

Dají se rozdělovat podle mnoha věcí (podle vyzařování, pohonu, kmitočtového rozsahu nebo počtu pásem)

**Parametry**

Frekvenční rozsah reproduktorů - udává od jak nízkých až po jak vysoké frekvence bude reproduktor hrát.

Výkon - pouze v hodnotách RMS (jedině on má svůj standart). Nepoužívat MAX, sinus, P. M. P. O.

Impendance - udává se v ohmech. Reprosoustava musí mít vyšší impendaci než její zesilovač.

Citlivost - Rozdíl mezi nejvyšší a nejslabší složkou signálu. Jako třeba mezi šepotem a křikem.

* **Mikrofon**

Převádí akustický signál na elektrický

Dělení:

Ruční (nejpoužívanější, zpěváci)

Klopák (nenápadný mikrofon v pořadech)

Stolní mikrofony

Poté se dělí dle směrové charakteristiky - Všechny tyto typy se liší podle toho, odkud a v jak velké míře slyší akustický signál.

Všesměrová

Kardioidná

Hyperkardioidní

Osmičková

Úzce směrová

**Parametry**

Citlivost - Jednotky jsou mV/Pa. Čím vyšší, tím vyšší je i odstup výsledného užitého signálu od šumu.

Šum - Odstup šumového napětí v dB.

Kmitočtová charakteristika - Udává závislost výstupního napětí na kmitočtu.

**Typy zvukových souborů**

* WAV - bezeztrátový, vysoká kvalita zvuku, zabírá hodně místa na disku
* WMA - dříve ztrátový, od verze 9.1 podporuje vícekanálové bezztrátové audio
* MP3 - ztrátový, maximální možný bitrate 320 kbps
* OGG - kontejnerový formát, používá se pro Vorbis, Speex nebo FLAC
* MIDI - tyto soubory slouží ke komunikaci hudeb. nástrojů cizích značek
* FLAC - nejpoužívanější bezztrátový formát, nejčastěji používaný na ripování CD bez ztráty kvality

## Videotechnologie

**Princip videa, velikost, komprese**

* Technologie pro zaznamenávání, přehrávání, přenos a obnovu pohyblivých obrázků používající elektrické signály nebo digitální média.
* Kvalita je závislá na metodě zachycování a ukládání obrazu.
* Nejdůležitějším kritériem je formát uložení, různé formáty mají různý poměr kvalita/objem.
* Velikost se udává v MB, ale může se pohybovat od řádů kB až po stovky GB.
* Komprese

Je to zmenšení objemu dat a zachování obsažených informací.

Kompresní poměr = podíl velikosti nekomprimovaných dat k velikosti komprimovaných dat

Ztrátová

Při kompresi jsou některé informace nenávratně ztracené.

Použití - komprese zvuku nebo videa.

Bezeztrátová

Soubor jde po dekompresi převést do původního stavu.

Použití - počítačová data, text, výsledky měření.

**Kodeky, typy videosouborů**

* Kodek je zařízení nebo počítačový program, který dokáže transformovat datový proud nebo signál.
* Video kodeky

Bezeztrátové - Huffyuv, Lagarith, LCL

Ztrátové - Divx, Xvid, Windows Media Video

* Typy videosouborů

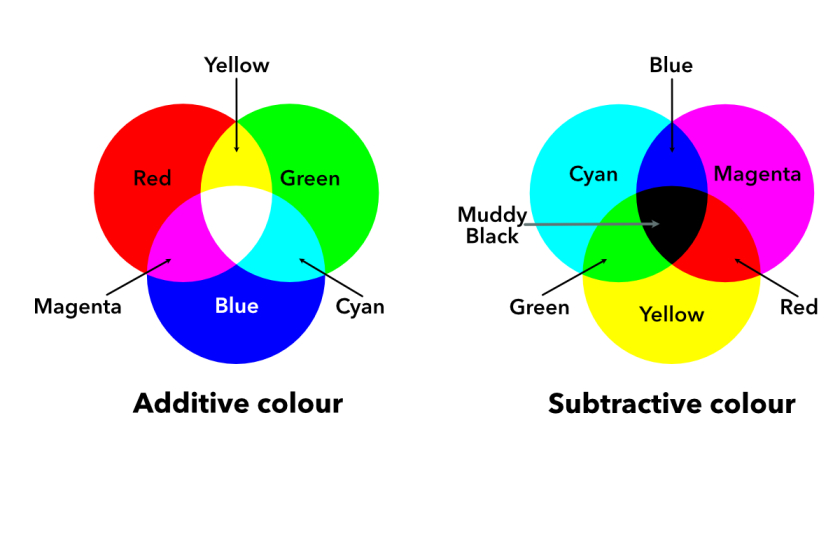
AVI - mohou obsahovat zvukovou i video stopu, což umožňuje synchronní přehrávání videa a zvuku.

MP4 - Oproti AVI může MP4 obsahovat menu, více titulků i zvukových stopa dokonce i 3D objekty. Umožňuje také bezproblémové streamování videa.

WMV - komprimovaný souborový videoformát pro několik proprietárních kodeků od polečnosti Microsoft.

MPEG, MOV, MTV,…

**Barevné modely**

* RGB

Červená, zelená, modrá.

Aditivní způsob míchání barev.

Systém pro míchání barev.

Monitory, projektory.

* CMYK

Azurová, purpurová, žlutá, černá.

Subtraktivní způsob míchání barev.

Způsob pro míchání barev.

Tiskárny.

**Snímače CCD, CMOS**

* Slouží k snímání optické obrazu a jeho převádění do elektrických signálů.
* Používají se v digitálních fotoaparátech, videokamerách, televizních kamerách, ...
* CCD - převádní obraz do analogové podoby a pro digitalizaci je potřeba A/D převodník.
* CMOS - převádí obraz přímo do digitální podoby.

**Zobrazovací technologie – CRT, LCD**

* CRT

Vysoký kontrastní poměr

Nízká odezva

Vysoká spotřeba energie

Zastaralý typ

Velmi prostorný

* LCD

Kompaktní a lehký

Nízká energetická spotřeba

Náchylný v nízkých a vysokých teplotách

IPS a VA mají široké pozorovací úhly

**Monitory, projektory – charakteristika, parametry**

* Monitor

Základní výstupní zařízení.

LCD, LED, OLED, CRT

Parametry

Úhlopříčka, rozlišení obrazovky, doba odezvy, obnovovací frekvence,…

* Projektor

Zařízení, které slouží k promítání z počítače nebo televizoru.

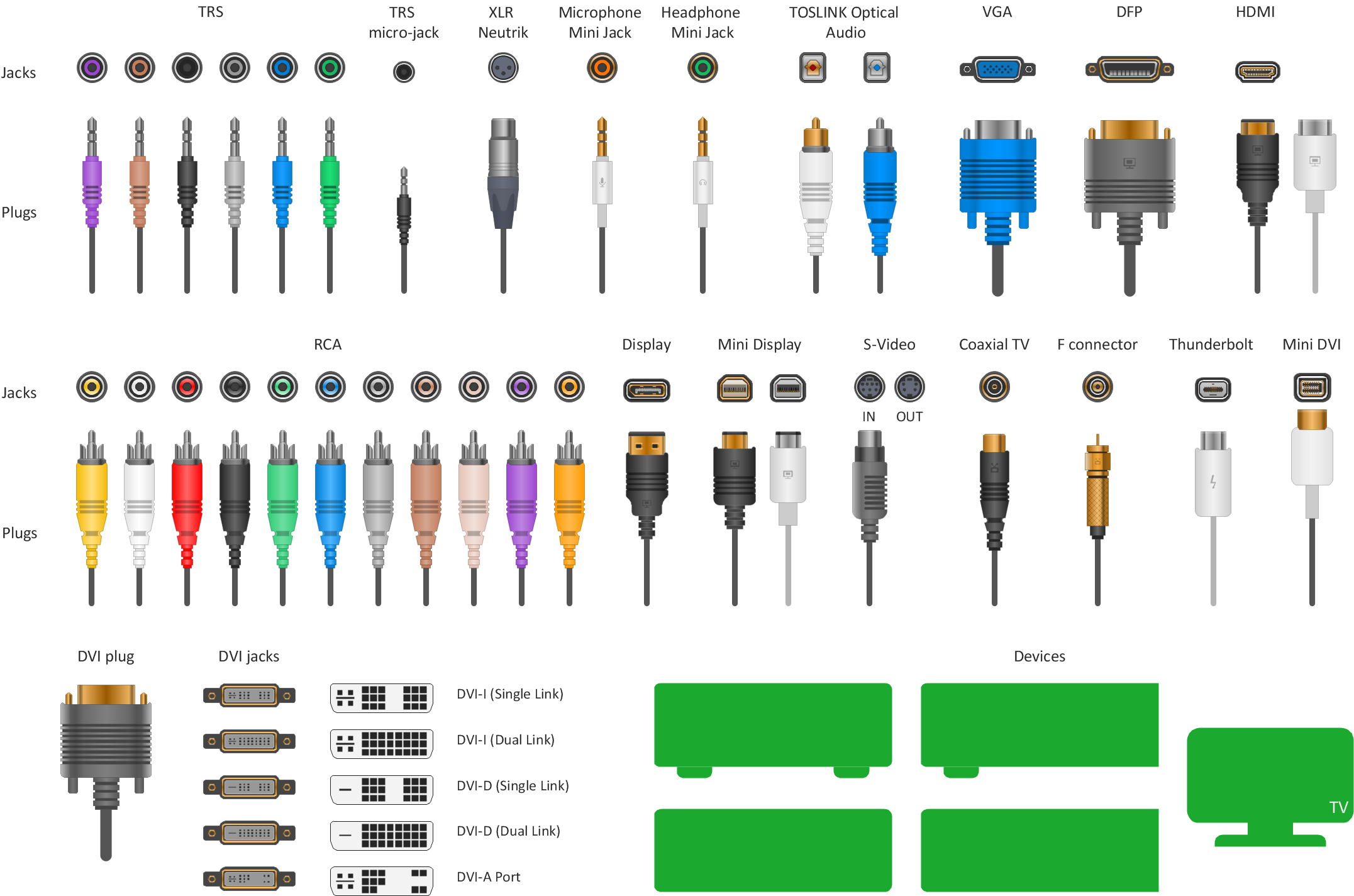
Využívá světlo k promítání.

Používá LCD nebo DLP technologie.

Parametry

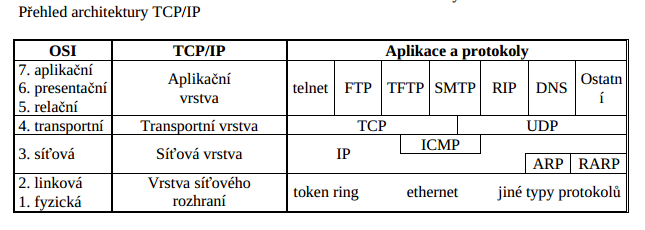
Projekční vzdálenost, rozlišení, vstupy, Wi-Fi, ostatní funkce,…

**Další zařízení pro práci s videem**

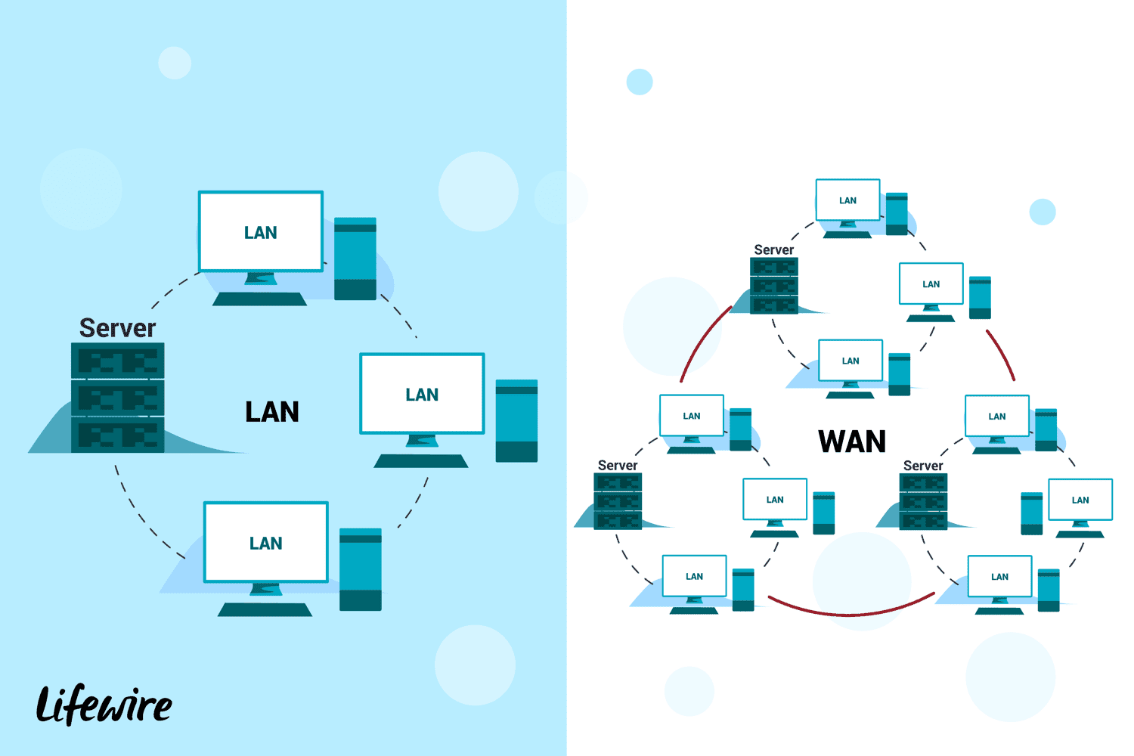
* Webová kamera - vstupní zařízení, které pořizuje aktuální obraz do počítače.
* Kamerový systém - užití kamer ke sledování prostoru a zobrazování a archivaci záběrů.
* Zrcadlovka - fotoaparát, používající k zobrazení scény v hledáčku zrcadlovky.
* Video rekordér - slouží k záznamu televizního vysílání.

# Typy sítí, standardizace síťové komunikace

**Typy sítí**

* Počítačová síť vznikne ve chvíli, kdy dva (někdy se říká minimálně tři) nebo více počítačů propojíme dohromady pomocí telekomunikačního systému za účelem sdílení zdrojů.
* V praxi je dnes nejrozšířenější síť založena na technologii ethernet a používá **protokol TCP/IP.**
* Počítačové sítě se dělí podle řady kritérií. 

**Podle velikosti/měřítka**

* **LAN**
  + Lokální počítačová síť se vyznačuje tím, že počítače jsou propojeny na menším geografickém území (tedy v rámci firmy, budovy, místnosti atp.).
  + V rámci LAN se nejvíce používá přepínaný ethernet nebo WiFi (**IEEE802.11**).
  + Infrastruktura je většinou tvořena metalickými kabely a případně optickou páteří.
  + LAN může být samostatná síť, které propojuje řadu zařízení, ale v dnešní době je většinou propojena do internetu, tedy WAN sítě.
* **WAN**
  + WAN je komunikační síť, která pokrývá rozsáhlé území, jako je spojení zemí či kontinentů.
  + Obecně můžeme říct, že jednotlivé LAN sítě se propojují přes WAN síť, aby se zajistila komunikace na velké vzdálenosti.
  + Tímto způsobem pracuje internet jako nejrozsáhlejší a nejznámější WAN.
  + Nejvíce se dnes asi používá technologie **Frame relay** případně ATM.
* **WLAN**
  + Bezdrátová lokální síť je opět obdobou běžné LAN, ale jednotlivé prvky nejsou fyzicky propojeny drátem (metalikou či optikou), ale jsou propojeny bezdrátově.
  + Využívají se rádiové vlny a určitá modulace pro přenos dat.
  + Výhoda bezdrátu je jasná pro mobilní zařízení.
  + Nevýhodou je například to, že se špatně omezuje šíření signálu, a případný útočník nemusí získat přímo fyzický přístup k zásuvce, jako v případě drátových sítí.
* **MAN**
  + Síť, která spojuje jednotlivé LAN, ale nepřekračuje hranice města či metropolitní oblasti, se označuje jako **metropolitní síť** – MAN.
  + V rámci MAN se často používá bezdrátové spojení nebo optická vlákna.
  + MAN může být vlastněna jednou organizací, ale většinou se jedná o propojení několika nezávislých objektů.
  + Můžeme mít například několik poboček firmy v jednom městě propojených do MAN sítě.  
    

**Podle síťové topologie**

* **Hvězda (star) - Hub and Spoke**
  + Hvězda je dnes nejpoužívanější topologie pro ethernet. Je zde centrální prvek, který realizuje propojení zařízení, a do něj jsou připojena jednotlivá zařízení.
  + Jako centrální prvek slouží hub nebo switch, ale z jiného pohledu se může jednat i o router.
  + Obdobná je Rozšířená topologie hvězda, která vznikne, když několik samostatných hvězd propojíme dohromady přes centrální prvky.
  + V praxi se dnes často používá tří vrstvá topologie (případně dvouvrstvá, kdy se jedna vrstva vynechá) využívající propojení do hvězdy.
    - V nejvyšší vrstvě je jeden nebo dva (pro redundanci) switche, tzv. jádro (core).
    - Druhá vrstva obsahuje několik switchů připojených do jádra a zvaných distribuční (distributed).
    - Poslední vrstva, připojená do distribuční, je přístupová (access) a do ní se připojují stanice a servery.
  + Některé důležité servery se mohou připojit přímo do distribuční vrstvy.
  + Zapojení, kdy zdvojíme centrální prvek, se také označuje jako Dvojitá hvězda.
  + **Topologie hvězda se také označuje jako zapojení Hub and Spoke.**
  + Zde není použit termín hub jako aktivní síťový prvek, ale je to odvozeno od označení pro kolo u vozu, kde máme střed (Hub) a paprsky/loukotě (Spoke).
  + Tento termín se používá také u routování nebo VPN, kdy Hub je centrála/centrální prvek a Spoke jsou pobočky.
* **Kruh (ring)**
  + V kruhové topologii je každý uzel připojen ke dvěma sousedním a dohromady tvoří kruh.
  + Standardně existuje pouze jedna cesta mezi dvěma uzly.
  + Rozšířením je, že komunikace probíhá ve směru i proti směru hodinových ručiček.
  + Používá se pro síťové technologie FDDI a Token Ring.
  + Principem sítě Token ring je předávání vysílacího práva pomocí speciálního rámce (tzv. tokenu) mezi adaptéry, zapojenými do logického kruhu.
  + **Fyzicky je síť zapojena do hvězdicové topologie, ale centrální hub slouží pouze jako spoj pro uzly v sousedních ramenech hvězdy.**
* **Sběrnice (bus)**
  + Sběrnice byla používána v prvních dobách ethernetu a realizovala se pomocí koaxiálního kabelu a BNC konektorů, na konci musel být vždy terminátor.
  + Všechna zařízení jsou zapojena na společnou sběrnici.
  + V sítích se od této technologie ustoupilo a dnes se používá převážně zapojení do hvězdy.
* **Mřížka (mesh)**
  + V topologii mesh jsou uzly propojeny s více sousedy.
  + Buď se může jednat o Full Mesh (plnou mřížku), kdy je každý uzel spojený se všemi ostatními, takže může komunikovat s každým přímo a v případě výpadku nějaké linky může jednoduše nalézt cestu.
  + Pří více uzlech se jedná o složité a drahé zapojení.
  + Nebo o Partial Mesh (částečnou mřížku), kdy některé uzly jsou přímo spojeny (point-to-point) s více jinými uzly.
  + Logická Full Mesh se používá například pro routování BGP protokolem.  
    **Dále se Mesh topologie používá v některých WiFi sítích.**



**Standardizace síťové komunikace**

* Standardizace je proces sjednocení pomocí zavádění standardů.
* Vede ke koordinaci, kompatibilitě a opakovatelnosti v kvalitě výroby. Standardy se zavádějí buď centrálně, což je způsob, který převažuje, nebo jde o ustálený převažující standard, který vznikl z rozšířeného užívání.

**IETF**

* Vyvíjí a podporuje internetové standardy.
* Zabývá se především standardy TCP/IP a internetovými protokoly.
* Jedná se o otevřenou organizaci vydávající standardy, nevyžadující žádné formální členství nebo členské požadavky.

**ISO**

* Mezinárodní organizace pro normalizaci
* Zabývá se tvorbou mezinárodních norem ISO a jiných druhů dokumentů ve všech oblastech normalizace kromě elektrotechniky.
  + Jsou to např.:
  + TS – technické specifikace
  + TR – technické zprávy
  + PAS – veřejně dostupné specifikace
  + TTA – dohody o technických trendech
  + IWA – dohody z pracovní konference průmyslu
  + Pokyny ISO

**IEEE**

* Mezinárodní nezisková profesní organizace usilující o vzestup technologie související s elektrotechnikou.
* IEEE je jedna z předních standardizačně-vývojových organizací na světě. Vykonává vývojové a údržbové funkce, a to skrze IEEE Standards Association (IEEE-SA).
* **Jedním z důležitých standardů IEEE je skupina standardů IEEE 802 LAN/WAN**, která zahrnuje IEEE 802.3 Ethernet a IEEE 802.11 Wireless Networking standardy.

**IANA**

* Organizace, která dohlíží celosvětově na přidělování IP adres, správu kořenových zón DNS, definování typů medií pro MIME a další náležitosti internetových protokolů.
* IANA je plně zodpovědná za přidělování celosvětově jedinečného jména a čísla internetových protokolů, jež jsou zveřejňovány jako RFC dokumenty.
* Při plnění této úlohy úzce spolupracuje s Internet Engineering Task Force (IETF) a Request for Comments (RFC).

## Přístup k síti

**Internet**

* Internet je celosvětový systém navzájem propojených počítačových sítí („síť sítí“), ve kterých mezi sebou počítače komunikují pomocí rodiny protokolů TCP/IP
* Společným cílem všech lidí využívajících internet je bezproblémová komunikace (výměna dat)
* Nejznámější službou poskytovanou v rámci internetu je WWW (kombinace textu, grafiky a multimédií propojených hypertextovými odkazy) a e-mail (elektronická pošta), avšak nalezneme v něm i desítky dalších

**Způsoby připojení**

* V současnosti existuje několik možností pro připojení počítače k internetu:
* Připojení pomocí pevné linky
  + - **Dial-up**

Neaktuální – u dial-upu se platilo za čas, který jste na internetu strávili a tarify byly stanoveny podle časových „pásem“

Ta byla celkem dvě: 07:00 – 19:00, což bylo to dražší pásmo a po zbytek času se účtovala levnější sazba, která činila asi 16Kč za hodinu

Klasické analogové vytáčené spojení (jinak též známé jako dial-up) **je omezeno maximální možnou rychlostí, která činí v ideálním případě 56kb/s**, což je dáno omezením frekvenčního pásma v analogových ústřednách klasických telefonních linek

K tomuto typu připojení musíte mít speciální telefonní modem, který vytočí číslo některého z providerů, a následně už můžete brouzdat po internetu

Zpočátku se platilo i za samotné připojení, ale posléze se objevili poskytovatelé, kteří poskytovali připojení zdarma a platilo se pouze za „provolané“ minuty

**Výhody:** připojit se lze odkudkoliv, kde je k dispozici telefonní linka, minimální pořizovací náklady

**Nevýhody:** dosti vysoká provozní cena, při připojení nelze telefonovat, nebezpečí různých druhů malwarů a dialerů

* + - **IDSN (Integrated Services Digital Network)**

Tento druh připojení (tedy IDSN) je jakousi nadstavbou na klasiku v podobě dial-upu, který se plošně a ve velké míře využíval. Přenosová rychlost tohoto typu připojení je 64kb/s (128kb/s

V dnešní době už o ISDN není téměř slyšet, protože bylo takřka kompletně vytlačeno konceptem ADSL

**Výhody:** možnost volat a brouzdat na internetu zároveň

**Nevýhody:** dosti vysoké náklady na pořízení a provoz

* + - **ADSL (Asymetric Digital Subscriber Line)**

Tento typ připojení je v současnosti **nejpoužívanější variantou DSL internetu**

ADSL plynule vystřídala zastaralé metody Dialu a ISDN

Digitální ústředny – vysoká přenosová rychlost

Běžné rychlosti tohoto připojení nabízené internetovými providery se totiž dnes pohybují **od 3 do 10Mbit/s.**

Systém ADSL internetu se vyznačuje asymetrickým připojením, kdy je rychlost dat směřujících k uživateli vyšší než rychlost dat od uživatele směrem do internetu

-> problém nastává u aplikací, které více vytěžují upload, jako jsou například nejrůznější IM klienty typu Skype apod.

**ADSL existuje ve dvou variantách, a to:**

ADSL - asymetrická linka, využívající měděné kroucené dvojlinky, sdílení s telefonní linkou, až 8/1 Mbit/s

DSL2+ - vylepšená verze ADSL, až 25/4 Mbit/s

**Výhody:** velmi vysoké procento pokrytí, příznivá cena, velmi rychlý internet

**Nevýhody:** nutnost mít zavedenou pevnou linku

* + - **Připojení přes mobilní technologie**

**O kvalitě připojení rozhoduje:**

agregace (tj. kolik uživatelů sdílí jednu linku)

doba odezvy (dlouhé odezvy mohou mít negativní vliv např. při internetové telefonii)

rychlost připojení poslední míle

technologie použitá pro připojení "poslední míle" (Česká republika)

* **Možnosti připojení v ČR**
  + - Drátové a bezdrátové, mobilní  
      <https://www.dsl.cz/jak-na-to/jak-se-pripojit-k-internetu>
* **Způsoby připojení**
  + - **DSL**

DSL (Digital Subscriber Line) je technologie, která umožňuje využít stávající vedení telefonu nebo kabelové televize pro vysokorychlostní přenos dat.

Využívá telefonní rozvody plochým nekrouceným kabelem, kroucenou dvojlinku nebo koaxiální kabel kabelové televize.

Obecně platí, že čím větší vzdálenost od ústředny nebo méně kvalitní vedení, tím nižší maximální dosažitelná rychlost.

Pro běžné domácí nasazení se obvykle využívá asymetrická varianta (ADSL), kde je vyšší přenosová rychlost ve směru k zákazníkovi (anglicky download) a nižší rychlost směrem od zákazníka (anglicky upload).

**Rozdělení služeb DSL:**

ADSL - asymetrická linka, využívající měděné kroucené dvojlinky, sdílení s telefonní linkou, až 8/1 Mbit/s

ADSL2+ - vylepšená verze ADSL, až 25/4 Mbit/s

SHDSL - symetrická linka, max. 4,5 Mbit/s při použití 2 párů kroucené dvojlinky

SHDSL.bis - vylepšená verze SHDSL, max. 5696 kbit/s na jednom páru, podpora až 4 párů

VDSL2 - symetrická linka, max. 100 Mbit/s

SDSL - symetrická linka, max. 2,3 Mbit/s

HDSL - symetrická linka, max. 4 Mbit/s při použití 2 párů kroucené dvojlinky

VDSL - symetrická linka, max. 36 Mbit/s

CableDSL - využívající koaxiální rozvody kabelových televizí

* + - **Bezdrátové připojení**

**Telefonní síť**

Telefonní síť je telekomunikační síť, určená především pro telefonní hovory mezi dvěma nebo více účastníky.

**Existují různé druhy telefonních sítí:**

**Veřejná** (komutovaná) telefonní síť (hovorově pevná telefonní síť), která slouží k poskytování telefonních služeb účastníkům z řad veřejnosti; umožňuje přenos mluvené řeči, ale i jiných forem komunikace, jako jsou faksimilní a datové přenosy mezi koncovými body sítě.

Telefony jsou účastnickým vedením propojeny s telefonní ústřednou .

**Mobilní telefonní síť** využívá rádiové vlny pro připojení mobilního telefonu, takže jeho uživatel se může pohybovat po celé oblasti, která je pokryta signálem sítě.

**Soukromá** (privátní) telefonní síť, ve které je skupina telefonních přístrojů připojena k pobočkové telefonní ústředně, pomocí které jsou obvykle připojeny i k veřejné telefonní síti. Toto řešení je často používané ve firmách a v call centrech.

**Mobilní telefonní síť**

Mobilní síť je telefonní síť, jejíž uživatel není vázán na místa, kde je k dispozici telefonní přípojka. Telefonní přístroj komunikuje se sítí pomocí rádiových vln.

Mobilní telefonní sítě lze rozdělit na:

**Celulární** (buňkové) radiové sítě – pokrytí rozsáhlejší oblasti je zabezpečeno větším množstvím základnových stanic

**Sítě satelitních telefonů** – pro komunikaci využívá systém umělých družic Země, z nichž každá pokrývá rozsáhlé území

**Sítě bezšňůrových telefonů** – telefony mohou komunikovat buď s pevnou základnovou stanicí nebo přímo mezi sebou

Protože naprostá většina mobilních telefonů používá buňkovou technologii, jako GSM, CDMA nebo starší NMT (v USA AMPS), jsou v mnoha zemích a jazycích slova mobilní telefon (anglicky mobile phone) a buňkový telefon (anglicky cellphone) prakticky synonymy

**Wi-Fi**

Wi-Fi je v informatice označení pro několik **standardů IEEE 802.11** popisujících bezdrátovou komunikaci v počítačových sítích (též Wireless LAN, WLAN)

Samotný název Wi-Fi vytvořilo Wireless Ethernet Compatibility Aliance

Tato technologie využívá tak zvaného „bez licenčního frekvenčního pásma“, proto je ideální pro budování levné, ale výkonné sítě bez nutnosti pokládky kabelů

**SOUSTAVY**

**Poziční soustava**

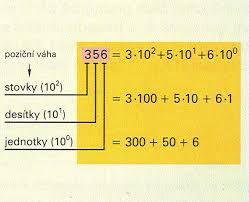
* Má řády, abychom věděli, na které pozici se číslo nachází

**Nepoziční soustava**

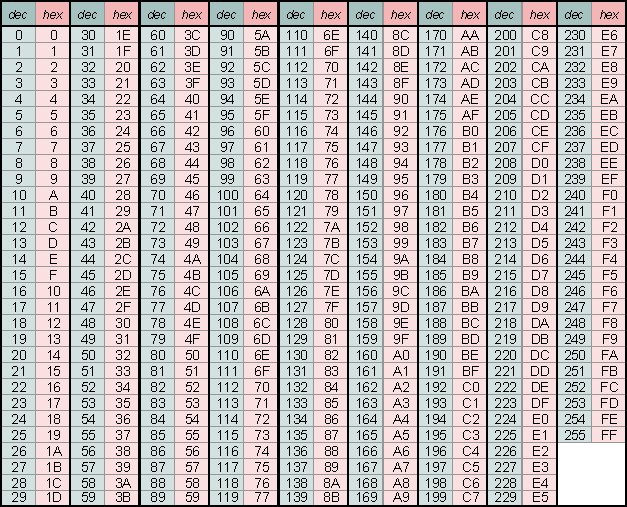
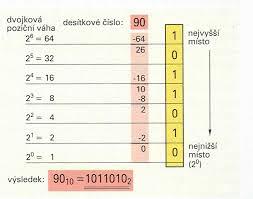
* Nezvyšují se jednotlivé řády (římská soustava)

**Desítková soustava**

* 10 číslic (0 - 9)

[](https://www.google.cz/imgres?imgurl=x-raw-image:///098a0b92256f144a210d9c7a37d02ffcd93139b86c204c279439a162ded30361&imgrefurl=http://mikroelektro.utb.cz/request.php?24&docid=pUk9a0IKlqdNjM&tbnid=ShRvJZILigk8MM:&vet=10ahUKEwjFmOz3nZznAhXSKFAKHS3ZD5oQMwj5AShTMFM..i&w=573&h=465&bih=633&biw=1366&q=%C5%A1estn%C3%A1ctkov%C3%A1%20soustava%20%C4%8D%C3%ADslo%20pozi%C4%8Dn%C3%AD&ved=0ahUKEwjFmOz3nZznAhXSKFAKHS3ZD5oQMwj5AShTMFM&iact=mrc&uact=8)

**Dvojková (binární) soustava**

 [](https://www.google.cz/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&ved=2ahUKEwjxopvdnZznAhUYAWMBHY9RBfQQjRx6BAgBEAQ&url=http://files.sedlackova.webnode.cz/200000080-c2617c35bc/2_%C4%8D%C3%ADseln%C3%A9%20soustavy.pdf&psig=AOvVaw2QrT5NyYNEckhTr6_BPvOD&ust=1579955143571255)

**Osmičková soustava**

* 0 – 7
* Z dvojkové: rozdělím po třech

**Šestnáctková soustava**

* 0 – F
* Z Dvojkové: rozdělím po čtyřech

## Síťová vrstva

**Účel**

* Tato vrstva se stará o směrování v síti a síťové adresování.
* Poskytuje spojení mezi systémy, které spolu přímo nesousedí.
* Obsahuje funkce, které umožňují překlenout rozdílné vlastnosti technologií v přenosových sítích.
* Nejznámější protokol, který pracuje na této vrstvě je **Internet Protocol**.
* **Nejdůležitějším úkolem síťové vrstvy je tedy tzv.** směrování (routing), které rozhoduje o směru odesílání jednotlivých paketů.
* Potřebuje alespoň základní informace o topologii celé sítě.
* Konkrétních způsobů směrování, či spíše postupů volby vhodného směru přitom existuje celá řada.
* **Od jednoduchých statických metod**, které nejsou schopny reagovat na dynamické změny v síti, až po **adaptivní metody**, které se dokážou přizpůsobit aktuálnímu stavu sítě, jejímu zatížení, případným výpadkům některých uzlů či spojů.

**Směrování**

* Základy směrování
  + Směrování paketů > zajišťuje router.
* Routing, česky řečeno směrování.
  + Jedná se o techniku, která slouží k propojení jednotlivých sítí (přesněji subnetů).
  + Původním zařízením, určeným pro routování byl router, ale v dnešní době se velmi využívají L3 switche, firewally nebo pouze servery/počítače.
  + **Router přeposílá komunikaci z jedné sítě do jiné.**

**Princip činnosti routeru**

* Pokud router přijme paket na rozhraní, koukne se na MAC adresu v rámci, zda je zpráva opravdu určena pro něj.
* Pokud MAC adresy souhlasí, otevře IP paket a podívá se na IP adresu příjemce.
  + Poté prohledá svojí routovací tabulku; pokud má onu IP adresu uloženou, poté paket pošle na aktivní port, u které má danou MAC adresu přiřazenou.
  + Pokud IP adresu v routovací tabulce nemá, a nemá nastavenou statickou cestu (nulová adresa) - “vše co neznáš, posílej tudy”; pak paket zahodí.
* Do routovací tabulky se vytváří několik typů záznamů cest (route), záleží na tom, jakým způsobem vznikly.
* Pakety jsou podle toho směrovány jedním ze základních způsobů routování

**Statické routování**

* ručně zadané cesty (záznamy v routovací tabulce), bezpečné a dobré, ale nereflektuje změny v topologii sítě

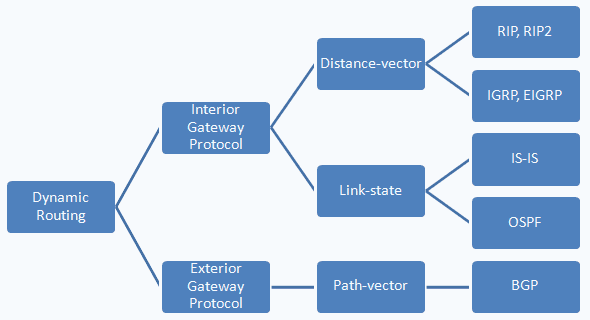
**Dynamické routování**

* síť se automaticky přizpůsobuje změnám v topologii a dopravě, automaticky se vypočítávají cesty pomocí routovacího protokolu

**Defaultní routování**

* dochází k němu, pokud neexistuje žádná jiná odpovídající cesta
* pokud není nastavena defaultní routa, paket je zahozen

**Směrovací protokoly**



* Dynamické routovací protokoly jsou dvou základních typů
  + **distance-vector routing protocol**
    - routery udržují routovací tabulku s informací o (vektoru) vzdálenosti do dané sítě
    - periodicky routovací tabulku zasílají sousedům, ti si upraví svoji tabulku a tu opět odešlou dál
    - pro výpočet nejlepší cesty se používá jedna (počet hopů u RIP) nebo více metrik (propustnost linky a zpoždění u IGRP)
  + **link-state routing protocol**
    - routery udržují komplexní databázi síťové topologie (vytvořenou pomocí LSA)
    - **vyměňují si link-state advertisements** (LSA)
    - LSA jsou vyvolány nějakou událostí v síti, do svého okolí také odesílá Hello pakety, kde zasílá informace o sobě
    - rychle reaguje na změny topologie, ale spotřebovává více pásma a zdrojů na routeru
    - metrika je komplexní
    - nejlepší cesta se počítá pomocí **Dijkstrova algoritmu** shortest path first (SPF)
* Dále dělíme dynamické protokoly podle toho, zda jsou určeny pro nasazení uvnitř lokální sítě (nebo fungují napříč sítěmi (spojují AS dohromady)
  + **interior gateway protocol**
    - IGP
    - routuje uvnitř Autonomous System (AS)
  + **exterior gateway protocol**
    - EGP
    - routuje mezi AS

**Směrovací tabulka, administrativní vzdálenost, metrika**

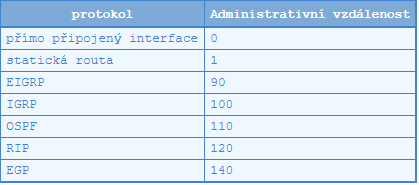
**Směrovací tabulka**

* Primární funkcí routeru je přeposílat pakety do jejich cílové sítě, kterou určuje cílová IP adresa v hlavičce paketu.
* K tomuto potřebuje prohledávat routovací informace v routovací tabulce.
* **Routovací tabulka je datový soubor uložený v RAM paměti, který je používán k uchovávání informací** ohledně přímo připojených i vzdáleně připojených sítích.
* Její obsah napovídá routeru, kterým rozhraním je možno nejoptimálněji dosáhnout cílové sítě.
* Next hop, neboli cesta do další sítě bývá vyjádřen názvem rozhranní (Fa0/0) nebo IP adresou sousedního rozhraní (192.168.0.2)
* Příklazem ***show ip route*** je zobrazena routovací tabulka.
  + Tato tabulka říká, na které rozhraní poslat paket podle jeho cílové adresy.
  + Jedná se tedy o tzv. směrování podle cíle.
  + Informace na konci jednotlivého zápisu reprezentuje rozhraní nebo IP adresu next-hopu (cesty k vedlejšímu routeru).

**Administrativní vzdálenost**

* číselný identifikátor spolehlivosti Routy
* čím nižší je vzdálenost, tím věrohodnější a spolehlivější je
* nachází se před lomítkem
* **vlastnost používaná na routerech k určení nejlepší cesty mezi více routovacími protokoly**
* definuje spolehlivost protokolu a priorizuje lepší nižším číslem
* Na Cisco routerech můžeme měnit defaultní hodnoty

[Dobrý web](https://www.samuraj-cz.com/clanek/tcpip-routing-smerovani/)



**Metrika**

* za lomítkem
* číselný identifikátor udávající hodnotu a dostupnost vzdálené sítě
* čím nižší je, tím víc ji budu preferovat
* **počítána pomocí algoritmů, které využívá daný směrovací protokol**

**Výchozí cesta (default route)**

* tuto trasu používají Routery tehdy, když není žádná jiná možnost – například, když nemohou nalézt požadovanou síť ve své tabulce.
* může se použít pouze na kraji sítě, kde je jeden port, který vede mimo síť
* pokud není definována jiná cesta k dosažení cíle, tak se použije defaultní routa, což je gateway

**Statické a dynamické směrování**

**Statické směrování**

* používá se například mezi ISP a firmou, není zde třeba, aby běhal složitý routovací protokol
* pro každou síť je vložen záznam do routovací tabulky
* žádná zátěž, pouze pro malé sítě

**Dynamické routování**

* síť se automaticky přizpůsobuje změnám v topologii a dopravě, automaticky se vypočítávají cesty pomocí routovacího protokolu

**Protokoly síťové vrstvy**

**IP (Internet Protokol)**

* Funguje na třetí vrstvě ISO/OSI modelu.
* Přijímá segmenty od transportní vrstvy, přidává k nim IP hlavičku a patičku, IP adresy, předává pakety linkové vrstvě.
* Není závislá na přenosovém médiu.
  + IP protokol není schopný zajistit spolehlivé doručení všech paketů (některé pakety se cestou sítí můžou ztratit).
* **Nespojovanost:** Před předáním paketů se neurčuje spojení s cílovým zařízením

**ARP**

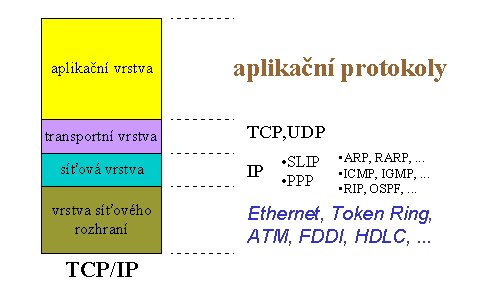
* Získání linkové adresy síťového rozhraní protistrany ve stejné podsíti pomocí známé IP adresy.
* Používá k tomu MAC adresu ve tvaru **ff:ff:ff:ff:ff:ff**
* Posílá se broadcast.

**ICMP**

* Používají operační systémy v síti pro odesílání služebních informací, například chybových zpráv pro oznámení, že požadovaná služba není dostupná nebo že potřebný počítač nebo router není dosažitelný.
* Generován na základě nějaké události
* Výjimkou je např. nástroj ping, který posílá ICMP zprávy „Echo Request“ (a očekává příjem zprávy „Echo Reply“), aby určil, zda je cílový počítač dosažitelný a jak dlouho paketům trvá, než se dostanou k cíli a zpět.

**IPsec**

* Bezpečnostní rozšíření IP protokolu založené na autentizaci a šifrování každého IP datagramu.
* V architektuře OSI se jedná o zabezpečení již na síťové vrstvě, a proto poskytuje transparentně bezpečnost jakémukoliv přenosu (kterékoliv síťové aplikaci).
* Bezpečnostní mechanismy vyšších vrstev (nad protokoly TCP/UDP, kde pracují TLS/SSL, SSH apod.) vyžadují podporu uvnitř aplikací.
* IPsec používá kryptografické bezpečnostní služby pro ochranu komunikace prostřednictvím skrze IP protokol.
* Poskytuje ověřování dat, původ dat, jejich integritu, důvěrnost (šifrování) a ochranu proti útoku pomocí přehrání (viz replay attack).



1. Transportní a aplikační vrstva

## TRANSPORTNÍ VRSTVA

**Účel transportní vrstvy**

* Hlavním úkolem transportní vrstvy **je poskytovat efektivní přenosové služby své bezprostředně vyšší (tj. relační) vrstvě**. Tyto služby přitom mohou mít spojovaný i nespojovaný charakter.
* Umožňuje adresovat přímo aplikace (například v protokolech TCP/IP pomocí čísel portů).
* Poskytuje transparentní, spolehlivý přenos dat s požadovanou kvalitou.
* Vyrovnává různé vlastnosti a kvalitu přenosových sítí.
* Provádí převod transportních adres na síťové, ale nestará se o směrování.
* Transportní vrstva se stará o doručení dat k příslušnému aplikačnímu procesu na hostitelském počítači.
* Pro transportní vrstvu je velmi podstatné, jakou kvalitu může předpokládat u služeb, které jí poskytuje síťová vrstva.
* Této kvalitě pak musí být přizpůsobeny přenosové protokoly transportní vrstvy, mají-li vyšší vrstvy odstiňovat od všech specifik a nedokonalostí komunikační podsítě resp. síťové vrstvy.
* Služby síťové vrstvy se v této souvislosti rozdělují na tři kategorie

**Kategorie / Vlastnosti**

* A / bez chyb při přenosu paketů a bez výpadků spojení
* B / bez chyb při přenosu paketů, s výpadky spojení
* C / s chybami při přenosu paketů, s výpadky spojení
  + **Do kategorie A** patří síťové služby, které ztrácí či přenáší s chybou jen zcela zanedbatelné procento paketů, a u kterých prakticky nedochází k výpadkům již jednou navázaných spojení.

Takovéto síťové služby lze považovat za (téměř) dokonalé, a transportní vrstva pak má při jejich použití nejméně práce.

Kvality kategorie A dosahují síťové služby některých lokálních sítí, zatímco v případě rozlehlých sítí je kategorie A vzácná.

* + **Do kategorie B** spadají síťové služby, které jsou z hlediska přenosu jednotlivých paketů stejně spolehlivé jako kategorie A, ale u kterých dochází častěji k výpadkům spojení.

Tedy k situacím, kdy z různých důvodů (přetížení, poruchy hardwaru, chyby v softwaru apod.) dochází k předčasnému ukončení (výpadku) dříve navázaného spojení.

Transportní vrstva se s tím dokáže účinně vyrovnat (opětovným navázáním nového síťového spojení) a existenci výpadků před vyššími vrstvami skrýt.

Do kategorie B spadá obvykle většina veřejných datových sítí na bázi doporučení X.25.

* + **Do kategorie C** pak patří všechny ostatní síťové služby.

Transportní vrstva je musí považovat za nespolehlivé, a potřebnou spolehlivost zajistit sama.

Protokoly transportní vrstvy, které musí využívat služby této kategorie, jsou pak samozřejmě nesložitější a nejkomplikovanější.

* Třídy transportních protokolů
* Referenční model ISO/OSI se vyrovnává s odlišnou kvalitou služeb na úrovni síťové vrstvy zavedení pěti různých tříd přenosových protokolů na úrovni transportní vrstvy (transportních protokolů)

**Třída transportních protokolů / Funkce**

* **TP0** / jednoduchá obálka síťových protokolů

Patří sem transportní protokoly, které využívají síťové služby kategorie A. Jsou jednoduchou "obálkou" nad příslušnými síťovými protokoly, a navíc zajišťují prakticky jen nezbytné navazování a rušení transportních spojení.

* **TP1** / zotavení z výpadků síťových spojení

Třída 1 (TP1) předpokládá použití síťových služeb kategorie B, a musí se tedy umět vyrovnat s případnými výpadky spojení na úrovni síťové vrstvy.

* **TP2** / multiplexování jednoho síťového spojení mezi více transportních spojení

Třída 2 (TP2) předpokládá opět použití síťových služeb kategorie A, tedy takových, které může považovat za dostatečně spolehlivé - stejně jako v případě třídy 0. Transportní protokoly třídy 2 jsou však navíc schopné zajistit potřebné multiplexování jednoho síťového spojení mezi více spojení transportních (viz výše).

* **TP3** / multiplexování a zotavení z výpadků

Třída 3 (TP3) je kombinací tříd 1 a 2. Předpokládá síťové služby kategorie B, a dokáže je multiplexovat mezi více spojení transportních.

* **TP4** / detekce a oprava chyb, zotavení z výpadků

Třída 4 (TP4) je určena pro použití nad síťovými službami kategorie C. Transportní protokoly této třídy musí "počítat s nejhorším", a jsou tedy ze všech transportních protokolů nejsložitější a nejkomplikovanější.

**Protokoly transportní vrstvy**

**TCP**

* Je nejpoužívanějším protokolem transportní vrstvy v sadě protokolů TCP/IP používaných v síti Internet.
* Použitím TCP mohou aplikace na počítačích propojených do sítě vytvořit mezi sebou spojení, přes které mohou obousměrně přenášet data. Protokol garantuje spolehlivé doručování a doručování ve správném pořadí.
* TCP také umožňuje rozlišovat a rozdělovat data pro více aplikací (například webový server a emailový server) běžících na stejném počítači.
* TCP využívá mnoho populárních aplikačních protokolů a aplikací na internetu, včetně WWW, e-mailu a SSH.
* **Navázání a ukončení spojení**

Protože TCP je spojovaná transportní služba, musí se před odesíláním dat navázat spojení mezi klientem a serverem.

K tomu slouží **trojcestný handshaking** (three-way handshake). V průběhu navazování spojení se obě strany dohodnou na číslu sekvence a potvrzovacím čísle.

Pro navázání spojení se odesílají datagramy s nastavenými příznaky **SYN a ACK.**

Jak z názvu vyplývá, navázání spojení probíhá ve třech krocích:

1. Klient odešle na server datagram s nastaveným příznakem SYN a náhodně vygenerovaným číslem sekvence (x), potvrzovací číslo=0.

2. Server odešle klientovi datagram s nastavenými příznaky SYN a ACK, potvrzovací číslo=x+1, číslo sekvence je náhodně vygenerované (y)

3. Klient odešle datagram s nastaveným příznakem ACK, číslo sekvence=x+1, číslo odpovědi=y+1.

* Obě strany si pamatují číslo sekvence své i protistrany. Používají se totiž i pro další komunikaci a určují pořadí paketů. Když úspěšně proběhne trojcestný handshaking, je spojení navázáno a zůstane tak až do ukončení spojení. To se může zneužít na SYN flood útok.
* Ukončení spojení probíhá podobně jako jeho navázání. Používá se k tomu příznaků **FIN a ACK:**

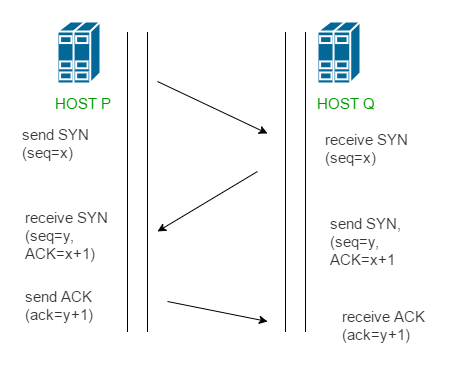
1. Klient odešle datagram s nastaveným příznakem FIN

2. Server odpoví datagramem s nastaveným příznakem ACK

3. Server odešle datagram s nastaveným příznakem FIN

4. Klient odpoví s nastaveným příznakem ACK

* Teprve po těchto čtyřech krocích je spojení ukončeno



**UDP**

* Nedává záruky na datagramy, které přenáší mezi počítači v síti.
* Někdy je označován jako nespolehlivý, ale jde o velmi zavádějící označení.
* Na rozdíl od protokolu TCP totiž nezaručuje, zda se přenášený datagram neztratí, zda se nezmění pořadí doručených datagramů, nebo zda některý datagram nebude doručen vícekrát.
* Protokol UDP je vhodný pro nasazení, které vyžaduje jednoduchost nebo pro aplikace pracující systémem otázka-odpověď (např. DNS, sdílení souborů v LAN).
* Jeho bezstavovost je užitečná pro servery, které obsluhují mnoho klientů nebo pro nasazení, kde se počítá se ztrátami datagramů a není vhodné, aby se ztrácel čas novým odesíláním (starých) nedoručených zpráv (např. VoIP, online hry).
* UDP poskytuje velmi jednoduché rozhraní mezi síťovou vrstvou pod a aplikační vrstvou nad.
* UDP neposkytuje žádné záruky doručení a odesilatelova UDP vrstva si u jednou už odeslaných zpráv neudržuje žádný stav.
* UDP pouze přidává kontrolní součty a schopnost roztřiďovat UDP pakety mezi více aplikací běžících na stejném počítači.

**Rozdíl mezi UDP a TCP**

* TCP je spojově orientovaný protokol což znamená, že k navázání "end-to-end" komunikace potřebuje, aby proběhl mezi klientem a serverem tzv. "handshaking". Poté, co bylo spojení navázáno, data mohou být posílána oběma směry.
* Charakteristické vlastnosti TCP protokolu jsou:

**spolehlivost** – TCP používá potvrzování o přijetí, opětovné posílání a překročení časového limitu. Pokud se jakákoliv data ztratí po cestě, server si je opětovně vyžádá. U TCP nejsou žádná ztracená data, jen pokud několikrát po sobě vyprší časový limit, tak je celé spojení ukončeno.

**zachování pořadí** – Pokud pakety dorazí ve špatném pořadí, TCP vrstva příjemce se postará o to, aby se některá data pozdržela a finálně je předala správně seřazená.

**vyšší režie** – TCP protokol potřebuje např. tři pakety pro otevření spojení, umožňuje to však zaručit spolehlivost celého spojení.

UDP je jednodušší protokol založený na odesílání nezávislých zpráv. Charakteristika protokolu:

bez záruky – Protokol neumožňuje ověřit, jestli data došla zamýšlenému příjemci. Datagram se může po cestě ztratit. UDP nemá žádné potvrzování, přeposílání ani časové limity. V případě potřeby musí uvedené problémy řešit vyšší vrstva.

nezachovává pořadí – Při odeslání dvou zpráv jednomu příjemci nelze předvídat, v jakém pořadí budou doručeny.

jednoduchost – Nižší režie než u TCP (není zde řazení, žádné sledování spojení atd.).

## APLIKAČNÍ VRSTVA

**Účel transportní vrstvy**

* Aplikační vrstva je 7. vrstva modelu vrstvové síťové architektury (OSI)
* Účelem aplikační vrstvy je poskytnout aplikačním procesům přístup ke komunikačnímu systému a tím umožnit jejich vzájemnou spolupráci.
* **Služby poskytované aplikační vrstvou:**

přenos zpráv

identifikace komunikujících parametrů (jmény, adresami, podpisem)

zjištění stupně okamžité připravenosti komunikujícího partnera

stanovení pověření pro komunikaci

dohoda o mechanismech ochrany zpráv

ověření přípustnosti komunikujících parametrů

určení přiměřenosti prostředků

určení způsobu přiřazení tarifů a přijatelné kvality poskytovaných služeb

synchronizace spolupracujících aplikací

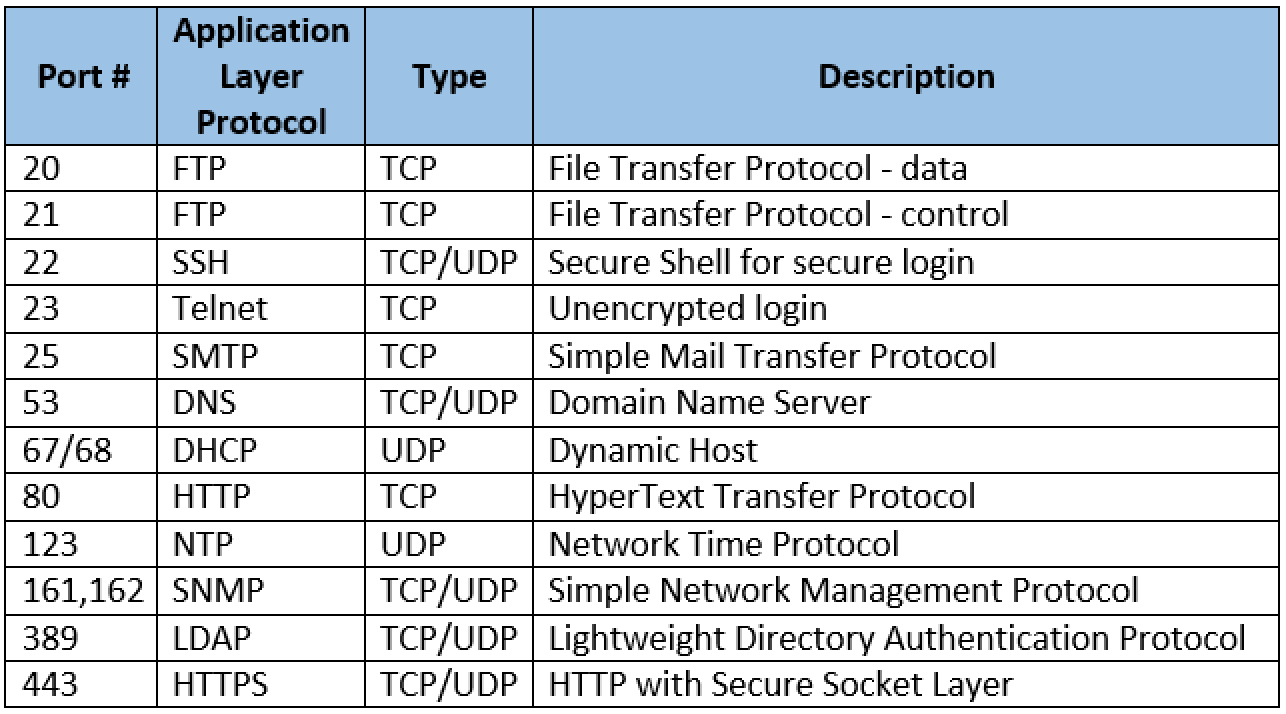
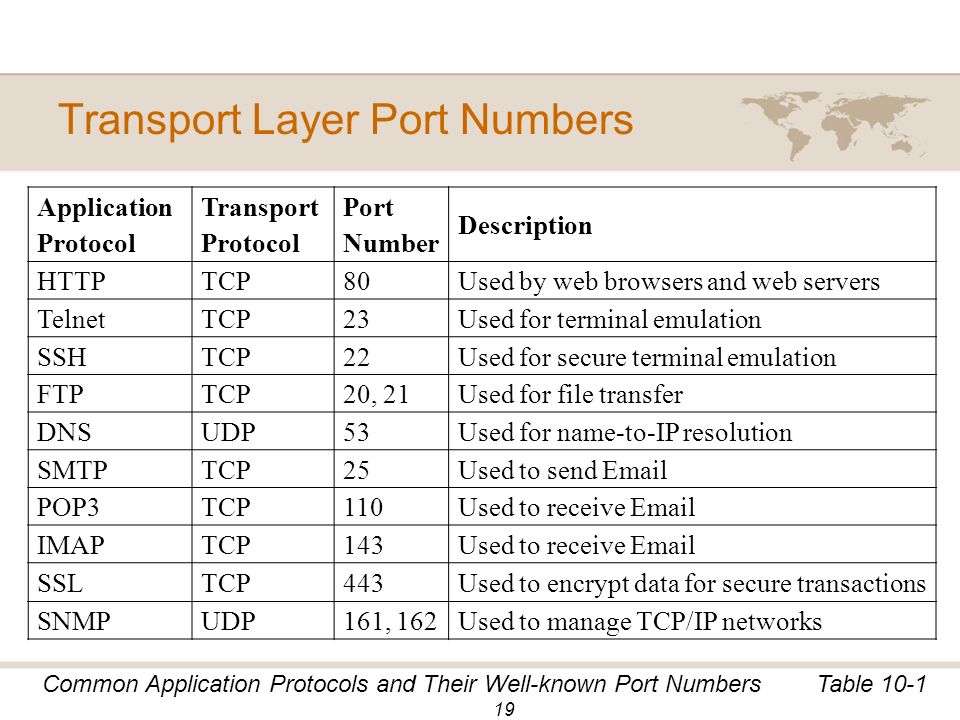
výběr způsobu dialogu včetně postupu jeho navázání a ukončení

dohoda o odpovědnosti za opravy chyb a o postupech řízení celistvosti dat

dohoda o omezeních týkajících se syntaxe zpráv (struktura, kódy, abecedy)

* K zabezpečení služeb aplikační vrstvy jsou potřebné funkce, které jsou zahrnuty v nižších vrstvách a poskytovány formou patřící jen do aplikační vrstvy.
* Na rozdíl od ostatních vrstev mohou funkce v aplikační vrstvě provádět nejen programy a technické prostředky, ale i lidé.

**Protokoly transportní vrstvy**

****

**DHCP** (UDP – 67/68)

* Používá se pro **automatickou konfiguraci počítačů připojených do počítačové sítě**.
* DHCP server přiděluje počítačům pomocí DHCP protokolu zejména IP adresu, masku sítě, implicitní bránu a adresu DNS serveru.
* Platnost přidělených údajů je omezená, proto je na počítači spuštěn DHCP klient, který jejich platnost prodlužuje.
* Klienti žádají server o IP adresu, ten u každého klienta eviduje půjčenou IP adresu a čas, do kdy ji klient smí používat (doba zapůjčení, anglicky lease time). Poté co vyprší, smí server adresu přidělovat jiným klientům.
* Klient komunikuje na UDP portu 68, server naslouchá na UDP portu 67. Po připojení do sítě klient vyšle broadcastem DHCPDISCOVER paket.
* Na ten odpoví DHCP server paketem DHCPOFFER s nabídkou IP adresy. Klient si z (teoreticky několika) nabídek vybere jednu IP adresu a o tu požádá paketem DHCPREQUEST.
* Server mu ji vzápětí potvrdí odpovědí DHCPACK. Jakmile klient obdrží DHCPACK, může už IP adresu a zbylá nastavení používat.
* Klient musí před uplynutím doby zapůjčení z DHCPACK obnovit svou IP adresu. Pokud lhůta uplyne aniž by dostal nové potvrzení, klient musí IP adresu přestat používat.

**FTP** (TCP – 20/21)

* Protokol pro přenos souborů mezi počítači pomocí sítě.
* Využívá protokol TCP a může být nezávislý na OS.
* Využívá porty 20 (pouze data) a 21 (k řízení).
* Připojení k FTP serveru je možné realizovat v aktivním nebo pasivním režimu. Pasivní režim je bezpečnější, ale ne vždy je technicky realizovatelný.
* Aktivní režim

Na portu TCP/20 jsou přenášena data (data connection). V aktivním režimu navazuje připojení pro přenos dat server, klient naslouchá. Problém zpravidla nastává v případě, kdy se klient připojuje z privátní sítě a jeho IP adresa je překládána (NAT).

* Pasivní režim

V pasivním režimu navazuje data connection klient, kterému při sestavování připojení poslal server svou IP adresu a TCP port, na kterém naslouchá.

* Jako FTP klienta a server můžeme použít například FilleZillu.

**SMTP**

* Protokol určený pro přenos zpráv elektronické pošty mezi přepravci el. pošty.
* Zajišťuje doručení pošty pomocí přímého spojení mezi odesílatelem a adresátem.
* Zpráva je doručena do tzv. poštovní schránky adresáta, ke které potom může uživatel kdykoli přistupovat (vybírat zprávy) pomocí protokolů POP3 nebo IMAP.

**POP3**

* Protokol pro stahování emailových zpráv ze vzdáleného serveru na klienta.
* Ze vzdáleného serveru se stáhnou všechny zprávy, třeba i ty, které uživatel číst nechce, nebo spam (pokud ho již nefiltruje poštovní server).
* Většina POP3 serverů sice umožňuje stáhnout i pouze hlavičky zpráv (a následně vybrat zprávy, které se stáhnou celé), ale podpora v klientech vesměs chybí.
* Tuto nevýhodu může odstranit protokol IMAP, který pracuje se zprávami přímo na serveru.

**IMAP** (TCP – 143)

* Protokol pro **vzdálený přístup k emailové schránce** prostřednictvím klienta.
* Nabízí pokročilé možnosti vzdálené správy a práci v on-line a off-line režimu oproti od POP3.
* Protokol IMAP umožňuje trvalé (tzv. on-line) připojení k e-mailové schránce. Díky tomu je možné s celou poštovní schránkou plně pracovat z libovolného místa.
* Všechny zprávy a složky jsou uloženy na poštovním serveru a na počítač se stahují jen nezbytné informace, takže při zobrazení složky se stáhnou jen záhlaví zpráv a jejich obsah až v případě, že zprávu chce uživatel přečíst.
* U jednotlivých zpráv se uchovává jejich stav (nepřečtená, odpovězená, důležitá), uživatel může zprávy přesouvat mezi složkami, složky vytvářet, mazat, prohledávat na straně serveru apod.
* Protokol umožňuje současné připojení více klientů zároveň.

1. Kryptografie, PKI a legislativa v IT

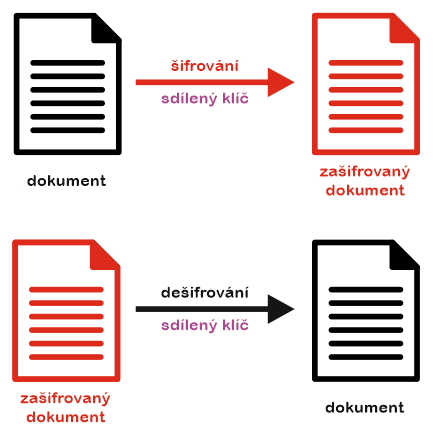
## KRYPTOGRAFIE

* Bez šifrování by byla síťová komunikace nebo jakákoliv data viditelná jako tzv, *plaintext*. - data může kdokoliv odchytit a přečíst je.
* Právě toto znemožní šifra, ta vytvoří text, který se dá zobrazit jen po rozšifrování.
* Šifrovací algoritmus většinou generuje pseudonáhodná čísla.

**Hashovací funkce**

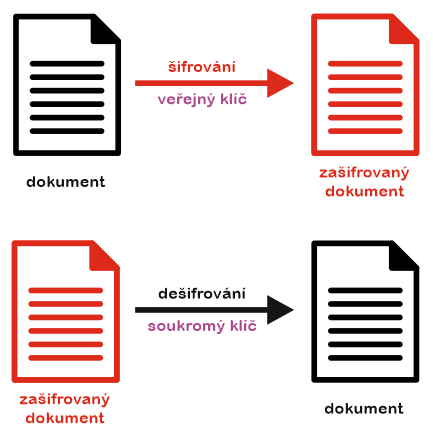
* **MD5** 
  + Vytváří 16 bytové hashovací hodnoty v hexadecimálních číslech po 32 číslicích
  + V dnešní době už není bezpečné
  + Využití dnes už jen jako kontrolní součet pro ověření datové integrity
* **SHA** 
  + Obsahuje 3 různé SHA algoritmy (SHA-0; SHA-1; SHA-2)
    - SHA-0 je velmi vzácně využíván
    - SHA-1 je nejvyužívanější (produkuje 20 bytové hashovací hodnoty)
* **SHA2** 
  + Set 6 hashovacích algoritmů (pokládán za nejsilnější)
  + Je doporučen **SHA-256 nebo vyšší** pro nejlepší bezpečnost
  + SHA-256 vytváří 32 bytové hashovací hodnoty
* **Prolomení**
  + *Brute force attack*
  + Postupné zkoušení všech známých použitých znaků, jejichž otisky se porovnávají s originálem
  + Použití slovníkových hesel
  + Vygooglovat hash
* **Solení hesel**
* K heslu se přidá ještě nějaký řetězec (sůl) a až potom se udělá HASH
* Je to například *username*

## Symetrická kryptografie



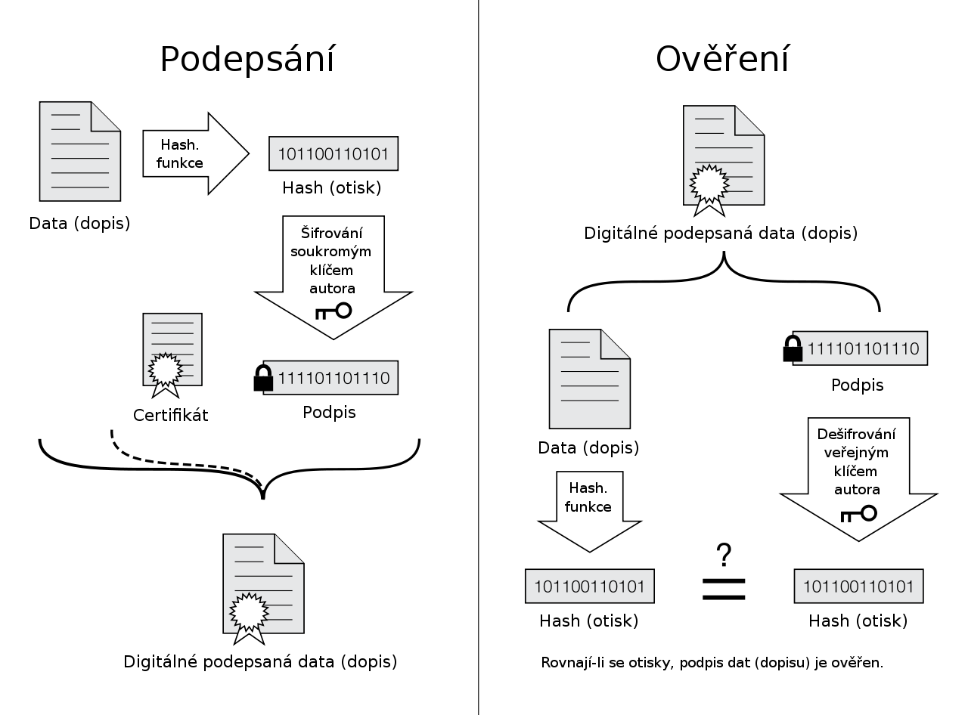
* Toto šifrování funguje pomocí jednoho klíče. Tento klíč musí být tajný **(Private Key).** 
  + -> Obě strany ho musí získat dopředu (tzv. Pre-sharedKey/Secret).
* Předání buď proběhne fyzicky, nebo ho také můžete získat díky: Diffie–Hellman Key Exchange či Asymetrické kryptografii s veřejných klíčem
* Pokud nezískáme tento bezpečný kanál, tak nemůžeme komunikovat.
* V případě symetrického šifrování se tentýž klíč používá jak pro šifrování, tak i pro dešifrování
* Náročnost algoritmů není vysoká, takže výpočet není tak složitý -> často se používá kód pro autentizaci zprávy (MAC).
* Tento typ kryptografie je docela **náchylný na útoky.**
* Je o hodně **rychlejší než asymetrické šifrování**. (AES 256 x RSA 4,096)
* *Byte stream* (Proudové) – upravují data po jednotlivých číslech (po bytech)
* *Block* – zpracovávají data po stanovených celcích (64 bit nebo 128 bit)

## Asymetrická kryptografie



* Zde se pro šifrování a dešifrování využívají dva klíče – Veřejný a soukromý klíč **(Public, Private Key).**
* Veřejný klíč a soukromý klíč jsou vzájemně spojeny a tvoří tzv. **klíčový pár**.
* Nejpoužívanější technika asymetrického šifrování se nazývá **RSA**.
* Veřejný klíč šifruje a soukromý klíč rozšifruje data.
  + Tyto klíče musí být párové, pouze jeden soukromý klíč je schopný získat původní data.
  + Složitost závisí na náročnosti veřejného klíče.  
    Veřejný klíč klidně může být volně k přístupu, aniž by to ovlivňovalo bezpečnost.
* Často se používá pro získání společného tajemství (Pre-shared secret), aby přenos byl rychlejší. Asymetrické šifrování je **mnohem pomalejší.** (AES 256 x RSA 4,096)
* Tento princip se používá při podepisování zpráv elektronickým podpisem.

## Elektronický podpis



* Je to označení dat, které v počítači nahrazují vlastnoruční podpis
* Je připojen k datové zprávě nebo je s ní logicky spojen -> umožňuje ověření totožnosti podepsané osoby ve vztahu k datové zprávě.
* **Ověření identity odesílatele.**
* Je vytvořen pro konkrétní data a je možné počítačem ověřit, zda je platný a jestli jsou data ve stejné podobě, ve které byla odeslána. Součástí toho je identifikace, kdo podpis vytvořil.

**Autenticita**

* Lze ověřit identitu subjektu, kterému patří elektronický podpis

**Integrita**

* Lze prokázat, že nedošlo k žádné změně v podepsaném dokumentu, tj. že dokument není úmyslně či neúmyslně poškozen

**Nepopiratelnost**

* Autor nemůže tvrdit, že elektronický podpis příslušný k dokumentu nevytvořil. Důvodem je fakt, že po vytvoření el. podpisu je potřeba privátní klíč.

**Časové ukotvení**

* Elektronický podpis může obsahovat časové razítko, které prokazuje datum a čas podepsání dokumentu.
* Časové razítko vydává důvěryhodná třetí strana, a protože je součástí elektronického podpisu, lze ji ověřit stejným postupem, jako elektronický podepsaný dokument.

**Princip**

* 1. Spočte se kontrolní součet (hash) z dokumentu.
* 2. Výsledný kontrolní součet se šifruje soukromým klíčem uživatele, který podpis vytváří.  
  Soukromým klíčem šifrovaným hash ze zprávy se nazývá **elektronický podpis zprávy.**

**Verifikace**

* 1. Příjemce samostatně spočte kontrolní součet z přijaté zprávy.
* 2. Příjemce dešifruje přijatý elektronický podpis veřejným klíčem odesílatele.
* 3. Příjemce porovná výsledek získaný z bodu 1 s výsledkem získaného z bodu 2. Pokud jsou stejné, pak elektronický podpis mohl vytvořit pouze ten kdo vlastní soukromý klíč odesílatele --> **odesílatel**.
* Je nutné si svůj soukromý klíč střežit a chránit.
* Na rozdíl od šifrování elektronický podpis použije klíč odesílatele, ale odesílatel pro podpis použije svůj soukromý klíč.

## PKI

**Certifikát**

* Certifikát obsahuje mj.: informace o tom, kdo jej vydal, sériové číslo certifikátu, identifikační údaje uživatele, platnost certifikátu a pochopitelně veřejný klíč uživatele.
* **Certifikát je digitálně podepsán** za využití soukromého klíče certifikační autority.

**Certifikační autorita**

* Subjekt, který vydává certifikáty, funguje na principu důvěry
* Tyto certifikáty obsahují identifikační údaje svého majitele
* Při žádosti musí přesvědčit to, že je to skutečně on (fyz. osoba --> občanský průkaz  
  práv. osoba --> ověření výpisu z obchodního rejstříku)

## LEGISLATIVA V IT

**Možnosti útoků**

* **Brute-Force Attack**
  + Pokouší se určit heslo tím, že zkouší každou možnou kombinaci.
  + Počet pokusů je omezený.
  + Nemusí nutně vyzkoušet svoje pokusy v pořadí.
  + Pokročilé programy dokážou vyvolat určité předpoklady.
    - Př. John the Ripper, Aircrack-ng a Rainbow crack.
* **Rainbow Tables**
  + Je obecně offlinový útok
  + Pracuje zpětně z šifrovaného/hashovanéhotextu
  + Útočník prochází celým algoritmem, aby dostal každý možný výstup s každým možným vstupem
  + Jednoduché vyhledávání je pak možné vzhledem k šifrované/hashované verzi hesla
* **Dictionary**
  + Většinou se jedná o odhadovaný útok s předkompilovaným seznamem možností
  + Než každou možnost, zkouší jen kompletní možnosti, které pravděpodobně budou fungovat
  + Mají tendenci vylučovat náhodné znaky hesel
    - Př. Cracx, Vigenere a PyForce

1. Bezpečnost IT

**Fyzická ochrana**

* Fyzicky počítač můžeme chránit například skleněnou skříní na zámek. Další možný způsob je například zamykat dveře do místnosti, kde PC je.
* Zamykat počítačovou skříň příslušným zámkem
* Používat hesla. Ve firmách používat Active Directory.

**Aktualizace**

* Aktualizace softwaru je v informatice postup, při kterém je do počítače instalována novější verze jeho programového vybavení.
* Aktualizace se provádí buď z důvodů implementace vyšší bezpečnosti, oprav chyb, nebo z důvodu přechodu na novější verzi software, která obvykle poskytuje novou funkcionalitu a vlastnosti.
* Používá se i slovo patch (ang. záplata).

**Uživatelské účty a hesla**

* Pokud Počítač používá více uživatelů, je lepší, když má každý svůj účet a vytvořené vlastní heslo. Tak nemají jiní uživatelé přístup do osobních dat.
* Každé heslo by mělo obsahovat minimálně 8 znaků, velká a malá písmena, číslo a speciální znak.
* **Autentizace** je proces ověření proklamované identity subjektu. Po dokončení autentizace obvykle následuje **autorizace**, což je souhlas, schválení, umožnění přístupu či provedení konkrétní operace daným subjektem. Autenticita je pak vlastnost subjektu, jejíž přítomnost se procesem autentizace ověřuje.

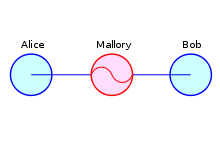
**Biometrie**

* Biometrie je automatická metoda autentizace založená na rozpoznávání jedinečných biologických charakteristik subjektu - živé osoby.
* Metoda vychází z přesvědčení, že některé biologické charakteristiky (morfologické, fyziologické) jsou pro každého živého člověka jedinečné a neměnitelné.
* Patří sem např.:
  + Autentizace otisku prstů
  + Autentizace oční duhovky
  + Podle oční sítnice
  + Podle obličeje
  + Dynamika stisku kláves
  + Charakteristika hlasu
  + Charakteristika písma

**Antivirová ochrana**

* Antivirový program (zkráceně antivir) je počítačový software, který slouží k identifikaci, odstraňování a eliminaci počítačových virů a jiného škodlivého software (**malware**). K zajištění této úlohy se používají dvě odlišné techniky:
  + Prohlížení souborů na lokálním disku, které má za cíl nalézt sekvenci odpovídající definici některého počítačového viru v databázi
  + Detekcí podezřelé aktivity nějakého počítačového programu, který může značit infekci. Tato technika zahrnuje analýzu zachytávaných dat, sledování aktivit na jednotlivých portech či jiné techniky.
* Úspěšnost závisí na schopnostech antivirového programu a aktuálnosti databáze počítačových virů. Aktuální virové databáze se dnes nejčastěji stahují z internetu.

**Útoky:**

**[](https://commons.wikimedia.org/wiki/File:Man_in_the_middle_attack.svg)Man-in-the-middle**

* Je název útoku na kryptografii. Jeho podstatou je snaha útočníka odposlouchávat komunikaci mezi účastníky tak, že se stane aktivním prostředníkem.

[*Alice a Bob*](https://cs.wikipedia.org/wiki/Alice_a_Bob) *chtějí spolu komunikovat, ale Mallory se je snaží odposlouchávat a případně i měnit zprávy, které si zasílají. Zároveň vůči Alici předstírá, že je Bob a vůči Bobovi předstírá, že je Alice*

* Podle obrázku vpravo není v současných počítačových sítích nutné, aby útočník (Mallory) byl na fyzické cestě mezi účastníky (Alice a Bob), protože síťový provoz lze přesměrovat.

**IP spoofing**

* Označuje vytvoření IP datagramu s falešnou zdrojovou IP adresou, který je následně odeslán počítačovou sítí k cílovému počítači, před kterým má být zatajena totožnost odesílatele.
* IP spoofing slouží k vedení útoků (např. DoS) a podobně. Na většině systémů je pro vytvoření datagramu s podvrženou IP adresou nutný raw přístup k internet soketu, což je umožněno pouze správci počítače, nikoliv běžnému uživateli.
* Základním protokolem Internetu je IP protokol, který pracuje s IP datagramy. V hlavičce každého IP datagramu je uvedena zdrojová a cílová IP adresa. Zdrojová adresa identifikuje odesílatele paketu. Falšováním záhlaví může útočník odeslat IP datagram tak, aby to vypadalo, že byl odeslán z jiného počítače. Počítač přijímající IP datagramy s falešnou zdrojovou adresou pošle odpověď na podvrženou adresu.
* Tato technika se používá tam, kde útočník nepotřebuje na odeslaný IP datagram reagovat nebo když může odpověď odchytit na zpáteční cestě IP datagramu.

**Brute force attack**

* Útok hrubou silou je většinou pokus o rozluštění šifry bez znalosti jejího klíče k dešifrování. V praxi se jedná o systematické testování všech možných kombinací nebo omezené podmnožiny všech dostupných kombinací.

**DoS/DDoS**

* Denial of service (DoS) (česky odepření služby) je typ útoku na internetové služby nebo stránky, jehož cílem je cílovou službu znefunkčnit a znepřístupnit ostatním uživatelům; může k tomu dojít přehlcením požadavky či využitím nějaké chyby, která sice útočníkovi neumožní službu ovládnout, ale umožní ji rozbít.
* Podtypem útoku DoS je tzv. distributed denial of service (DDoS), při kterém je pro přehlcení cílové služby požadavky využito velké množství rozptýlených počítačů.

**Password attack**

* Prolomení hesla je proces spočívající v získávání hesla resp. hesel z dat, které byly buďto již uloženy na úložné zařízení nebo jsou přenášena pomocí nějakého počítačového systému.
* Jeden z nejběžnějších přístupů je opakované zkoušení resp. odhadnutí hesla.

**Druhy škodlivého softwaru:**

**Trojský kůň**

* Škodlivý kód, který je ukryt v počítačovém programu a který se může na první pohled tvářit užitečně.
* Jde třeba o drobnou hru, spořič obrazovky, anebo právě program na odstranění malwaru. Často využívá legitimitu důvěryhodného zdroje – emailová zpráva s přílohou (v níž je trojský kůň) vytvářející domnění, že pochází např. od společnosti vyvíjející antivirové programy.

**Červ**

* Jedná se o škodlivý kód, jehož cíl je stejný jako u počítačových virů – poškození uživatele, resp. jeho dat.
* Od počítačového viru se liší zejména formou, jakou se šíří. Počítačový červ se dokáže replikovat sám a do dalších počítačových systémů se zpravidla šíří prostřednictvím počítačových služeb. Takto vytvořené kopie je schopen „na dálku“ aktivovat a spustit.
* Ke svému šíření rovněž využívají programových chyb systémů a dalších programů, které mají k systému přístup nebo mohou ovlivnit běh systému.

**Virus**

* Vir se nedokáže sám šířit 🡪 potřebuje „pomoc“ hostitele, tedy nakažené osoby. Viry se proto nejčastěji přenáší na přenosných zařízeních, jako jsou externí Harddisky, CD, DVD, Blue-ray atp.
* Na druhou stranu, i když se vir neumí sám přenášet, dokáže se polymorfně kopírovat. Tedy další vir bývá jiný než ten původní. V tomto spočívá také nebezpečnost virů.
* Protože to jsou programy schopné se měnit a mutovat a přizpůsobovat se, proto dokáží často zničit celou síť počítačů.
* Vůbec první vir vznikl ještě dříve než první červ.

**Spyware**

* Program, který využívá internetové stránky k odesílání dat z počítače (či mobilního telefonu nebo jiného zařízení) bez vědomí jeho uživatele.
* Někteří autoři spyware se hájí, že jejich program odesílá pouze data typu přehled navštívených stránek či nainstalovaných programů za účelem zjištění potřeb nebo zájmů uživatele a tyto informace využít pro cílenou reklamu.
* Existují ale i spyware odesílající hesla a čísla kreditních karet nebo spyware fungující jako zadní vrátka. Protože lze jen těžko poznat, do které skupiny program patří, a vzhledem k postoji k reklamě řada uživatelů nesouhlasí s existencí a legálností jakéhokoliv spyware.

**Adware**

* Je označení pro produkty znepříjemňující práci nějakou reklamní aplikací. Ty mohou mít různou úroveň agresivity - od běžných bannerů až po neustále vyskakující pop-up okna nebo ikony v oznamovací oblasti.
* Další nepříjemnou věcí je např. změna domovské stránky v Windows Internet Exploreru, aniž by o to uživatel měl zájem.
* Většinou ale nejsou přímo nebezpečné jako spyware a jsou spojeny s nějakým programem, který je freeware. To se dělá z důvodu toho, že díky těmto reklamám mohou vývojáři financovat dál svůj program. Nebo když se jedná o placený produkt, může se díky těmto reklamám prodávat program se slevou.
* Nějaký adware je taky shareware, ale není to totéž. Rozdíl mezi adware a shareware je ten, že u adware je reklama podporovaná. Některé produkty nabízejí uživateli možnost odstranění reklam po zaplacení.

**Pojmy:**

**Kerberos**

* Je síťový autentizační protokol umožňující komukoli komunikujícímu v nezabezpečené síti prokázat bezpečně svoji identitu někomu dalšímu. Kerberos zabraňuje odposlechnutí nebo zopakování takovéto komunikace a zaručuje integritu dat.
* Byl vytvořen primárně pro model klient-server a poskytuje vzájemnou autentizaci – klient i server si ověří identitu své protistrany.
* Kerberos je postavený na symetrické kryptografii, a proto potřebuje důvěryhodnou třetí stranu.
* Volitelně může využívat asymetrického šifrování v určitých částech autentizačního procesu.
* Standardně používá **port 88**

**Firewall**

* Brána firewall pomáhá zabránit hackerům a škodlivému softwaru (například červům) v získání přístupu k počítači prostřednictvím sítě nebo internetu. Brána firewall může rovněž zabránit tomu, aby počítač odesílal škodlivý software do jiných počítačů.
* Firewall kontroluje veškerou odchozí i příchozí komunikaci. Firewall může správce počítače nebo sítě kdykoliv vypnout, ale silně se to nedoporučuje.
* Pokud chceme používat webový server, sdílet tiskárny a podobně, nabízí Firewall výjimky, kde můžeme nastavit propustnost některých služeb do daného počítače.
* Výjimky můžeme přidávat dvěma způsoby. Buď můžeme do Firewallu přidat rovnou program, který má Firewall povolit, nebo zadat port přes který daný program prochází.

**Záloha**

* Záloha nebo záložní kopie je kopie dat uložená na jiném datovém nosiči (nebo i místě). Záložní data jsou využívána v případě ztráty, poškození nebo jiné potřeby práce s daty uloženými v minulosti. Zálohování probíhá nepravidelně (např. v domácnostech) nebo pravidelně podle rozvrhu (např. ve firmách).

**Šifrování**

* Šifrování dat je proces, kterým se nezabezpečená elektronická data převádí za pomoci kryptografie na data šifrovaná, čitelná pouze pro majitele dešifrovacího klíče. Šifrování dat slouží k jejich ochraně proti nežádoucímu zjištění cizí osobou a uplatňuje se při ukládání dat i při jejich přenosu včetně telekomunikace.

**Perimetr**

* Je zabezpečená oblast podnikové sítě. Může zahrnovat další síťové a bezpečnostní složky (hraniční směrovače, firewally, IDS systémy, zařízení VPN, SW, DMZ, sledované podsítě. Perimetr je určen strategickým rozhodnutím na základě identifikace síťových zdrojů a jejich hrozeb, na základě využívání zdrojů a odpovědnosti vůči nim.

**Hrozba**

* Je možné nebezpečí, které by mohlo zneužít chybu zabezpečení k porušení bezpečnosti, a způsobit poškození

**Riziko**

* Možnost, že daná hrozba využije zranitelnost, a tím uškodí.

**Zranitelnost**

* Programátorská chyba, která v software nebo v hardware způsobuje bezpečnostní problém. Je-li v programu přítomna chyba způsobující zranitelnost, označuje se její uskutečnitelné využití jako exploit.

**Zadní vrátka (backdoor)**

* metoda, která umožňuje obejít běžnou autentizaci, která za běžných okolností brání uživateli v neoprávněném využívání počítačového systému.
* Pokud jsou zadní vrátka v softwaru či hardwaru zabudována, mohou být využívána k seriózním účelům, avšak často jsou zneužívána či vládními organizacemi, takže jsou klasifikována jako bezpečnostní riziko, resp. zranitelnost.

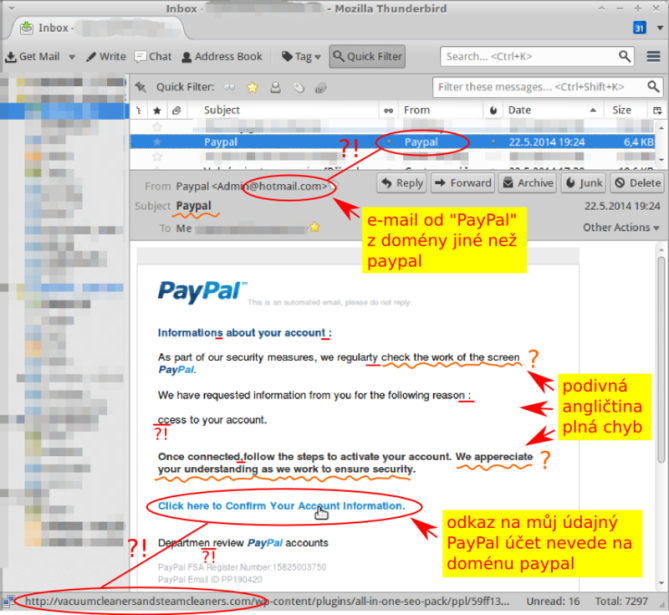
**Keylogger**

* Software, který snímá stisky jednotlivých kláves. Antivirem bývá považován za virus. V případě software se jedná o určitou formu spyware, ale existují i hardwarové keyloggery.
* Keylogger neohrožuje přímo počítač, ale slouží ke zjišťování hesel jiných lidí. Některé z nich bývají v operačním systému Microsoft Windows proti svému zničení chráněny pomocí Archivace a Skrytí souborů, takže je není možné pomocí Průzkumníku najít (je nutné použít vyhledávání).
* Existují i takové, kterých si antivirus nevšimne a které nejdou odinstalovat, takže je nutné je jednoduše smazat.

**Rootkit**

* Sada počítačových programů, pomocí kterých lze maskovat přítomnost zákeřného softwaru v počítači, například přítomnost virů, trojských koní, spywaru a podobně.
* Rootkit maskuje přítomnost zákeřných programů skrýváním adresářů, v nichž jsou instalovány, volání API, položek registru Windows, procesů, síťových spojení a systémových služeb tak, aby přítomnost zákeřného softwaru nebyla běžně dostupnými systémovými prostředky odhalitelná.
* Tyto programy jsou nástroje, které umožňují skrývat běžící procesy, soubory a systémové údaje, takže pomáhají útočníkovi zůstat skrytý (upravují operační systém tak, aby nebyly běžnými prostředky uživatele zjistitelné).

**Phishing**

*  Podvodná technika používaná na Internetu k získávání citlivých údajů (hesla, čísla kreditních karet apod.) v elektronické komunikaci.
* K nalákání důvěřivé veřejnosti komunikace předstírá, že pochází z populárních sociálních sítí, aukčních webů, on-line platebních portálů, úřadů státní správy nebo od IT administrátorů.
* Principem phishingu je typicky rozesílání e-mailových zpráv nebo instant messaging, které často vyzývají adresáta k zadání osobních údajů na falešnou stránku, jejíž podoba je takřka identická s tou oficiální.
* Stránka může například napodobovat přihlašovací okno internetového bankovnictví. Uživatel do něj zadá své přihlašovací jméno a heslo.
* Tím tyto údaje prozradí útočníkům, kteří jsou poté schopni mu z účtu vykrást peníze.

**WIPS**

* Wireless intrusion prevention system síťové zařízení, které kontroluje rádiové spektrum a hledá v něm neautorizované přístupové body (což je Intrusion Detection) a může podnikat automatická protiopatření (Intrusion Prevention).

**WIDS**

* Sleduje rádiové spektrum a hledá v něm neautorizované rogue AP, případně projevy použití nástrojů k útoku na bezdrátové sítě. Systém monitoruje rádiové spektrum používané bezdrátovými sítěmi a okamžitě informuje administrátora, pokud byl objeven rogue AP uvnitř sítě. Obvykle se to zjistí porovnáním MAC adres dotčených bezdrátových zařízení.
* **Cracker** (též black hat) je označení pro člověka, který zneužívá své vědomosti o počítačové bezpečnosti ke svému prospěchu při průnicích do software.
* Cracker musí mít dobré znalosti o principech fungování počítačů (informační technologie), programování, počítačové bezpečnosti, kryptografii a podobně. Nevhodným návrhem programů a existencí programátorských chyb vznikají v software zranitelnosti, které lze využít naprogramováním exploitu.
* Opakem crackera je **Hacker** (white hat), který využívá své znalosti ve prospěch uživatelů počítačových systémů (tj. odstraňuje programátorské chyby, diagnostikuje vadný hardware, programuje obtížné algoritmy). V médiích je často nesprávně používán termín hacker pro crackery.

**Zabezpečení bezdrátové sítě**

* Zabezpečení bezdrátové sítě (wireless security) slouží k tomu, aby se do sítě k počítačům nedostali neoprávnění uživatelé, kteří by mohli ostatním způsobit nějakou škodu.
* Mezi nejběžnější typy zabezpečení patří **Wired Equivalent Privacy** (WEP) a Wi-Fi Protected Access (WPA). WEP je jedna z nejméně bezpečných forem zabezpečení.
* WEP je starý IEEE 802.11 standard z roku 1999, který byl v roce 2003 nahrazen WPA.
* V dnešní době je **standardem WPA2**.

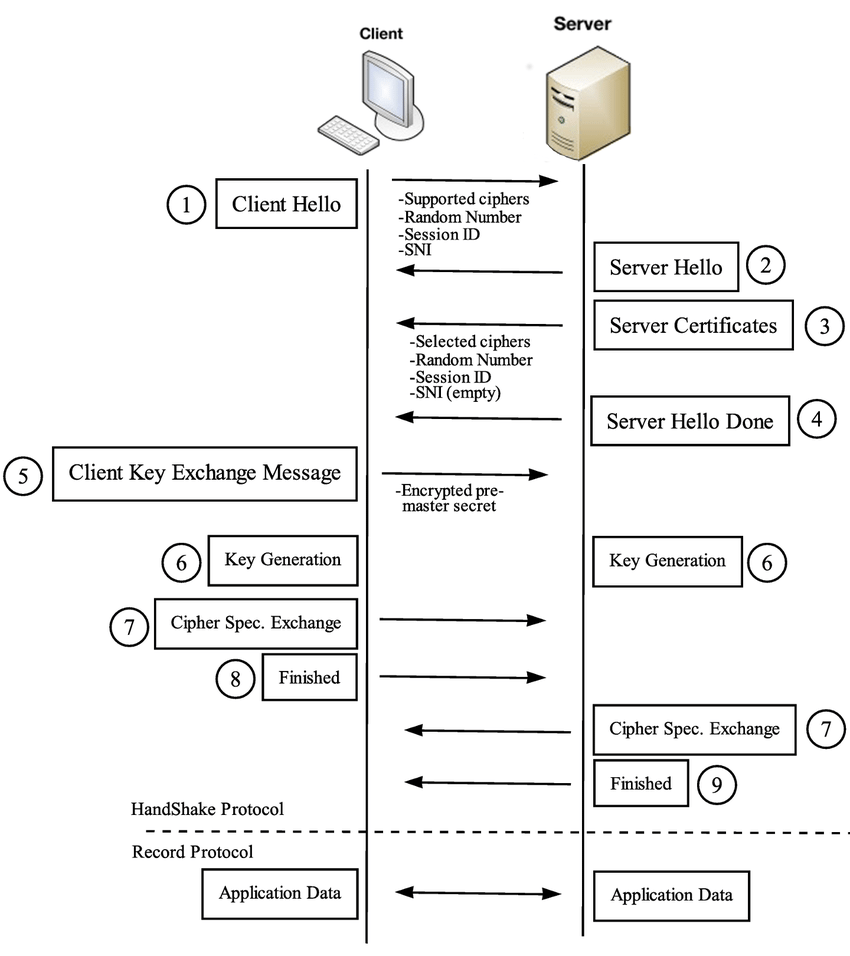
**Systém řízení bezpečnosti informací**

* (Information Security Management System - **ISMS**)
* Je dokumentovaný systém, ve kterém jsou chráněna definovaná informační aktiva, jsou řízena rizika bezpečnosti informací a zavedená opatření jsou kontrolována.
* Při zavádění systému řízení bezpečnosti informací v organizaci se postupuje podle normy **ISO/IEC 27001**, která poskytuje doporučení, jak ze souboru doporučených nejlepších postupů, které uvádí norma ISO/IEC 27002, případná certifikace se pak provádí podle normy ISO/IEC 27001.

**ISMS** je efektivní dokumentovaný systém řízení a správy informačních aktiv s cílem eliminovat jejich možnou ztrátu nebo poškození tím, že:

* + jsou určena aktiva, která se mají chránit
  + jsou zvolena a řízena možná rizika bezpečnosti informací
  + jsou zavedena opatření s požadovanou úrovní záruk a ta jsou kontrolována.

**TLS – praktická část**

* Klient pošle zprávu ClientHello oznamující nejvyšší verzi TLS, kterou podporuje, náhodné číslo a seznam doporučených šifrovacích sad a kompresních metod.
* Server odpoví zprávou ServerHello obsahující zvolenou verzi protokolu, náhodné číslo, šifrovací a kompresní metodu vybranou z klientem nabídnutého seznamu.
* Server pošle svůj certifikát (Certificate), pokud to zvolená šifra umožňuje. Současné certifikáty jsou založeny na X.509, ale existuje návrh na používání certifikátů vycházejících z OpenPGP.
* Server může pomocí CertificateRequest vyžadovat certifikát od klienta, aby bylo spojení autentizováno vzájemně.
* Server pošle zprávu ServerHelloDone, která signalizuje, že ukončil iniciační dohodu na používaných mechanismech.
* Klient odpoví zprávou ClientKeyExchange, jež může obsahovat PreMasterSecret, veřejný klíč nebo nic (v závislosti na zvolené šifře).
* Klient a server následně z náhodných čísel a PreMasterSecret pomocí pečlivě navržené pseudonáhodné funkce vypočítají „master secret“. Veškeré ostatní klíče jsou odvozeny z něj (a z generovaných náhodných hodnot).
* Klient nyní odešle zprávu ChangeCipherSpec, jíž v podstatě sděluje „veškerá další data ode mne budou šifrována“. Za pozornost stojí, že ChangeCipherSpec je sám o sobě protokolem záznamové vrstvy s typem 20, nikoli 22.
* Na závěr klient pošle šifrovanou zprávu Finished obsahující hash a MAC předchozích iniciačních zpráv.
* Server se pokusí dešifrovat klientovu zprávu Finished a ověřit její hash a MAC. Pokud dešifrování či ověření selže, inicializace je považována za neúspěšnou a spojení by mělo být ukončeno.
* Konečně server pošle zprávy ChangeCipherSpec a svou zašifrovanou Finished a klient provede analogické dešifrování a ověření.
* [](https://www.google.cz/url?sa=i&url=https://www.researchgate.net/figure/TLS-handshake-protocol_fig1_298065605&psig=AOvVaw22SABFmwMXs8gesd7IlBHf&ust=1582034287886000&source=images&cd=vfe&ved=0CAIQjRxqFwoTCIDd6ZLf2OcCFQAAAAAdAAAAABAD)V tomto okamžiku je inicializace dokončena a je povolen aplikační protokol, jehož typem obsahu je 23. Aplikační zprávy vyměňované mezi klientem a serverem budou zašifrovány.

1. Funkce a typy OS, OS Windows

**Charakteristika OS:**

* základní SW PC
* umožňuje komunikaci uživatele s počítačem
* zajišťuje spolupráci SW a HW PC
* spuštění OS zajišťuje **BIOS** (firmware OS)

**Základem OS je jádro (kernel)**

* + **Monolitické jádro** – jádro je jeden funkční celek (GNU/Linux)
  + **Mikrojádro** – malé jádro, oddělitelné části pracují jako samostatné procesy v user space (GNU/Hurd)
  + **Hybridní jádro** - kombinace (Win XP, Win Vista...)

**Funkce OS:** (správa procesů, paměti, zařízení, souborů, zabezpečení, uživatelské rozhraní)

* **OS** = podpora a rozhraní aplikací a hardware PC
* **Řídí a spravuje přístup ke zdrojům výpočetního systému** - rozdělování času procesoru, přidělování operační paměti, přístup k vnějším pamětem (disk, CD ROM, DVD), správa komunikace s periferiemi (klávesnice, myši, tiskárny).
* **Organizuje přístup k datům** - zamezení neoprávněného přístupu.
* **Řídí zpracování** **úloh** – jejich přípravu, plánování a průběh tak, aby byla zajištěna maximální efektivita jejich zpracování.
* **Podporuje komunikaci s** **uživatelem** - provádění uživatelem zadaných příkazů a spouštění aplikací, informuje uživatele o vzniklých problémech či chybách.
* Proces může pracovat ve více vláknech 🡪 běžící, spuštěný program, umístěný v RAM paměti v podobě sledu instrukcí, které vykonává CPU.
  + Obsahuje nejenom kód programu, ale také dynamicky se měnící data, která program zpracovává
  + Program je posloupnost instrukcí, která popisuje realizaci dané úlohy operačním systémem (PC)
    - Jeden program tak může v počítači běžet jako více procesů najednou s různými daty - více otevřených webových stránek najednou v jednom prohlížeči)

**Uspořádání dat na disku**

**Typy úložiště:**

* + **Directly attached storage** (DAS) - datový nosič přímo spojený s počítačem, který k němu přistupuje.
  + **Network attached storage** (NAS) - datové úložiště připojené k místní síti LAN.
  + **Storage area network** (SAN) - dedikovaná (oddělená od LAN, WAN, atd) datová síť, která slouží pro připojení externích zařízení k serverům
* V první řadě v počítači existují tzv. **soubory** -> skupina informací, které spolu souvisí – tvoří jeden celek.
* Každý soubor v počítači musí mít svůj **název** a **příponu**.

**TYPY SOUBORŮ**

Obsah obrázku snímek obrazovky

Popis byl vytvořen automaticky

**SLOŽKY**

* Jsou základním organizačním systémem OS Windows.
* Jejich množství na disku je omezeno pouze kapacitou disku.
* Existuje také tzv. **kořenový adresář** -> je nadřazen všem ostatním adresářům – tento adresář tedy obsahuje všechny ostatní adresáře v PC.

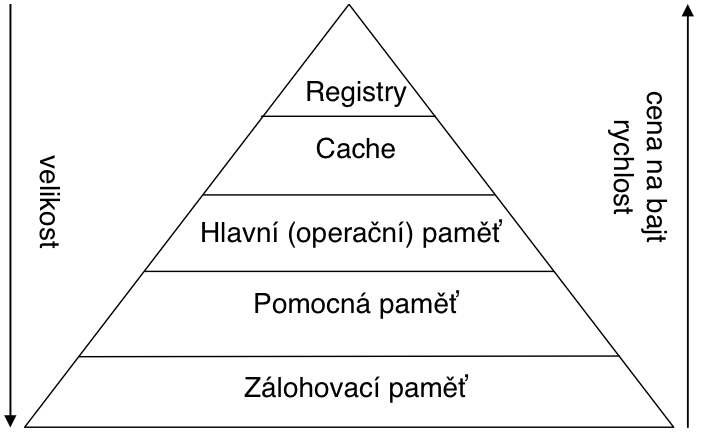
**STROMOVÁ STRUKTURA**

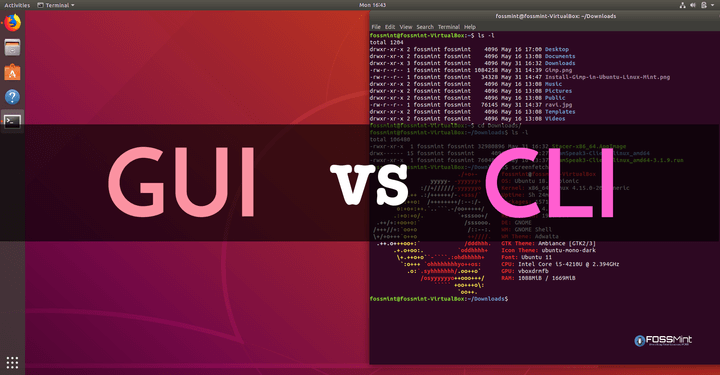
* Každá složka může obsahovat libovolné množství dalších složek, které můžou obsahovat další složky   
  -> vzniká nám tak **rozvětvený systém adresářů**.
*  V těchto složkách také může být obsaženo libovolné množství souborů (opět omezeno pouze kapacitou disku).
* **Cesta:**
  + Cestu k jednotlivým souborům uvidíte v záhlaví otevřené složky. Vždy začíná názvem (většinou písmenem) disku a pokračuje názvy všech složek a podsložek

**Správa aplikací**

* Kromě správy běhu aplikace (procesů) poskytuje většina OS i nástroje pro jejich instalaci, aktualizace a odstranění. Software lze na většině systémů spouštět i přímo, bez instalace řízené operačním systémem.
* Existují **dva základní koncepty**, jak instalaci nových programů provádět:
  + **Decentralizované distribuce** dá výrobce OS k dispozici nástroje pro vývojáře software k vytváření instalačních balíčků a ti je již vytvářejí a distribuují uživatelům samostatně (například instalace software ve Windows a Mac OS X).
    - -> uživatel si příslušnou aplikaci nahraje do počítače a provede její instalaci.  
      V poslední době je tento koncept již opouštěn.
  + **Centralizovaný přístup** využívá repositáře aplikací, který je provozován výrobcem OS a ze kterého si uživatelé z něj vybírají a instalují aplikace.
    - Jde o seznam ověřených aplikací, které jsou nahrány na centrálním uložišti, ze kterého jsou poté stahovány a instalovány prostřednictvím systémové aplikace. Takto zpracovaný software může operační systém také jednoduše kontrolovat a nabízet k němu příslušné aktualizace.

**Správa paměti:**



* přiděluje oper. paměť jednotlivým procesorům, když si jí vyžádají
* udržovat informace o paměti – která je volná a která je přidělená a komu
* zařazovat paměť, kterou procesy uvolní do volné části
* odebírat paměť procesům, je-li to zapotřebí
* **Správa zařízení:**
  + Systém spravuje a komunikuje se zařízeními pomocí ovladačů – driverů (driver = program, který umožňuje kontrolu daného zařízení)
  + Systém také rozhoduje, jaký proces bude používat jaké zařízení
* **Správa zabezpečení**
  + Zabezpečení poskytuje ochranu OS před viry, červy, malware a jinými útoky
  + Zajištění přístupu k OS pouze autorizovaným uživatelům
* **Správa UI**
* *UserInterface* = uživatelské rozhraní
  + **Grafické / Textové**

**Grafické uživatelské rozhraní**

* uživatelsky "přívětivější" než textový režim.
* obrázky, ikony, tlačítka…
* ovládání je intuitivní
* ovládá se myší, ale i pomocí kláves. zkratek

**Textové uživatelské prostředí**

* Složené výhradně z příkazového řádku a znaků (tj. písmen a číslic).
* Veškerá komunikace uživatele počítače je založena na zadávání příkazů a jejich parametrů.
* Práce je poměrně náročná – uživatel si musí pamatovat mnoho příkazů, musí je pracně vypisovat do příkazového řádku a všechny příkazy musí zadávat bezchybně a správně.
* Po zadání příkazu a odeslání klávesou ENTER počítač vykoná požadovanou operaci.
* **V současné době se používá grafický uživatelský režim**. Spustit textový režim ale přesto umožňují prakticky všechny současné operační systémy.

**Typy OS**(*multiuser, multitasking*…) pracovní stanice/servery/mobilní zařízení

* **Jednouživatelské jednoúlohové** - žádné prostředky ochrany souborů a disků, neumožňuje běh více procesů (programů) najednou, například: **MS-DOS**
* **Jednouživatelské víceúlohové** - multitasking, možnost paralelního (současného) běhu několika procesů, kooperativní multitasking, preemptivní multitasking, například: Windows 9x, Me, **Windows XP**
* **Víceuživatelské víceúlohové** - možnost zpracovávat požadavky více uživatelů přihlášených do systému, například: Windows NT, **Windows 2000**, Unix, Linux, Windows XP
* Systémy s reálným časem (real-time) – zejména pro řízení technologických procesů
* **MultiUser**
  + OS, na kterém může najednou pracovat více uživatelů – windows 7, windows server, linux
* **Pracovní stanice/servery/mobilní zařízení**
  + **Pracovní stanice**: Windows 7, Windows 10
  + **Servery**: Windows 2008, Windows 2012, Linux servery - Ubuntu server, debian
  + **Mobilní zařízení**: Android, iOS (Apple), BlackBerry OS, Ubuntu Phone

**Druhy OS**

* Windows (níže) | Linux (otázka 13)
* **Mac OS (Macintosh Operating System)**
  + OS určený pro počítače Apple Macintosh. Tento operační systém si nemůžeme pořídit na běžné PC, je určen výhradně pro hardware od společnosti Apple.
  + Aktuální operační systém nese označení Mac OS X.
* **MS DOS**
  + **DOS (diskový operační systém)**
  + textový operační systém (příkazy se zadávaly slovy)
  + mohl běžet pouze jeden program - každý program měl jiné ovládání, pro práci se soubory různé nadstavby, vznikly první grafické nadstavby – Windows
  + napomohl obecnému rozšíření osobních počítačů IBM, pro které byl určen

**Kompatibilita**

* **Softwarová kompatibilita** znamená, že konkrétní software je spustitelný v konkrétním softwarovém prostředí (určitý operační systém, běhové prostředí nebo běhová knihovna).
  + Některé programy mohou být na úrovni zdrojového kódu schopné běhu v různém softwarovém prostředí (například webový prohlížeč Mozilla Firefox může být přeložen pro různé softwarové platformy – Microsoft Windows, OS X nebo Linux
* **Hardwarová kompatibilita** - schopnost spolupráce jednotlivých komponent (počítače). V úvahu je potřeba brát nejen shodu konektorů, ale i použité elektrické napětí, použité signály, sběrnice a podobně.

**Historie OS**

* Windows 1. Byl uveden v roce 1985 (Bill Gates), Windows 10 v roce 2015; používaný teď
* Linux vznikl v roce 1991 (Linus Torvald)
* iOS 2007 (Steve Jobs)

**Licencování OS**

**Linux**

* Open source – otevřený zdrojový kód, několik různých licencí
* Nejčastěji **Apache License**; GPU License (můžeme kód modifikovat, musíme uvést autora, vše, co uděláme, děláme na své riziko)
* BSD licence

**Windows**

* Closed source, uzavřený systém, komerční, nutnost za OS zaplatit
* Nelze jej upravit a redistribuovat
* **Distribuován pod MS-PL** (Microsoft public license)  
  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ACPI**

* **= otevřená specifikace pro komunikaci jádra operačního systému s hardwarem.**
* Obsahuje spoustu funkcí, uživatelsky nejpotřebnější je zřejmě řízení spotřeby energie u počítačů či mobilních telefonů.
* Tato specifikace je nyní dostupná jak pro platformy Microsoft Windows, tak Linux.
  + Je definována za pomocí **několika stavů (G0-G3**), které vyjadřují určité stupně šetření energie. Je zde uvedeno, které komponenty, jak v konkrétním stavu pracují.
  + **G1 = spánek; G0 = zapnutý; G3 = spánek, nechá se napájet operační paměť**

**Důležité pojmy**

**Multitasking**

* Funkce umožňující souběžné zpracování více úloh v jednom okamžiku, tzn. že běží několik procesů současně.
  + Běh takto dvou spuštěných aplikací je však pouze zdánlivý. Ve skutečnosti procesor běžící procesy rychle střídá a uživateli vznikne dojem, že běží současně. Přepínání mezi procesy se nazývá **změna kontextu**.
* **Kooperativní multitasking** - Řízení si navzájem předávají jednotlivé procesy, je velmi zranitelný. V tomto případě záleží na každé úloze, jak často svým systémovým voláním umožní předat řízení zpět operačnímu systému, aby mohlo dojít ke spuštění jiné úlohy.  
  Velkou nevýhodou je ovšem, když úloha z důvodu chyby nepředá řízení nazpět. Tím je zastaveno provádění dalších úloh.
* **Preemtivní multitasking** - Zdroje přiděluje operační systém, havárie jednoho procesu neznamená ukončení práce celého systému. Funguje to tak, že v pravidelných intervalech, které nastanou 100 až 1000 krát do sekundy, dojde k přerušení právě běžící úlohy, aby systém zjistil, jaké další úlohy žádají o přístup na procesor a s jakou prioritou. Podle toho buď dostane přednost úloha jiná nebo se dále pokračuje v úloze právě prováděné.

**Driver (ovladač, ovladač zařízení)**

* Software, pomocí kterého dokáže operační systém zacházet s daným hardwarem - komunikovat s ním, obsluhovat ho.
* Spousta ovladačů je obsažena přímo v operačním systému, případně s aktualizacemi systému dochází k jejich doplňování. Proto často při zapojení např. tiskárny nepotřebujeme ovladače, které jsou se zařízením dodávané na CD nebo DVD.

**Bootování**

* Zavádění operačního systému do operační paměti RAM po zapnutí počítače.
* Jde o sekvenci úkonů, které se provedou, aby došlo ke správnému spuštění počítač. Mezi ně patří také kontrola funkčnosti součástek počítače, aktivace připojených zařízení a poté dochází k nahrávání části operačního systému do paměti RAM.
* označení procesu, který probíhá při spuštění nebo po restartu počítače.

**Proces**

* spuštěný a zpracovávaný počítačový program
* je umístěn v operační paměti počítače v podobě strojových instrukcí, které vykonává procesor v určitém sledu
* obsahuje nejen kód vykonávaného programu, ale i dynamicky měnící se data, které se zpracovávají
* jeden program může v počítači běžet jako více procesů s různými daty (např. vícekrát spuštěný webový prohlížeč zobrazující různé stránky)
* správu procesů vykonává operační systém, který zajišťuje jejich oddělený běh, **přiděluje** jim **systémové prostředky počítače** a umožňuje uživateli procesy spravovat (spouštět, ukončovat)

**Vlákno (vlákno řízení, thread)**

* Odlehčený proces, který snižuje režie operačního systému při střídání procesů na procesoru. Proces se tedy může dělit na vlákna, která pak pomocí sdílené paměti usnadňují komunikaci - vlákno vlastní paměť nemá, sdílí ji společně s ostatními vlákny běžícími ve stejném procesu.
* -> Vlákna tak mohou přistupovat ke stejným datům současně. To ovšem může přinést riziko tzv. **souběhu** (race condition). Zaleží také na tom, jestli operační systém vlákna podporuje nebo ne.
  + **Výhody**: program, kde se počítá určitý výpočet, a přitom se výsledek zobrazuje v podobě grafu. Výpočet je však náročný a trvá systému několik vteřin, než jej vypočte
* ****Z hlediska spravování vláken OS můžeme vlákna rozdělit na: - **vlákna na uživatelské úrovni** (ULT) - správu vláken provádí tzv. vláknová knihovna na úrovni aplikačního procesu **- vlákna na úrovni jádra** (KLT) - správa vláken je řízena OS

**SOHO (Small office/home office)**

* = soukromý podnik a osoby samostatně výdělečně činné. SOHO je často charakterizován jako malý jak ve velikosti kancelářských prostor, tak v počtu zaměstnanců.
* Obecně se za malou kancelář považuje každá firma s méně než 10 zaměstnanci

**NOS (Síťový operační systém)**

* OS, který spravuje síťové prostředky: v podstatě operační systém, který obsahuje speciální funkce pro připojení počítačů a zařízení do místní sítě (LAN).
* NOS spravuje více požadavků (vstupů) současně a poskytuje zabezpečení nutné ve víceuživatelském prostředí.

**Virtualizace**

* vytvoření virtuálního, zdánlivého počítače uvnitř počítače skutečného (fyzického) pomocí vhodného softwaru.
* Virtualizovat lze i jednotlivé aplikace.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

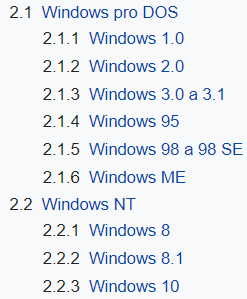
## OS Windows

* 🡪 několik různých operačních systémů od firmy Microsoft, všechny mají grafické uživatelské rozhraní, avšak liší se použitým jádrem systému, úrovní podpory multitaskingu (současného běhu více úloh najednou) i používanými knihovnami a účelem použití.
* V současné době jsou aktuální desktopové verze Windows 8 a Windows 10 nebo serverové edice Windows Server 2008, Windows Server 2012 nebo Windows Server 2016.
* Narozdíl od OS Linux **není tento systém open-source.**  🡪 pokud ho budeme chtít používat musíme si zaplatit.

**Obsah obrázku snímek obrazovky

Popis byl vytvořen automatickyInstalace**

* **Historie**
* Windows pro DOS je označení pro první grafický systém, který firma Microsoft začala prodávat v roce 1985.



\****NT = New Technology***

## Obsah obrázku snímek obrazovky Popis byl vytvořen automatickyHardwarové požadavky

## Aktualizace

* Abychom nemuseli stahovat a instalovat aktualizace pro Windows ručně, běží v operačním systému služba: **Windows Update**, která toto obstarává automatický místo nás.
* Při kontrole dostupnosti aktualizací může služba Windows Update nejprve provést také vlastní aktualizaci, poté instaluje potřebné balíčky či opravy

## Uživatelské účty

* Systém Windows nabízí tři typy uživatelských účtů: **Administrátor**, **Standardní** a **Host**
* Windows dává každému typu účtu oprávnění k provádění různých věcí v počítači.

**Správce:**

* + Správce řídí celý počítač a rozhoduje, kdo s ním bude hrát a co může každý uživatel na něm dělat. Zřizuje účty pro každého člena domácnosti a rozhodne, co může a nemůže dělat s počítačem.

**Standardní:**

* + Standardní držitelé účtu mají přístup k většině počítačů, ale nemohou na nich provádět žádné velké změny. Nemohou například spouštět nebo instalovat nové programy, ale mohou spouštět stávající programy.

**Host:**

* + Mohou používat PC, ale počítač je nerozpozná podle jména. Účty hosta fungují podobně jako standardní účty, ale **bez soukromí**: K účtu hosta se může přihlásit kdokoli a počítač bude vypadat tak, jak jej opustil poslední host.

**Oprávnění**

* Žádný systém Windows pro DOS neposkytoval možnost nastavení oprávnění v souborovém systému FAT na soubory a adresáře, i když tyto systémy nabízely možnost definování uživatelských profilů pro různé uživatele.
  + Teprve Windows NT přinesly možnost definování oprávnění, avšak pouze v souborovém systému NTFS.
  + Současně je v jádře systému podpora pro rozlišení jednotlivých uživatelů a jejich procesů spolu s omezením jejich možností zasahovat si navzájem do své práce.

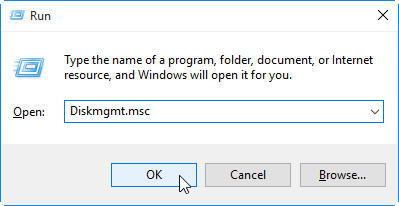
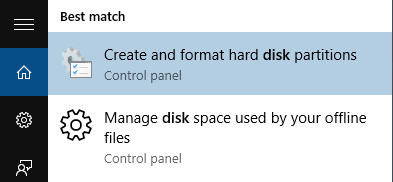
**Nepreemptivní multitasking**

* Jádro systémů Windows pro DOS používalo nepreemptivní multitasking
  + Aplikace se musely samy vzdát procesoru pomocí speciálního volání služby operačního systému.
  + Pokud byla taková aplikace špatně naprogramována, ponechala si procesor jen sama pro sebe a ostatní aplikace i části operačního systému se k procesoru nedostaly (neběžely).
* Jádro systémů Windows NT využívalo naplno schopnosti procesorů, a proto podporovalo preemptivní multitasking.

**Víceuživatelský systém**

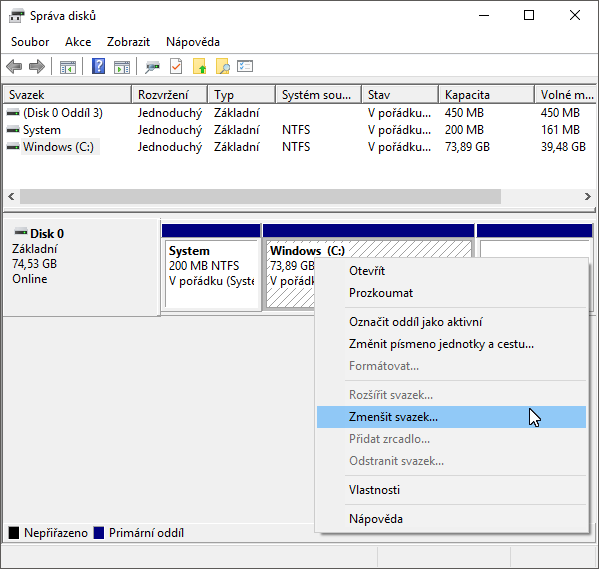
* Systémy Windows pro DOS i Windows NT umožňují definovat v systému uživatele a používat pro každého z nich jiný profil (tj. nastavení prostředí, nastavení jednotlivých programů, vlastní domácí adresář).
* Takový systém však není možné považovat za víceuživatelský, protože **jednotliví uživatelé nepracují v systému zároveň**, nýbrž postupně (po odhlášení se může přihlásit jiný uživatel).
* Systém Windows NT 4.0, který již podporoval systém oprávnění, byl v roce 1996 vydán ve variantě Terminal Server, která umožňovala současnou práci se systémem více uživatelům zároveň pomocí **Remote Desktop Services** (v té době nazývané Terminal Services).
  + Tato možnost je dostupná pouze pro serverové vydání Windows NT (poslední je Windows Server 2008).
* Desktopové systémy (Windows XP, Windows Vista a Windows 7) obsahují pouze částečnou podporu Remote Desktop Services, která umožňuje v jednom okamžiku přihlášení pouze jediného uživatele (tzv. Vzdálená plocha).

## Správa disků

* Rozšíření oddílu na disku lze provést **třemi způsoby:**
* **Způsob 1:** Nejjednodušší způsob, jak otevřít správu disků v systému Windows 10, je z plochy počítače.  
  Klikněte pravým tlačítkem myši na nabídku Start (nebo stiskněte klávesovou zkratku Windows + X) a poté vyberte „*Správa disků*“.
* **Způsob 2:** Pomocí klávesové zkratky Windows + R otevřete okno Run. Poté zadejte „Diskmgmt.msc“ a klikněte na „OK“ nebo stiskněte klávesu „Enter“.
* **Způsob 3:** Do vyhledávacího pole zadejte přímo „*správu disků*“ a z výsledků vyberte „Vytvořit a formátovat oddíly pevného disku“.

**Vytvoření nového oddílu**

* Rozdělení disku začneme zmenšením stávajícího oddílu. Klikneme pravým tlačítkem myši na oddíl **C** a zvolíme příkaz **Zmenšit svazek**.



* Velikost, o kterou je možné oddíl zmenšit, je závislá na zaplněnosti oddílu.

## Zálohování

* Stačí si připravit médium pro zálohu, nejlépe externí disk nebo mít k dispozici síťové úložiště (NAS).
* Poté otevřete vyhledávání **Win+Q**, vyhledejte a spusťte **Zálohování a obnovení (Windows 10)**.

**Možnosti zálohování**

* zálohovat jednorázově bitovou kopii systému (**Vytvořit bitovou kopii systému**)
* nastavit pravidelné zálohování (**Nastavit zálohování**).
  + Při pravidelném zálohování lze do zálohy zahrnout i aktuální bitovou kopii systému.

1. Příkazové rozhraní, OS Linux

**Příkazové rozhraní**

**Výhody, nevýhody CLI vs. GUI**

CLI - Command Line Interface (pouze textové prostředí)

* Představuje uživatelské rozhraní, ve kterém uživatel s programy nebo operačním systémem komunikuje zapisováním příkazů do příkazového řádku.
* Na rozdíl od textového rozhraní a grafického uživatelského rozhraní nevyužívá myš ani menu a nedovede pracovat s celou plochou obrazovky (terminálu).

GUI - Graphical User Interface (grafická nadstavba - okna, tlačítka, …)

* Uživatelské rozhraní, které umožňuje ovládat počítač pomocí interaktivních grafických ovládacích prvků. Na monitoru počítače jsou zobrazena okna, ve kterých programy zobrazují svůj výstup.
* Uživatel používá klávesnici, myš a grafické vstupní prvky jako jsou menu, ikony, tlačítka, posuvníky, formuláře a podobně.

**Výhody CLI:**

* Rychlejší než GUI
* Většinou na serverech
* Nevyužívá se myš
* Vyšší bezpečnost

**Nevýhody CLI:**

* Méně intuitivní
* Nutnost znát příkazy
* Větší administrátorská znalost systémů

**Výhody GUI:**

* Větší přehlednost
* Výhodnější pro běžné uživatele
* Méně náročné na ovládání
* Není nutná úplná znalost příkazů pro CLI

**Nevýhody GUI:**

* Méně bezpečné
* Pomalejší prostředí (načítání více věcí pro uživatele (grafika oken programů, …))

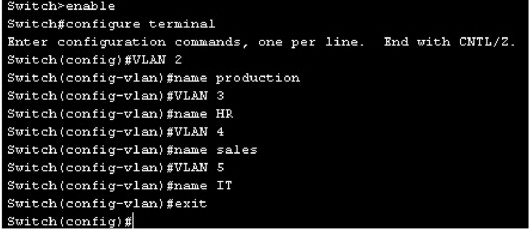
**Shell - popis, příklady CLI implementací**

* Je označení programu, který vytváří v počítači rozhraní pro práci uživatele, zpravidla v prostředí příkazového řádku.
* Shell umožňuje uživateli využívat služby operačního systému, zejména spouštět programy, zajišťovat pro ně vstupy, zobrazovat, uchovávat a přesměrovávat jejich výstupy, spojovat jednotlivé programy do kolon a podobně.
* Funkce: interpretace příkazů, generování a interpretace příkazových procedur, přesměrování souborů, řízení prostředí

**Prompt**

* Symbol či text, který se objeví v příkazové řádce nebo v programu počítače ve chvíli, kdy je počítač připraven přijmout vstup od uživatele
* Například:

C:/Windows> znamená, že se uživatel právě nachází v systémovém adresáři Windows na disku C.

* Může se měnit dle režimů, které příkazová řádka poskytuje (například v Ciscu existují celkem tři režimy):
  + **User EXEC mode =**
  + Omezené možnosti
  + Užitečný pro základní operace
  + Pouze omezené příkazy
  + Tento režim vypadá takto: Switch> // Router> ….
  + **Privileged EXEC mode =**
  + Vyšší konfigurační režim.
  + Někdy se mu také říká “enable mode”.
  + Na Unixových systémech podobný módu “sudo”.
  + Lze využít všechny příkazy
  + Vypadá takto: Switch# // Router# ….
  + **Globální konfigurační režim**
  + Slouží ke konfiguraci a základnímu nastavení zařízení
  + Poznáme jej: například Switch(config)# …..
  + V globálním konfiguračním režimu existují také režimy “sub-konfigurační”, které umožňují konfiguraci určité části nebo funkce zařízení IOS. Dva nejběžnější jsou:
    - **Line configuration mode** = Slouží ke konfiguraci konzole, Telnet, SSH nebo AUX přístupu. Značí se: např. Switch(config-line)#
    - **Interface configuration mode** = slouží ke konfiguraci portů zařízení.
  + Značí se: např. Switch(config-if)#
  + Mezi konfiguračními režimy se přepíná pomocí příkazu: “enable”  a “disable”
  + Přepínání mezi globálním režimem konfigurace se používá příkaz: “Configure terminal privileged EXEC mode”. Pro návrat zpět, použijte: “exit global config mode”.  
    

**Formát příkazu, syntaxe**

* Obecná syntaxe pro  příkazy: “Příkaz -parametr argumentPříkazu”
* Parametr definuje vlastnosti použitého příkazu (jak má příkaz naložit s argumentem) (např. “ls -a “)
* Argument je např. Proměnná. - Definuje vstup pro příkaz (například ls -a /var/www/)

**Aliasy, proměnné**

* Aliasy nám umožňují snadnější zápis příkazu
* Př -  Get-childItem - ls, gci

**Historie příkazů**

* Možnost vrátit se k výše zadaným příkazům
* Ukládá se do /.bash\_history
* Umístění uložení příkazů lze změnit, také počet zapamatovatelných příkazů
* History -c = maže historii

**PATH**

Systémová proměnná (cesta), ve které systém hledá příkazy, zadané uživatelem

**Globbing (zástupné symboly)**

* Zástupné znaky v příkazech
* Zpracovává shell
* Interpretuje stand. Wild card charaktery

**Quoting:**

* Uvozovky
* dvojité uvozovky “text je $text” ← vypíše včetně hodnoty proměnné $text. Brání interpretaci pouze některým znakům
* jednoduché uvozovky se brání v interpretaci jakýmkoliv znakům - vše se převádí na string.

**Escaping**

* Speciáloní významové charaktery, které jsou umístěny za zpětným lomítkem \n = nový řádek
* \r = zpět
* \t = tabulátor

**Podmíněné spojování příkazů**

* logické spojky v příkazech
* ; - nehledě na to, zda to skončí chybou nebo ne, vykonej část za středníkem
* || - pokud předchozí příkaz skončí chybou, vykonej část za znaky
* && - pokud předchozí příkaz neskončí chybou, vykonej část za znaky

**Roury**

* Výstup jednoho programu přesměruje na vstup programu druhého
* Př.: C:\>dir /b /s \*.txt | more //\* stránkování dlouhých doc.**Standardní vstup, standardní výstup a standardní chybový výstup, přesměrovávání**

<http://www.abclinuxu.cz/ucebnice/zaklady/prikazova-radka/standardni-vstup-a-vystup>

* Vstup - /stdin/ = místo, ze ktrého programy berou data
* Výstup /stdout/ = místo, kam data vpisují
* Chybový výstup - /stderr/
* Cat = čte data ze vstupu a vypisuje je na výstup (vstup - klávesnice, výstup - obrazovka)
* &> - přesměrovává chybový a standardní výstup

**Vyhledávání, řazení, vypisování**

* Řazení pomocí sort

**Skriptování (spouštění skriptu, proměnné, podmínky, cykly)**

* Skripty pro powershell - koncovka “.ps1”.
* Argumenty pro PS skripty získáme: args[x]
* **Cykly:** 
  + ForEach
  + ForEach-Object (cyklus pro objekty)
  + For (cyklus s přesným počtem opakování)
  + While (cyklus s podmínkou na začátku)
  + Do While (cyklus s podmínkou na konci)
* $\_. ← příkaz v cyklu foreach-Object, používá se pro odkaz na aktuální objekt, který je parsován skrz pipe.-
* **Porovnávací operátory:**
  + -eq (rovno)
  + -ne (nerovno)
  + -gt (větší než)
  + -lt (less than - menší než)
  + -le (menší nebo rovno než)
  + -ge (větší nebo rovno než)
* Get-ChildItem - aliasy: ls/ gci / dir
* K použití rekurze slouží argument –**Recurse**
* Příkaz Test-Path slouží k otestování, zda něco někde existuje - lze použít v Ifu
* Přesun něčeho: Move-Item
* Kopírování něčeho: Copy-Item
* Odstranění něčeho: Remove-Item
* Pokud chceme potvrzení; parametr -Confirm
* Vypsání všeho, včetně skrytých souborů: -Force
* Hláška “Co by se stalo kdyby se provedl příkaz bez tohoto parametru” -WhatIf
* hashtable - pole, který může uchovávat vícero typů proměnných
* Proměnné prostředí se v Powershellu značí env:
* ScriptBlock = část programu/příkazu, který je ohraničen složenými závorkami
* skript pro argumenty:

*Write-Host "Num Args:" $args.Length;*

*foreach ($arg in $args)*

*{*

*Write-Host "Arg: $arg";*

*}*

**Parametry:**

* věci, které ovlivňují chod skriptu
* přepínače (např. -Recurse; -.Filter\*.cs (vypíše všechno, co má příponu .cs))

**Nápověda, manuály**

* Pomocí příkazu man+daný příkaz - součást balíku man-db
* - nápověda syntaxe
* Apropos - prohledává u man tránek sekci název a popis
* Whatis - popis daného příkazu

**OS LINUX**

**Obecná charakteristika OS - hledisko využití, hledisko uživatele**

* svobodný a otevřený počítačový operační systém, který je založený na linuxovém jádru.
* Linuxové systémy jsou šířeny v podobě distribucí, které je možné nainstalovat nebo používat bez instalace (tzv. live CD).
* Používané licence umožňují systém zdarma a velmi volně používat, distribuovat (kopírovat, sdílet) i upravovat.
* Tím se odlišuje od proprietárních systémů (např. Windows či macOS), za které je nutné platit a dodržovat omezující licence.
* Na tomto jádru se poté budují operační systémy - linuxové distribuce (Debian, Ubuntu, Fedora,...)
* Vychází z Unixu, což je proprietární jádro OS
* Linux je open source software, což znamená, že jsou k dispozici jeho zdrojové kódy, které lze za dodržení jistých podmínek upravovat a vše dále šířit.
* **Vyvinul jej Linus Torvalds**
* Může mít podobu grafického prostředí či terminálového prostředí (příkazová řádka)
* Hlavně pro zkušenější uživatele, administrátory

**Funkce OS**

* Opensource – otevřený kód, takže jej může kdokoliv upravovat
* Mnoho distribucí, každá na jiný účel, existují i speciální distribuce pro specifický účel (Raspbian, určený pro mikropočítač raspberry Pi, ...)
* Mnohostranné využití - servery, mikropočítače, telefony, drony, superpočítače
* Zabírá velmi málo místa na disku; možnost využití jiného grafického prostředí

**Hardwarové požadavky**

* + 1GB RAM (doporučuje se 2 GB pro pohodlné používání).
  + 15 GB volného místa na disku (doporučeno 20GB).
  + Grafická karta schopná rozlišení 1024 × 768px.
  + CD/DVD mechanika nebo USB port  
    Složení OS
* Linuxové jádro - kernel 🡪 kód, který zajišťuje komunikaci mezi hardwarem a samotnou distribucí
* Systémové Knihovny - další důležitou součástí OS; obsahují programové funkce například pro vykreslování tlačítek a jiných grafických prvků, programy pro práci se soubory - ls; atp
* Programy a procesy
* Shell - program, který zpracovává příkazy zadané uživatelem a předává je jádru
* Pravidla vyplývající z licence GNU/GPL
* Možnost úpravy zdrojového kódu
* Možnost využít software i pro komerční účely, možnost sdílení sw s ostatními lidmi
* Uživatel musí při šíření sw uvádět GNU licenci, copyright, instalační instrukce k sw, který šíří

**Co je to distribuce OS?**

* Linuxová distribuce je operační systém, založený na Linuxovém jádře - to mají všechny distribuce stejné
* Dále distribuce obsahuje linuxové knihovny, GNU nástroje, nějaké uživatelské prostředí - například Gnome, Cinnamon, KDE, …
* Jsou hlavní nadstavby na linuxovém jádře - například Debian a Red Hat
* Z těchto dvou se poté odvíjí další OS, jako je Ubuntu, Raspbian - založený na Debianu - jsou tedy založený na stejném OS, ale je upravený jejich kód tak, že vypadají jinak - Ubuntu má prostředí Unity, zatímco Debian má jako výchozí volbu Gnome.
* Používají také jiné balíčkovací systémy – repozitáře - Debian má .deb; Red Hat má .rpm balíčkovací systémy
* Z Red Hatu vycházejí  distribuce jako je například Fedora či CentOS
* Je zde také distribuce, která je klon zmiňovaných dvou - SUSE

**Pojmy: Kernel, Shell, Terminál**

* **Kernel** = Jádro OS =  kód, který zajišťuje komunikaci mezi hardwarem a samotnou distribucí
* **Shell** - program, který zpracovává příkazy zadané uživatelem a předává je jádru; dva hlavní shelly - v Linuxu typicky Bourne Shell - Bash; umožňuje programovat skripty, které usnadňují práci se systémem a automatizuje úkony (značka $)
  + Adresář Shellu je v /bin/sh
* **Terminál** - příkazový řádek v Linuxu, umožňuje zadávat příkazy, které poté předává shellu

**Pravidla pro pojmenovávání úložišť (diskové oddíly, přenosná média)**

* Disky se pojmenovávají jako /sdaX - X může označovat celý disk či diskové oddíly (dev/sda1 /dev/sda2 /dev/sdb4 atd…
* Typicky když je připojeno více disků, tak se oddělují například /dev/sda a /dev/sdb
* A oddíly na nich se označují číslem /dev/sda1 /dev/sda2, /dev/sdb1 /dev/sdb2 atp

**Popis struktury zápisu cesty k souboru, vysvětlit rozdíl mezi relativní a absolutní adresou**

* Zápis cesty k souboru - relativní cesta
  + Pro pohyb mezi složkami se používá příkaz cd - change directory
  + Relativní cesta je postupná, tj, můžeme jí v průběhu změnit - několik příkazů cd (například cd var; cd www (/var/www/html, atp.)
* Zápis cesty k souboru - absolutní cesta
  + Absolutní, předem daná – označuje se z kořenového adresáře tedy lomítkem před názvem adresáře například *cd /var/www/html* - je okamžitá cesta do adresáře webového serveru www na systému
  + Pro rozpoznání aktuálního adresáře, ve kterém se nacházíme - jeho absolutní cesty - slouží příkaz pwd
* Popsat standard adresářové struktury (FHS)
  + Kořenový adresář - lomítko /;
  + podsložky jako jsou boot, obsahující informace o startu systému;
  + home, obsahující domovský adresář každého uživatele;
  + mnt, slouží pro zobrazení dočasně připojených filesystémů (například USB driverů, externího disku, atp..)
  + /root - složka administrátora
  + /proc - adresář, obsahující informace o běžících procesech
  + /etc - obsahuje konfigurační podsložky a soubory služeb

**Praktické úkoly:**

* **Ipconfig** – informuje o síťových rozhraních ve vašem systému
* **Ping** – umožňuje prověřit funkčnost spojení mezi dvěma síťovými rozhraními
* **Cd** – umožní změnit aktuální pracovní adresář
* **Mv** – přesune soubor do jiného umístění nebo soubor přejmenuje, pokud je jako druhý argument uveden název souboru, soubor se přejmenovává, pokud je jako druhý argument uveden název adresáře, soubor se přesouvá
* **Man** – ukáže manuál (návod) zvoleného příkazu
* **Mkdir** – vytvoří nový adresář zadaného jména
* **Adduser** – vytvoří nového obecného uživatele ve vašem systému
* **Gpasswd** – nastavení hesla skupiny
* **Script** – zadáním „script mujzaznam.txt“ se začnou do souboru se jménem mujzaznam.txt zaznamenávat všechny příkazy zadané z příkazového řádku a zároveň i odezvy systému. Zaznamenávání ukončíme stiskem Ctrl+D
* **Userdel** – vymaže uživatele
* **Cp** – vytvoří kopii zadaného souboru
* **Ls** – ukáže seznam souborů ve vašem aktuálním adresáři

1. Síťové služby

* **Počítačová síť** – spojení 2 nebo více počítačů tak, aby mohly navzájem komunikovat

**Základní služby**

* sdílení technických zařízení (disky, tiskárny)
* přístup na ostatní počítače (telnet, rlogin)
* přenos zpráv (e-mail, news)
* přenos souborů (NFS, ftp, www)
* správa sítě (DNS)

**DHCP**

* Protokol umožňuje prostřednictvím DHCP serveru nastavovat stanicím v počítačové síti sadu parametrů nutných pro komunikaci pomocí IP protokolu.
* Umožňuje předávat i doplňující a uživatelsky definované parametry.  
  Významným způsobem tak zjednodušuje a centralizuje správu počítačové sítě (například při přidávání nových stanic, hromadné změně parametrů nebo pro skrytí technických detailů před uživateli).
* DHCP servery mohou být sdruženy do skupin, aby bylo přidělování adres **odolné vůči výpadkům**.
* Typicky se pomocí DHCP nastavují tyto parametry:
  + IP adresa
  + maska sítě
  + implicitní brána ( default gateway)
  + DNS server (seznam jedné nebo více IP adres DNS serverů)

**Bezstavová autokonfigurace adres (SLAAC)**

* Host v IPv6 může být konfigurován automaticky, pokud je připojen na směrovanou IPv6 síť, za použití zpráv směrem k ICMP v6 směrovači. Při prvním připojení k síti host vyšle ***'router solicitation'*** multicast žádost o konfigurační parametry na místní linku.
* Odpovídajícím způsobem nastavený ICMPv6 směrovač odpoví na tuto žádost paketem ***'router advertisement'***, který obsahuje konfigurační parametry síťové vrstvy.
* SLAAC používá k práci **protokol NDP**. (autokonfiguraci síťových rozhraní)

Obsah obrázku snímek obrazovky

Popis byl vytvořen automaticky

**DHCPv6**

* Rozlišuje dva základní režimy:

**bezstavové (Stateless) DHCPv6**

* + Klient vždy začíná *Router Solicitation* a router v segmentu reaguje *Router Advertising*. Tentokrát má *Router Advertising* příznak nazvaný **other-config** nastavený na **1.**  
    Jakmile klient obdrží zprávu, bude stále používat SLAAC k vytvoření své vlastní adresy IPv6. Vlajka však klientovi říká, aby udělal něco víc.
  + Po úspěšném procesu SLAAC klient vytvoří **požadavek DHCPv6** a odešle jej prostřednictvím sítě. Server DHCPv6 nakonec odpoví všemi dalšími potřebnými informacemi, jako je DNS server nebo název domény.
  + Tento přístup nazýváme bez státní příslušnosti, protože server DHCPv6 **nespravuje žádný pronájem pro klienty**. Místo toho pouze poskytuje další informace podle potřeby.

**Stavové (Stateful) DHCPv6**

* + S tímto přístupem zpracovává server DHCPv6 nájem a další informace, stejně jako tradiční DHCP IPv4.
  + Klient začíná znovu s Router *Solicitation*, ale tentokrát *Advertisement* v odpovědi obsahuje **managed-configs** na **1**.  
    Toto říká klientovi, aby místo **toho nepoužíval SLAAC** , ale DHCPv6. Klient poté vygeneruje požadavek DHCPv6, aby získal jak adresy, tak další informace.

**Sdílení technických zařízení sítě**

* Síť LAN umožňuje používat všem účastníkům sítě společná technická zařízení.
  + Obvykle se jedná o velkokapacitní disky, laserové tiskárny, a další speciální hardware.
  + Sdílené prostředky jsou typicky ostatním počítačům nějakým způsobem nabízeny, například lze zobrazit všechny dostupné sdílené prostředky v síti.
  + Někdy je však nutné, aby uživatel přesně znal název a umístění sdíleného prostředku.
  + Klienti si sdílený prostředek vhodným způsobem připojí, například jako další disk (Microsoft Windows) nebo jako obsah vybraného adresáře (unixové systémy).

**Sdílení společných dat**

* 🡪 bývá nejčastější příčinou budování počítačových sítí LAN. Všichni uživatelé sítě mohou v síti využívat a zpracovávat společná data.
* Tato služba se používá, jakmile potřebuje větší počet pracovníků přístup ke stejným datům.
* Sdílené prostředky a data jsou typicky chráněny nutností **autentizace uživatele** přihlašovacím jménem a heslem (možné jsou i jiné autentizační mechanismy).
  + Po úspěšné autentizaci jsou dále aplikována přístupová oprávnění (oprávnění v systému souborů, oprávnění tisknout atp.).
* Konkrétní detaily jsou závislé na použité platformě.
* V případě programátorské chyby může na obou stranách síťové komunikace vniknout bezpečnostní problém, který označujeme jako **zranitelnost**.
  + Program využívající zranitelnost je pak **exploit**, který umožňuje zranitelnost zneužít.
* Proti nim se lze chránit zejména principem nejmenšího oprávnění, pravidelnými aktualizacemi softwarového vybavení, použitím firewallu, VPN (virtuální privátní sítě) a podobně.

**FTP je protokol pro přenos souborů mezi počítači pomocí počítačové sítě.**

* Využívá protokol TCP a může být používán nezávisle na použitém operačním systému (je platformně nezávislý).
* Jeho podpora je součástí webových prohlížečů nebo specializovaných programů (tzv. FTP klientů).
* **Network File System** (NFS) je internetový protokol pro vzdálený přístup k souborům přes počítačovou síť.
  + Vývoj má na starosti organizace Internet Engineering Task Force (IETF).
* Funguje především nad transportním **protokolem UDP**, avšak od verze 3 je možné ho provozovat také nad **protokolem TCP**.
* V praxi si můžete prostřednictvím NFS klienta připojit disk ze vzdáleného serveru a pracovat s ním jako s lokálním.
* V prostředí Linuxu se jedná asi o nejpoužívanější protokol pro tyto účely.
* Za kolegu NFS lze považovat **protokol NIS** (Network Information Service), který slouží k distribuci nejrůznějších konfiguračních dat v počítačové síti (např. uživatelé, doménová jména).

**Monitorování účastníků sítě**

* + - Výhodou služby je hlavně kontrola práce v síti.  
      Monitorování umožňuje zobrazovat na obrazovce počítače, obsahy obrazovek jiných počítačů.
    - Pokud je použit nějaký komplexní monitorovací systém pro dohled serverů, tak jsou většinou dvě základní možnosti pro přístup k informacím.
      1. **Monitorování s agentem**, kdy se na server instaluje speciální klient.
         * Musí tedy být k dispozici agent pro daný operační systém a musí být možnost jej na sever instalovat.
      2. **Monitorování bez agenta**, kdy se testují vlastní služby serveru nebo se data získávají pomocí určitých standardních protokolů (jako SNMP, WMI, IPMI).

**Seznam technologií, které lze pro monitoring použít:**

* + dostupnost serveru pomocí **ping testu**
  + dostupnost služby pomocí **navázání TCP spojení** nebo na aplikační úrovni
  + události ze serverů – Syslog
  + získávání údajů pomocí klienta
  + získávání údajů pomocí monitorovacích protokolů WMI, SNMP, IPMI
  + sledování síťových toků – **NetFlow**
  + analýza síťových protokolů – **network protocol analyzer**
  + bezpečnost v síti – **IDS/IPS**

**WWW**

* systém webových stránek zobrazovaných pomocí webového prohlížeče
* běžně používá protokol **HTTP**, pro zabezpečený přenos používá protokol **HTTPS**

**E-mail – elektronická pošta**

* + pro přenos zpráv používá **protokol SMTP** (25, 587, 465)
  + pro komunikaci s poštovními programy používá protokoly **POP3, IMAP**
  + **Instant messaging** – online (přímá, živá) komunikace mezi uživateli

**VoIP – telefonování pomocí internetu**

**FTP – přenos souborů**

* + služba se jmenuje stejně, jako protokol, port 20, 21

**DNS**

* Protokol, který **zajišťuje překlad názvů domén webových stránek** z nepřehledné (číselné) podoby využívané stroji na tzv. „doménové jméno“ – tedy název, který vidíte v prohlížeči a který zadáváte, když chcete na stránku vstoupit.
* Úkolem DNS serveru je zjistit podle doménového jména zadaného uživatelem číselnou adresu (IP adresu), která se pod tímto jménem skrývá a „vytočit“ ji. Tím se dostanete na požadovanou stránku.

**DNS servery**

**Rekurzivní**

* + **Opakují**, jsou odpovědné za poskytnutí správné IP adresy zamýšlené domény požadujícímu hostiteli.
  + Například při zadávání požadavku na web z vašeho prohlížeče hostitel (počítač) poté požádá rekurzivní server DNS o nalezení adresy IP přidružené k webu, to je za předpokladu, že váš operační systém a webový prohlížeč je už ji mají v mezipaměti.
  + Odtud rekurzivní server zkontroluje, zda je IP v mezipaměti a stále má platnou dobu do života (TTL). Pokud rekurzivní server nemá IP v mezipaměti, začne rekurzivní proces (opakování procesu a odkazování zpět na sebe) procházející autoritativní hierarchií serveru DNS.

**Autoritativní**

* + jsou zodpovědné za IP mapování zamýšleného webu.
  + Autoritativní odezva serverů na rekurzivní servery **obsahuje důležité informace pro každý web**, například: odpovídající adresy IP a další nezbytné záznamy DNS.
  + Autoritativní servery jsou obvykle spravovány webhostingovými společnostmi.

**Sdílení souborů**

* + NFS, GFS, AFS, …
  + protokol **SMB** – sdílení v sítích s Microsoft Windows

**Připojení ke vzdálenému počítači**

* + **Telnet** – klasický textový terminálový přístup
  + **SSH** – zabezpečená náhrada protokolu telnet
  + **VNC** – připojení ke grafickému uživatelskému prostředí
  + **RDP** – připojení ke grafickému uživatelskému prostředí v Microsoft Windows (proprietární protokol)

**Služební protokoly**

* **DHCP** (67) – automatická konfigurace stanic pro komunikaci v sítích s TCP/IP
* **SNMP** (161) – správa a monitorování síťových prvků

1. Číselné soustavy a uložení čísel v počítači

**Typy číselných soustav**

**Poziční soustavy**

* Poziční soustavy jsou charakterizovány tzv. základem neboli bází, což je obvykle kladné celé číslo definující maximální počet číslic, které jsou v dané soustavě k dispozici.
* Poziční soustavy (kromě jedničkové) se nazývají také *polyadické*, což značí vlastnost, že číslo v nich zapsané lze vyjádřit součtem mocnin základu dané soustavy vynásobených příslušnými platnými číslicemi.
* Každé číslo vyjádřené v poziční soustavě (kromě jedničkové) může mít část celočíselnou a část zlomkovou (např. u desítkové soustavy desetinnou část).
* Tyto části jsou odděleny znakem, nazývaným desetinnou čárkou (přestože obecně nejde o desetiny).
* Existují i soustavy, které využívají odečítání.
* Mezi nejčastěji používané poziční číselné soustavy patří:

**Jedničková (unární, r = 1)**

* + Přestože si to ani neuvědomujeme, tuto soustavu běžně používáme při počítání na prstech nebo při psaní čárek označujících počet piv na účet v restauracích.
  + Může být řazena mezi speciální poziční soustavy nebo i zcela mimo dělení na poziční/nepoziční soustavy.

**Dvojková (binární, r = 2)**

* + Přímá implementace v digitálních elektronických obvodech (použitím logických členů), čili interně ji používají všechny moderní počítače

**Osmičková (oktální, oktalová, r = 8)**

**Desítková (decimální, dekadická, r = 10)**

* + Nejpoužívanější v běžném životě

**Dvanáctková (r = 12)**

* + Dnes málo používaná, ale dodnes z ní zbyly názvy prvních dvou řádů – tucet a veletucet

**Šestnáctková (hexadecimální, r = 16)**

* + Používá se v oblasti informatiky, pro číslice 10 až 15 se používají **písmena A až F**

**Šedesátková (r = 60)**

* + Používá se k měření času pro zlomky hodiny; číslice se obvykle zapisují desítkovou soustavou jako 00 až 59 a řády se oddělují dvojtečkou; staré názvy prvních dvou řádů jsou kopa a velekopa.

**Nepoziční soustavy**

* Římské číslice jsou příkladem nepoziční číselné soustavy.
* Dnes se prakticky nepoužívají.

**Metody převodu mezi číselnými soustavami**

**Převod z desítkové do dvojkové**

* Číslo se dělí 2 a zbytek se zapisuje, dokud se nedostaneme do fáze, kdy máme 1 : 2.
* Takže to může vypadat takto:

70 : 2 = 35 0

35 : 2 = 17 1

17 : 2 = 8 1

8 : 2 = 4 0

4 : 2 = 2 0

2 : 2 = 1 0

1 : 2 = 0 1

* Číslo **70** se poté zapíše zespoda, takže bude vypadat takto: **1000110**
* U dalších soustav je to stejné s výjimkou u 16tkové, kde se čísla 10 - 15 nahrazují písmeny.

**10 - A, 11- B, 12 - C, 13 - D, 14 - E, 15 - F**

185 : 16 = 11 9

11 : 16 = 0 11

* Takže číslo 185 se převede na **B9**

**Převod z dvojkové do desítkové soustavy**

* Opačně bychom převedli takto.
* Mějme číslo 1100010 a převeďme ho do desítkové soustavy.
* Tento směr je jednodušší, stačí vypočítat tento součet:

1100010 = 1\*2^6 + 1\*2^5 + 0\*2^4 + 0\*2^3 + 0\*2^2 + 1\*2^1 + 0\*2^0

* Každý sčítanec má tvar x · 2^i, kde x je číslice z původního binárního čísla a i se zprava postupně zvětšuje vždy o jedna.
* Takže protože převádíme číslo 1100010, vypadá tento součet takto:

1100010 = 1\*2^6 + 1\*2^5 + 0\*2^4 + 0\*2^3 + 0\*2^2 + 1\*2^1 + 0\*2^0

* Číslo 1100010 má sedm číslic, takže mocniny u čísla dva budou postupně 6, 5, …, 1, 0.
* Po umocnění a vynásobení získáme výraz:

1100010 = 64 + 32 + 2 = 98.

* A stejně je tomu zase u jiných soustav, kde sčítáme **X \* (číslo soustavy) ^ (pozice čísla -1)**
* Takže B9 převedeme takto:

11\*16^1 + 9\*16^0 = 11\*16 + 9=185

**Uložení celých čísel v počítači**

* V současné době je standardem používat k uložení celého čísla 32 bitů. Protože je to docela dlouhé dvojkové číslo, použijeme pro následující příklad pouze čtyři bity.
  + Když potřebujeme do 4-bitového čísla uložit číslo s významem celého čísla se znaménkem, máme několik možností.
* V posledním sloupci tabulky je uveden dnes nejrozšířenější způsob - reprezentace celého čísla se znaménkem pomocí tzv. dvojkového doplňku.
* Je to technologicky nejméně nákladný způsob jak obvody  počítače naučit pracovat zároveň s čísly se znaménkem i bez znaménka.
* To proto, je stroj nemusí rozlišovat, zda sčítá či odčítá číslo se znaménkem nebo bez:
* Operace  1010+0010 = 1100 má podle okolností buď význam

10 + 2 = 12

a nebo

- 6 + 2 = - 4

* Protože pomocí 4 bitů můžeme reprezentovat čísla 0… 15 resp. -8..7 vedou některé operace k tzv. přetečení.
* Pro čísla bez znaménka je to např. operace 15+15, její výsledek neleží v intervalu 0… 15.
* Pokud se na tuto operaci díváme z pohledu operací se znaménkem, je ve O.K., nebo (-2)+(-2)=-4.
* Pro čísla se znaménkem je nedovolená např. operace 4+4=8, nebo jako horní mez rozsahu čtyřbitových oznaménkovaných čísel je 7.
* Z hlediska čísel bez znaménka je to ovšem operace dovolená.
* Pro nás znamená typ integer 32 bitové číslo se znaménkem povolující uložení celého čísla v rozsahu  -2 147 483 648  ..  2 147 483 647.
* V případě, že nějaká operace, např. v příkazu k:=i\*j vede k přetečení, není obvykle spuštěn žádný poplach a program se posléze chová podivně bez zjevných příčin, protože výsledek přiřazovacího příkazu je jiný, než zamýšlený.
* Je ale možné donutit program, aby si tato možná přetečení ohlídal, stráví se tím nějaký čas navíc, ale ušetří to čas při hledání problémů
* Abychom však nemuseli hlídat každý součin 200\*200 (nevejde se do SmallInt), budeme předpokládat, že těch bitů je nejméně 32, takže pozor na přetečení si budeme muset dávat až u součinů jako je 50000\*50000.
* S výjimkou typu *Int64* obecně neplatí, že operace s kratším formátem je rychlejší, takže důvody pro použití kratšího formátu čísla musí být v algoritmu samém, ne jeho optimalizaci.

**Uložení desetinných čísel v počítači**

**Mantisa**

* Je uložena na 23 bitech v přímém kódu se znaménkem -
* Znaménkový bit mantisy je označen **z**
* Kladné číslo má **znaménkový bit nulový**, u **záporného čísla je v z uložena 1**
* Nejvyšší bit mantisy je vždy 1 a nezobrazuje se (mantisa se ukládá počínaje druhým významným bitem-ještě zvyšuje přesnost zobrazení)
* Myšlená desetinná tečka je umístěna za nejvyšším bitem mantisy
* Absolutní hodnota mantisy se tedy zobrazí v intervalu 1≤│m│<2
* Od normovaného tvaru se upouští pouze tehdy, když výsledek operace je v absolutní hodnotě menší, než je schopen exponent zobrazit
* Mantisa se pak zmenší na úkor přesnosti a začne se zobrazovat i nejvyšší bit mantisy.

**Exponent**

* je uložen na 8 bitech v kódu s posunutou nulou
* Báze posunutí exponentu je 1 2 − n z (z = 2, a=8) tj. báze posunutí je 27 -1=127

**Zobrazení některých hodnot:**

**Nula:**

* zobrazena s obrazem mantisy i exponentu rovným nule ( podle hodnoty znaménkového bitu – kladná/záporná nula=>nula má dvě možná zobrazení v kódu IEEE 754 )

**Nekonečno:**

* exponent =128, na hodnotě mantisy nezáleží. (podle hodnoty znaménkového bitu – kladné/záporné nekonečno).

**Nenormovaný tvar:**

* má hodnotu exponentu nula a nenulovou mantisu.
* Číslo je uloženo ve čtyřech po sobě jdoucích slabikách.

**Uložení znaků v počítači**

* Znaky se v počítači často ukládají tak, že jeden znak zabírá jeden bajt.
* **Bajt je 8 bitů a může být zaplněn 28-256 způsoby.**
* Jaký znak odpovídá určitému rozložení nul a jedniček v bajtu určuje ASCII tabulka
* Například:

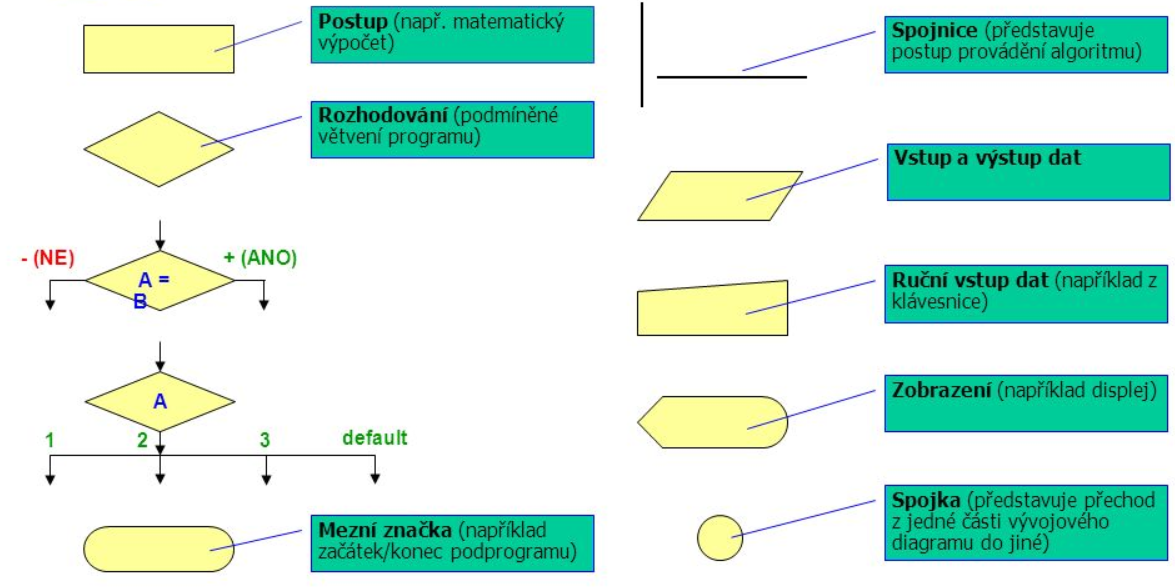
0011111B = 63D je znak ?

01100001B=97D je znak a

* Prvních 32 znaků (od 0 do 31) je vyhrazeno pro řídící kódy a nepředstavují proto tisknutelné znaky.
* Znaky s kódy 32 až 127 jsou běžná interpunkční a matematická znaménka, číslice a velká a malá anglická abeceda.
* Znaky 128 až 255 jsou speciální grafické symboly, například pro psaní rámečků a znaky národních abeced.
* Protože tento počet znaků nestačí pro všechny národní abecedy (neobsahuje například naše čárky a háčky), bývají tyto znaky programově měněny na znaky národní abecedy.
* Pro češtinu se používá hned několik systémů, což problém značně komplikuje.
* Jde zejména o češtinu v kódu Kamenických, kód Latin 2 a kódovou stránku CP1250 používanou ve Windows.
* Podle toho, v jaké češtině je text napsán, je třeba použít to správné kódování.
* **Při použití češtiny je třeba rozlišit dvě věci.** 
  + Samotné instalování češtiny, kdy se na obrazovce správně zobrazují české znaky, a vypínání a zapínání české klávesnice.
* To se často děje pomocí kláves CTRL+ALT+F1. K převodu mezi kódy lze použít některé textové editory nebo speciální programy.
* Další problémy nastávají při tisku českých znaků na tiskárnách.
* Kombinací kláves levý ALT+číselný kód napsaný na numerické klávesnici se zapíše do souboru i znak, který momentálně není na klávesnici k dispozici. (ö-148, ü-129, a pod.)
* Ve Windows je situace komplikována ještě různými fonty. (Pro češtinu jsou určeny fonty s příponou CE.)
* Nejmodernější textové editory postupně přecházejí na systém Unicode, kdy jeden znak zabírá 2 bajty.
* Potom máme k dispozici bez problémů 256\*256=65536 symbolů, což už je plně dostačující. Délka souborů je však dvojnásobná.

1. Grafické značky používané při prezentaci algoritmů, třídící algoritmy

## Značky



* Algoritmus je přesný postup, který je potřeba k vykonání určité činnosti

**Vlastnosti algoritmu:**

* + **jednoznačnost** – každý krok algoritmu musí být jednoznačně a přesně definován
  + **hromadnost** – algoritmus je schopen zpracovávat jakákoliv vstupní data, která vyhovují zadaným podmínkám
  + **konečnost** – musí skončit v konečném počtu kroků
  + **elementárnost** – postup je složený z jednoduchých kroků, které jsou pro počítač srozumitelné
  + **rezultativnost** – výpočet dává po konečném počtu prvků výsledek
  + **efektivnost** – výpočet se uskutečňuje v co nejkratším čase a s využitím co nejmenšího množství prostředků
* Vývojový diagram je grafické znázornění algoritmu
* Vývojové **diagramy se skládají z grafických značek**
  + Značky jsou různé a různě se kombinují, tím se simulují různé situace a různé příkazy
    - Do těchto značek se pak vypisují upřesňující údaje.

## Jednoduchý vývojový diagram



## Třídící algoritmy

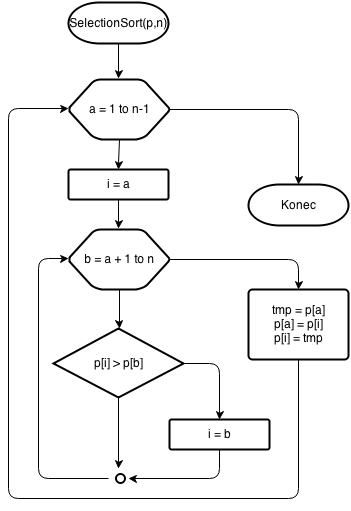
* 🡪 používají se pro třídění nějakých dat.
  + Například k seřazení čísel od nejmenšího k největšímu.
* Většinou třídí data z jednoho datového typu.
* (Pracuje se všemi hodnotami)

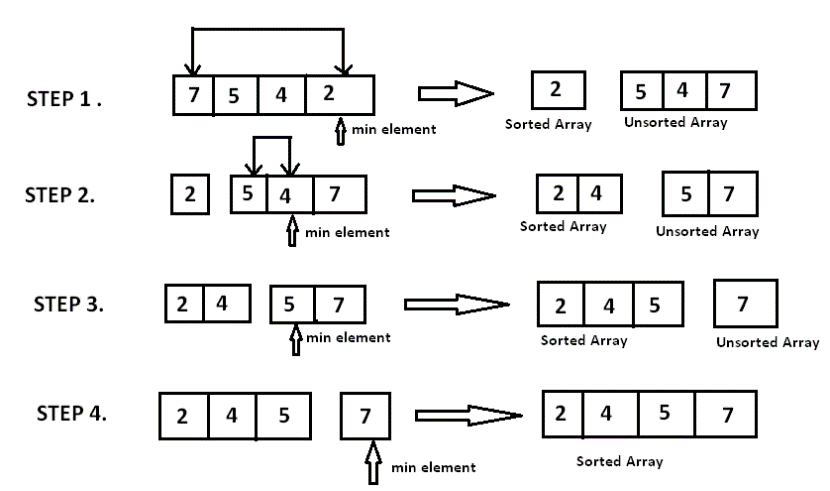
**Dělí se na:**

* **Vnitřní třídění**
  + Pracujeme s daty, které jsou v počítači.
  + Data jsou uloženy v poli a algoritmy využívají možnosti přímého přístupu.
* **Vnější třídění**
  + Použijeme, když se data nevejdou do počítače.
  + Přistupujeme k nim pouze sekvenčně (**postupnost**)
  + **MerchSort**

**Selection Sort –** přímý výběr

* Selection sort je jedním z nejjednodušších řadících algoritmů.
* Jeho myšlenka spočívá v nalezení minima, které se přesune na začátek pole (nebo můžeme hledat i maximum, a to dávat na konec).
  + V prvním kroku tedy nalezneme nejmenší prvek v poli a ten poté přesuneme na začátek.
  + V druhém kroku již nebudeme při hledání minima brát v potaz dříve nalezené minimum.
* Po dostatečném počtu kroků dostaneme pole seřazené.
* **Algoritmus má nepříliš výhodnou časovou složitost a není stabilní.**
* Je však velice jednoduchý na pochopení i implementaci.

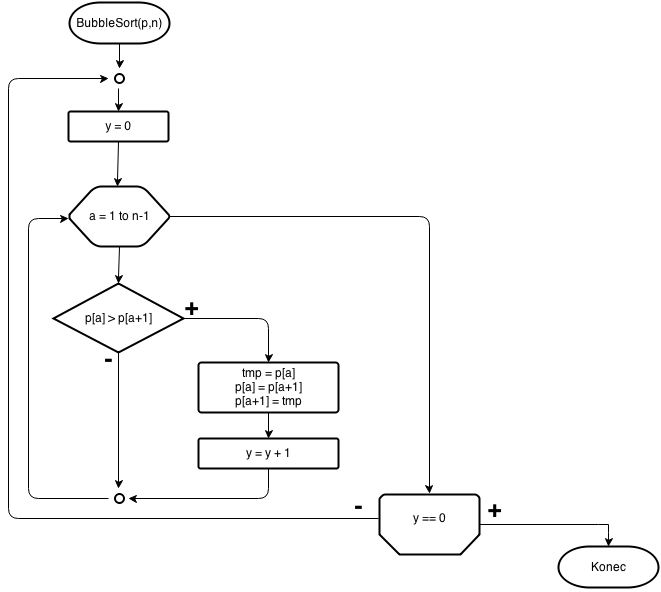


**Průběh**

* Už z popisu je jasné, že budeme používat cykly s daným počtem opakování.
* Potřebujeme měnit počátek rozsahu v rozmezí 1 až *n - 1* a použijeme na to první (vnější) cyklus (index α). Druhým (vnitřním) cyklem budeme procházet jednotlivé prvky pole od *a > 1* do n (index β ), porovnávat je s prvním v daném rozsahu (na indexu ) a hledat nejmenší číslo, od kterého si uložíme index.

**Bubble Sort –** přímá výměna

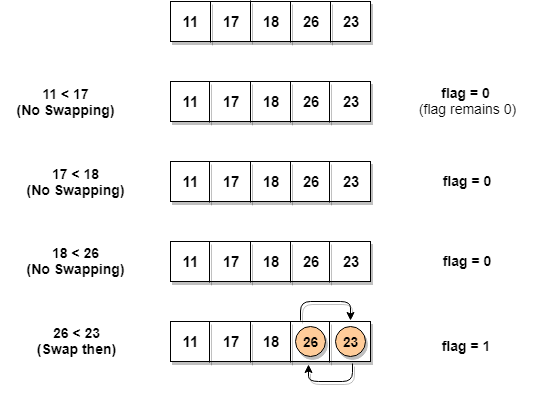
* Poměrně hloupý algoritmus, který se vyskytuje v podstatě jen v akademickém světě.
  + Nemá žádné dobré vlastnosti a je **zajímavý pouze svým průběhem**, který může připomínat fyzikální nebo přírodní jevy.
* Algoritmus probíhá ve vlnách, přičemž při každé vlně propadne "nejtěžší" prvek na konec (nebo nejlehčí vybublá nahoru, záleží na implementaci).
  + Vlna porovnává dvojice sousedních prvků a v případě, že je levý prvek větší než pravý, prvky prohodí.

**Vlastnosti**

* Časová složitost – ***O (n\*n)***
* Stabilní, velmi pomalý.

**Průběh**

* Základem jsou dva cykly.
* Vnější cyklus je s podmínkou na konci, kde testujeme, jestli se provedla nějaká záměna (počet změn si ukládáme do proměnné ***y***).
* Ve vnitřním cyklu s daným počtem opakování se prochází celé pole (do *n - 1* ), testují se sousední prvky pole a případně se zaměňuje jejich obsah (a počítá počet změn).
* Počínaje prvním prvkem (index = 0) porovnejte aktuální prvek s dalším prvkem pole.
* Pokud je aktuální prvek větší než další prvek pole, vyměňte je.
* Pokud je aktuální prvek menší než další prvek, přejděte k dalšímu prvku. Opakujte krok 1 .

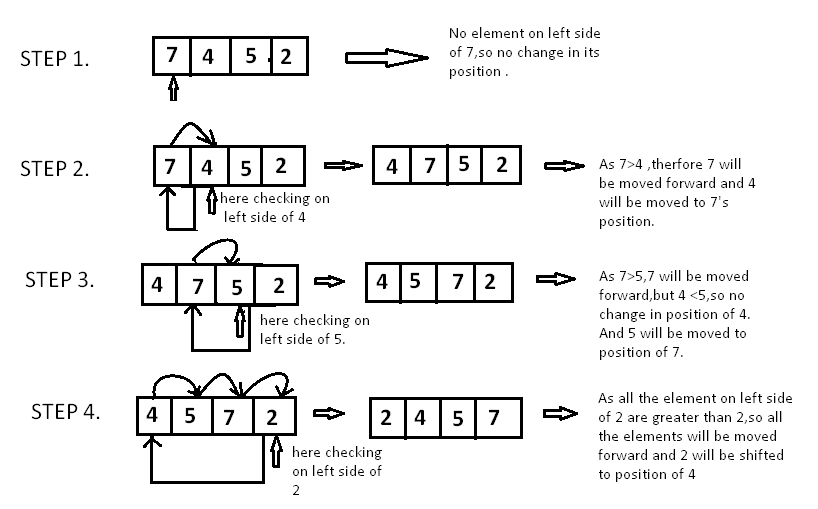
**Uvažujme pole s hodnotami {11, 17, 18, 26, 23}**

**Insertion sort – přímé** vkládání

* Insertion sort je králem mezi jednoduchými třídícími algoritmy.
* Je stabilní, jednoduchý na implementaci, chová se inteligentně na již přetříděných polích a na malých polích (řádově desítky prvků) díky své jednoduchosti **předběhne i QuickSort**.
* Insertion sort vidí pole rozdělené na 2 části – setříděnou a nesetříděnou.
  + Postupně vybírá prvky z nesetříděné části a vkládá je mezi prvky v setříděné části tak, aby zůstala setříděná.

**Vlastnosti**

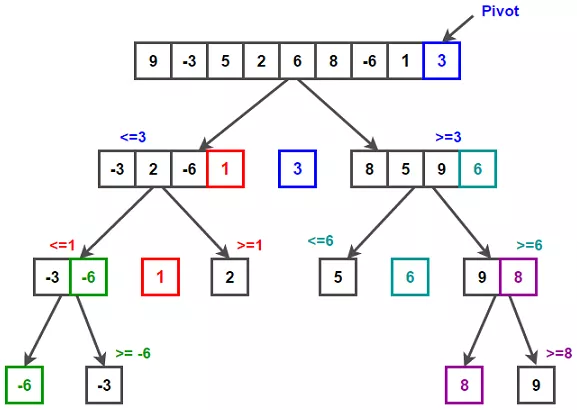
* Časová složitost – **O*(n\*n)***



**Průběh**

* <https://www.hackerearth.com/practice/algorithms/sorting/insertion-sort/tutorial/>

**QuickSort –** rozdělování

* Tato metoda je založena na principu rozdělení pole na dvě části.
* Pro každý prvek ***x*** z jedné části pole platí ***x< y***, kde ***y*** je libovolný prvek z druhé části.
  + Nejvýhodnější dělící hodnotou je medián.
* Nalezení mediánu je však časově náročné, proto se volí jiný prvek – střed pole.

## Základní datové typy, datové struktury

## Význam datových typů a struktur

**Datový typ:**

* Definuje druh nebo význam hodnot, kterých smí nabývat proměnná (nebo konstanta).
* Datový typ je určen oborem hodnot a zároveň výpočetními operacemi.

**Datová struktura:**

* Umožňuje uchovávat a zpracovávat množinu dat stejného typu nebo různorodých, ale logicky souvisejících dat.
* Tyto data lze reprezentovat různými datovými typy.

## Základní rozdělení datových typů

## Ordinální datové typy

* Níže uvedené typy jsou ordinální.
* Hodnoty ordinálního typu tvoří lineárně uspořádanou množinu, kde pro každý prvek je přesně definovaný předchůdce i následovník.

**Celé číslo:**

* Rozsah těchto typů je dán maximální a minimální hodnotou, které může proměnná tohoto datového typu nabývat

byte - 8 bitů

short - 16 bitů

int - 32 bitů

long - 64 bitů

* Některé jazyky (např. Python) místo přetečení pro číslo vyhradí větší množství paměti. Tím je usnadněno programování, avšak snižuje se výkon programu.
* Kromě sčítání, odčítání, násobení, celočíselného dělení a zbytku po dělení je možné provádět nad celými čísly i bitové operace, jako jsou logický součin, logický součet a logická negace. Tyto operace se provádí nad jednotlivými bity operandů ve dvojkové reprezentaci.

**Výčtový typ:**

* I když jde o programátorem definovaný typ, můžeme jej považovat za jistou variantu celočíselného typu. Jde o to, že uživatel může definovat typ např. pro barvu karet:

enum barva {

piky,

srdce,

kary,

krize

};

* V podstatě jde o celé číslo, které smí nabývat pouze zvolené hodnoty. Díky tomu mohou být programy přehlednější a také lépe laditelné.

**Logická hodnota:**

* Datový typ boolean.
* Může nabýt pouze dvou hodnota: true nebo false.
* Obě hodnoty lze vyjádřit prostřednictvím celočíselných hodnot: 1 = true; 0 = false.

**Znak:**

* Datový typ char.
* Používá se pro práci se znaky, pro kódování a sady znaků. Jeho hodnota odpovídá kódu příslušného znaku v ASCII tabulce.

## Neordinální datové typy

**Reálné číslo:**

* Reálná čísla, která lze vyjádřit nekonečně dlouhým desetinným rozvojem jsou představována desetinnými čísly.
* Reálné datové typy v Javě - float (7 desetinných míst), double (15 desetinných míst)

## Složené datové typy

**Pol**e **(array)**

* Může být vícerozměrné (např. dvourozměrné označujeme jako matici)
* Jednotlivé prvky pole jsou dostupné pod číslem, které určuje jejich pořadí (tzv. index).

pole[1] = -5

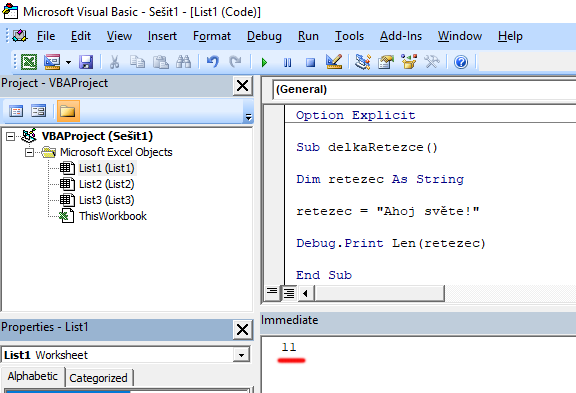
retezec[0] = 'A' (resp. ' ' a 'l')

* V některých jazycích musí prvky pole obsahovat pouze položky stejného typu, jiné jazyky to nevyžadují.
* Hlavní výhodou pole je možnost okamžitého přístupu ke kterékoli jeho položce.

**Seznam (list)**

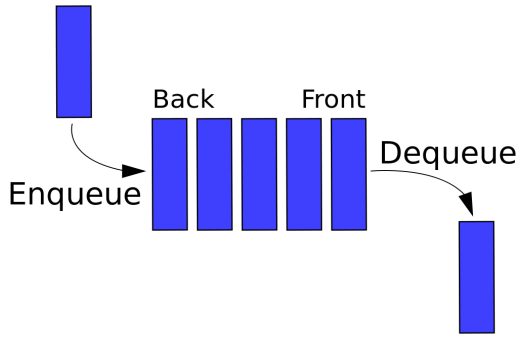
* Obdoba pole, ['a', 'b', 9, „řetězec“].
* Na rozdíl od pole nelze seznam přímo adresovat pomocí indexu.
* Seznam je tedy možné procházet pouze postupně, od začátku do konce, sekvenčně.
* Existují i obousměrné spojové seznamy, které je možné procházet od začátku i od konce, avšak omezení přístupu je výrazné.
* Výhodou seznamů proti polím je, že je možné snadno přidávat nebo odebírat i prvky nacházející se uprostřed seznamu.
* U pole to často znamená nutnost překopírovat velký počet prvků.
* Avšak ve srovnání s polem většinou převažují nevýhody dané složitější vnitřní reprezentací.

**Textový řetězec (string)**

[](https://www.google.cz/url?sa=i&url=https://www.financevpraxi.cz/excel-textove-funkce&psig=AOvVaw2inn4_XAlRvdq4Wkf6q7YA&ust=1581698958836000&source=images&cd=vfe&ved=0CAIQjRxqFwoTCMCSpv39zucCFQAAAAAdAAAAABAO)

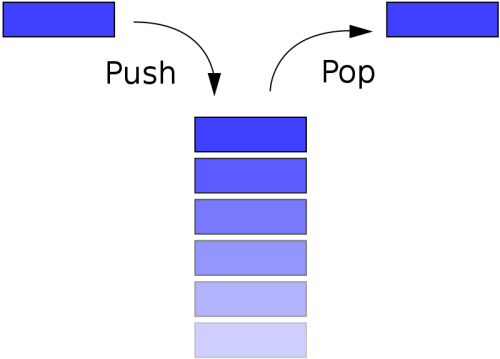
## FIFO, LIFO, strom

**FIFO (fronta)**

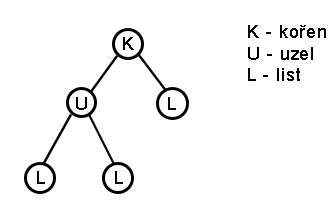
****

* Fronta je taková struktura seznamu, kde první prvek opouští danou strukturu. Konec fronty ohon (chvost). Není možné předbíhat.
* Čelo = začátek fronty
* Prvky vkládáme na jeden konec seznamu a z druhého konce odcházejí.
* Využití:
  + Využívá se tehdy, pokud potřebujeme dočasně uložit nějaké údaje a později je průběžně zpracovat a záleží nám na jejich pořadí.

**LIFO (zásobní)**

* V zásobníku jsou prvky řazeny za sebou.
* Dno zásobníku - 1. prvek.
* Vrchol je proměnlivý podle toho jak dané prvky do dané struktury přicházejí.
* Využití:
  + Využíváme v případě třídících algoritmů, rozděl a panuj (Quicksort) - pro uložení informace jaké dílčí úkoly musíme ještě vykonat
  + Řízení průchodu do hloubky
  + Při zpracování aritmetických výrazů
  + Mechanismus volání procedur a funkcí v programovacím jazyce

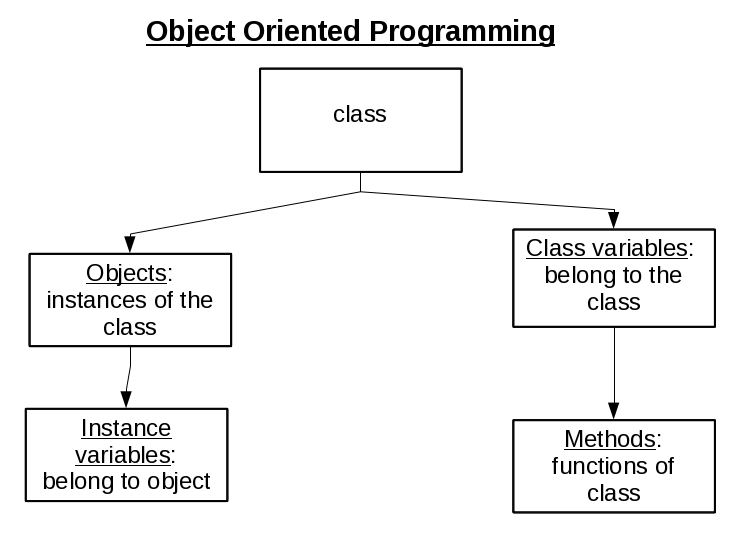
**Strom**

* ****Takové uspořádání dat, kdy každý prvek má nejvýše jednoho předchůdce a může mít více než jednoho následníka.
* Strom tvořen vrcholy (uzly).
* Typy uzlů
  + **Listy** - Uzly bez následníka. Není k nim připojen žádný podstrom.
  + **Kořen** - Uzel bez předchůdce = kořen. Existuje právě 1.
* Vnitřní uzly - Uzly, které nejsou listem ani kořenem.
* Patří mezi rekurzivní datové struktury: každý uzel je současně kořenem stromu a zároveň listem stromu vyšší úrovně.
* Pro každý uzel U platí, že všechny údaje v levém podstromu jsou menší než U a všechny údaje v pravém podstromu větší než U.

## Datové struktury v OOP

**Objekt**

* Objekt si “pamatuje” svůj stav (v podobě dat čili atributů) a zveřejněním některých svých operací (metod) poskytuje rozhraní, jak s ním pracovat.
* Při používání objektu nás zajímá, jaké operace (služby) poskytuje, ale ne, jakým způsobem to provádí – princip zapouzdření.
* Jestli to provádí sám nebo využije služeb jiných objektů, je celkem jedno.
* Vlastní implementaci pak můžeme změnit (např. zefektivnit), aniž by se to dotklo všech, kteří objekt používají.



## Webové technologie

* Tvorba webových stránek je odvětvím, které zahrnuje poměrně velkou škálu různých technologií a programovacích jazyků.
* -> Z tohoto důvodu vzniklo mezinárodní společenství **W3C**, jehož členové společně s veřejností vyvíjí webové standardy pro **World Wide Web**. (WWW)
* Každý moderní web by měl tyto standardy bezpodmínečně dodržovat, jedině tak si může zajistit přístupnost ze všech webových prohlížečů včetně nastupujících mobilních verzí.

**Mezi základní webové technologie patří:**

**Programovací** (či značkovací) jazyky

* + **HTML** - Základní značkovací jazyk pro tvorbu hypertextových dokumentů;

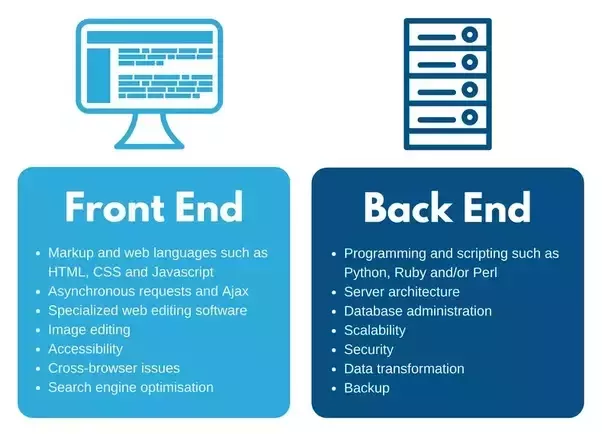
HTML je hlavním z jazyků pro vytváření stránek v systému World Wide Web, který umožňuje publikaci dokumentů na internetu.

* + **XML** - Obecný značkovací jazyk - je určen především pro výměnu dat mezi aplikacemi (např. RSS kanály, exporty zboží z e-shopů do externích vyhledávačů, export dat do ekonomických SW, apod.) a pro publikování dokumentů.
  + **CSS** - Jazyk pro popis způsobu zobrazení stránek napsaných v jazycích HTML, XHTML nebo XML
  + **JavaScript** - Programovací jazyk, který je vykonáván přímo ve webovém prohlížeči - je vhodný pro obsluhu a kontrolu formulářů, ulehčení administrace.
  + **AJAX** - Moderní technologie, která spojuje výše popsané jazyky dohromady, je vhodná pro skryté přenosy dat mezi klientem a serverem, čímž tak odpadá zbytečné znovunačítání stránek při každé akci.
  + **PHP** - Programovací jazyk, který pracuje na straně serveru. PHP generuje zdrojové kódy XHTML a XML na základě nastavení v administraci a databázi.

**Databázové systémy**

* + **SQLite** - Nejedná se sice o plnohodnotný databázový systém, pro jednoduché webové aplikace je však vhodný díky svojí minimální hardwarové náročnosti a multiplatformosti.
  + SQLite je databáze, která se vejde do jediného datového souboru.
  + SQLite se využívá na většině menších až středně velkých www.
* **MySQL**
  + Nejpoužívanější databázový systém pro webové aplikace.
  + MySQL bylo od počátku optimalizováno především na rychlost, a to i za cenu některých zjednodušení proti ostatním databázovým systémům.
  + Využívá se především pro větší webové prezentace a e-shopy.
* **PostgreSQL**
  + Plnohodnotný databázový systém vhodný především pro rozsáhlé webové projekty
* **PHP** 
  + Je skriptovací programovací jazyk.
  + Je určený především pro programování dynamických internetových stránek a webových aplikací například ve formátu HTML, XHTML či WML.
  + PHP lze použít i k tvorbě konzolových a desktopových aplikací.  
    Pro desktopové použití existuje kompilovaná forma jazyka.
  + Při použití PHP pro dynamické stránky jsou skripty prováděny na straně serveru
  + Interpret PHP skriptu je možné volat pomocí příkazového řádku, dotazovacích metod HTTP nebo pomocí webových služeb.
  + Syntaxe jazyka je inspirována několika programovacími jazyky (Perl, C, Pascal a Java).
  + **Jazyk PHP je nezávislý na platformě**, rozdíly v různých operačních systémech se omezují na několik systémově závislých funkcí a skripty lze většinou mezi operačními systémy přenášet bez jakýchkoli úprav.

**Front end vs Back end**



**Další důležité pojmy**

* **URL** –řetězec znaků definující doménovou adresu serveru, umístění zdroje a protokol.  
   URL je ale častěji znám jako adresa webu, obvykle v podobě [*http://www.kurzygrafiky.cz*](http://www.kurzygrafiky.cz)
* **Flash** –způsob vložení multimediálních prvků do stránek. Nejčastějším použitím jsou animované bannery. Flash je možné využít pro přehrávání videa a zvuku.
* **phpMyAdmin** - online administrační prostředí pro práci s databází
* **Webhosting** - online služby (obvykle třetích firem) poskytujících prostor a funkce nutné pro provoz webových stránek.
* **Doména** - unikátní název webových stránek (doménu můžete založit zdarma, nebo si objednat placenou)
* **FTP** – souborů od uživatele na server. Způsob, jak publikovat vaše webové stránky. Základním principem FTP je kopírování souborů na server (**upload**), nebo jejich stažení (**download**).
* **Redakční systémy** – Předem připravené webové stránky (šablony), které jsou rozděleny na dvě hlavní části – uživatelskou část (front-end) a správcovskou část (back-end).
* **IDE** – Integrated Development Environment – Vývojové prostředí

## Programování v aplikacích a tvorba formulářových aplikací

**Příklady aplikací, ve kterých je možné programovat; možnosti využití programování v aplikacích**

**# C**

* Jazyk s virtuálním jazykem (jako např.: Java)
  + Strojový kód se zkompiluje do mezikódu
  + - CIL (Common Intermediate Language)
  + Následně je interpretován virtuálním strojem
  + V NET. Tzv. CLR – Common Language Runtime
* Úzce spjatý s platformou .NET Framework

**.NET Framework**

* Platforma firmy Microsoft pro vývoj aplikací
* Primárně pro platformu Windows, ale existují i verze pro jiné OS
* S .NET Framework lze vytvořit: desktopové aplikace, aplikace pro Windows Store, webové aplikace

**Windows Forms**

* Starší, základní nástroj pro výuku aplikací pro Windows

**Windows Presentation Foundations**

* Nezávislé na platformě
* Součástí WPF je Silverlight (webové aplikace)

**Windows Presentation Foundations**

**VBA, vztah mezi VBA, Basic a Visual Basic**

* VBA neboli Visual Basic for Application. Vychází z VB (Visual Basic), ze kterého je odvozen a upraven pro produkty MS Office. Tj.
* VBA má stejnou syntaxi (příkazy se píší stejně, takže přechod z VBA na VB a naopak je bezproblémový). Každý z produktu MS office (Word, Excel, PowerPoint, Project, Access, Outlook, atd. má k dispozici VBA).
* Podobně VBA naleznete některých dalších SW produktů.

**Pojem makro a jak jej lze vytvořit**

* Je definice pravidla, jak bude vstupní posloupnost transformována na výstupní posloupnost (znaků, akcí, výpočtů a podobně).
* Tuto transformaci označujeme jako substituci nebo expanzi makra.
* Původní termín byl zaveden překladači jazyka symbolických adres, kde programátor může pomocí makra zaměnit dlouhou sekvenci příkazů jediným krátkým výrazem (makrem).

**Vytvoření**

* Makra se využívají v mnoha programech pro usnadnění práce s daty a ke zrychlení složitějších úkonů.
* Typicky se lze s makry setkat u kancelářských balíků, jako je například Microsoft Office.
* V něm lze vytvořit makra pomocí nahrávání.
* Uživatel si v daném programu zapne nahrávání a potom vytvoří sadu příkazů (například změní font, velikost, barvu písma apod.) a po skončení procesu nahrávání ukončí.
* Úkony, které uživatel provedl během nahrávání se uloží pod jeden příkaz, který poté může kdykoliv použít stisknutím kombinací daných kláves.
* Druhým způsobem vytvoření makra je napsaní jej pomocí textového editoru pro programátory, který se dá propojit s programem, kde chceme makro uplatnit.
* Mezi takovéto editory se řadí například EMACS, Vim či VBA.

**Makra v Excelu:**

Zobrazení → Makra

Soubor → Možnosti → Přizpůsobit pás karet → Vývojář

Zaznamenat makro → Ctrl + Shift + N

Přepínání oken: Alt + F11

**Rozdíl mezi funkcí a procedurou**

* **Funkce** je blok kódu, kterému předáme nějaké vstupní hodnoty pomocí proměnných (říká se jim argumenty, někdy také parametry), kód se provede a vrátí nějaký výsledek.
* **Procedura** žádný výsledek nevrací, ale jinak je stejná.

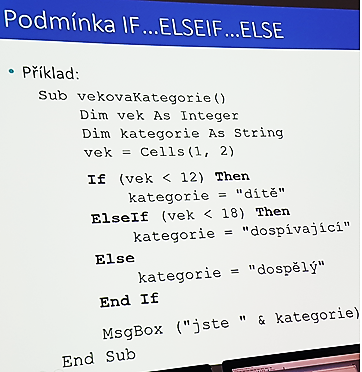
**Proměnná, její souvislost s pojmy platnost a datový typ**

* Proměnná je „úložiště“ informace (tedy vyhrazené místo v paměti – v některých jazycích se ovšem v průběhu výpočtu může místo, kde je proměnná uložena, měnit). Proměnná nebo (v beztypových jazycích) její hodnota má typ. Mezi nejčastější typy patří:
  + celé číslo – často je několik typů s různou maximální velikostí
  + znak – často ovšem znak patří mezi celočíselné proměnné
  + "reálné" číslo – číslo s pohyblivou řádovou čárkou – opět může být několik typů s různou maximální velikostí a přesností
  + řetězec (znaků)
  + pole, někdy i asociativní
  + struktura nebo objekt
  + ukazatel – obvykle může ukazovat na kterýkoliv z ostatních typů nebo na funkci
* Mezi některými typy je možné převádět automaticky.
* Pole, struktury a objekty obsahují další „úložiště“, která opět mají nějaký typ, a lze je používat podobně jako proměnné.
* Proměnné se používají ve vzorcích podobných těm matematickým, ale s odlišným významem: na jedné straně vzorce (téměř vždy pravé) je výraz, ve kterém mají všechny proměnné hodnotu.
* Tento výraz se vyhodnotí a výsledná hodnota (pokud má správný typ, jinak se jedná o chybu) se uloží do levé strany.
* Na levé straně je nejčastěji přímo proměnná, pokud se jedná o výraz musí být jeho výsledkem položka pole, cíl ukazatele nebo podobné označení „úložiště“.

**Pole**

* datová struktura, která sdružuje daný vždy konečný počet prvků (čísel, textových řetězců, …) stejného datového typu.
* K jednotlivým prvkům pole se přistupuje pomocí jejich indexu (celého čísla, označujícího pořadí prvku).
* Velikost pole zůstává při běhu programu neměnná (některé programovací jazyky toto omezení nekladou, zvětšení pole je ale časově náročná operace).
* Ve většině programovacích jazyků je pole vestavěným datovým typem.

**Podmínka a cyklus**

**Podmínka**

* Podmíněný příkaz je v informatice konstrukce - řídicí struktura, která v programovacích jazycích umožňuje provést různý zdrojový kód v závislosti na logickém vyhodnocení určitého výrazu (logický výraz s výstupní hodnotou typu boolean).

**Cyklus For**

Sub cyklus()

Dim i As Integer, cislo As Integer

cislo = 1

For i = 3 To (Cells(1, 1) + 2)

Cells(i, 1) = cislo

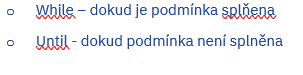
cislo = cislo + 1

Next i

End Sub

**Cykly Do … Loop (While | Until)**

* Cykly probíhají, dokud je poznámka true, resp. není true

**Škodlivý software a možnosti ochrany**

**Makrovirus**

* Makroviry jsou psány v pokročilých makrojazycích současných moderních kancelářských balíků a ve výhledu do blízkého budoucna představují jeden z nejnebezpečnějších virových problémů vůbec.

*Další viz Bezpečnost IT*

**Formulář a příklady ovládacích prvků**

**Formulář**

* označení pro speciální vstupní prvky zobrazitelné ve webovém prohlížeči, které umožňují uživateli odeslat data na webový server, kde jsou následně zpracována.
* Webové formuláře jsou definovány ve formálních programovacích jazycích jako HTML, Perl, PHP, Java, Javascript nebo .NET (včetně ASP.NET)
* **Příklady ovládacích prvků**

## Databáze

**Základní pojmy**

**Entita**

* objekt reálného světa, který je schopen nezávislé existence a je jednoznačně odlišitelný od ostatních objektů
* cokoli, o čem v systému potřebujeme uchovávat nějaké informace
* seznam entit patří k první fázi návrhu datového modelu
  + **konkrétní entity** (např. výrobky)
  + **abstraktní entity** (např. vztahy)

**Atribut**

* -> skutečnost, kterou o dané entitě evidujeme v systému (např. u entity ***VÝROBKY*** budeme evidovat atribut ***JMÉNO VÝROBKU***)

**Relace**

* relace je množina záznamů
* relací může být cokoliv, co je uspořádáno do struktury řádků a sloupců a co obsahuje skalární hodnoty
* relace **nesmí obsahovat 2 či více stejných záznamů** (řádků)

**Primární klíč**

* Primární klíč je jednoznačný identifikátor záznamu, řádku tabulky.
* Primárním klíčem může být jediný sloupec či kombinace více sloupců tak, aby byla zaručena jeho jednoznačnost.
* Pole klíče musí obsahovat hodnotu, tzn. **nesmí se zde vyskytovat** nedefinovaná prázdná **hodnota NULL.**
* V praxi se dnes často používají umělé klíče, což jsou číselné či písmenné identifikátory – každý nový záznam dostává identifikátor odlišný od identifikátorů všech předchozích záznamů (požadavek na unikátnost klíče)

**Cizí klíč**

* Též *nevlastní klíč* Slouží pro vyjádření vztahů, relací, mezi databázovými tabulkami. **Jedná se o pole či skupinu polí**, která nám umožní identifikovat, které záznamy z různých tabulek spolu navzájem souvisí.

**Kardinalita vztahu**

**1**. mezi daty v tabulkách není žádná spojitost, proto nedefinujeme žádný vztah.

**2.** 1:1 používáme, pokud záznamu odpovídá právě jeden záznam v jiné databázové tabulce a naopak.  
 každé entitě odpovídá maximálně jedna druhá entita (**každý člověk má pouze jedno rodné číslo**)

**3**. 1: N přiřazuje jednomu záznamu více záznamů z jiné tabulky.  
 nejpoužívanější typ relace, jelikož odpovídá mnoha situacím v reálném životě.   
vztah ***autobus - cestující***.   
V jednu chvíli cestující jede právě jedním autobusem a v jednom autobuse může zároveň cestovat více cestujících.  
***čtenář si půjčí více knih***, ale 1 kniha je čtena 1 čtenářem

**4.** M:N je méně častá kardinalita.

* první entitě odpovídá více než jedna druhá entita, druhé entitě odpovídá více než jedna první entita
* Umožňuje několika záznamům z jedné tabulky přiřadit několik záznamů z tabulky druhé.  
  V databázové praxi bývá tento vztah z praktických důvodů nejčastěji realizován kombinací dvou vztahů **1:N** a **1:M**, které ukazují do pomocné tabulky složené z kombinace obou použitých klíčů (tzv. **vazební tabulka**).
* vztah ***výrobek - vlastnost.*** Výrobek může mít více vlastností a jednu vlastnost může mít více výrobků.
* Dalším příkladem je vztah ***herec - film***. Jeden herec může hrát ve více filmech, a v jednom filmu může hrát více herců.

**Dekompozice M:N vztahu**

* Vztah M:N se rozkládá na dva jednodušší vztahy 1:N
  + => vytvoří se nová entita na pozici původního vztahu a vyjádří se nové vztahy k původním entitám

**Co to databáze je a jak si ji jednoduše představit?**

* Databáze je množství uložených údajů, tedy dat, vytvářející údajovou základnu – bázi dat
* Příkladem jednoduché databáze je např. **adresář s kontakty**.
  + Máte několik záznamů, kde **co jeden záznam = jeden kontakt**, jedna osoba. Ke každému záznamu máte zapsány další položky, které daný záznam specifikují jako např. jméno a příjmení

**Hlavní funkce databáze**

* **definice dat** - můžeme definovat, jaká data budou uložena v databázi...např. čísla, texty, datum atd.
* **manipulace s daty** - s uloženými daty lze pracovat mnoha způsoby. Můžeme vybírat pouze určité záznamy podle zadaných kritérií, různě je řadit, aktualizovat atd.
* **řízení dat** - můžeme definovat, kdo je oprávněn data pouze číst, kdo má naopak právo data vkládat a aktualizovat.
* **Základním stavebním prvkem** většiny databází jsou **tabulky**
  + Každá tabulka obsahuje informace o konkrétním předmětu, například o výrobcích, pacientech
  + Tabulky obsahují jednotlivé záznamy (neboli ***řádky***), které uchovávají informace o jedné skutečnosti a dále tabulky obsahují pole (neboli ***sloupce***), které uchovávají různé typy dat pro jeden záznam (např. název, žánr, nosič atd.).
    - Takovýmto tabulkám říkáme tabulky relační.
* Celá databáze je pak **RELAČNÍ DATABÁZÍ**

**Relační databáze**

* databáze založená na relačním modelu.
* Často se tímto pojmem označuje nejen databáze samotná, ale i její konkrétní softwarové řešení.
* Relační databázový model sdružuje data do tzv. **relací** (tabulek), které obsahují řádky.
* **Tabulky (relace) tvoří základ relační databáze.** 
  + Tabulka je struktura záznamů s pevně stanovenými položkami (sloupci - atributy).
  + Každý sloupec má definován jednoznačný název, typ a rozsah, neboli doménu.  
     **Záznam se stává řádkem tabulky.**
* Pokud jsou v různých tabulkách sloupce stejného typu, pak tyto sloupce mohou vytvářet vazby mezi jednotlivými tabulkami.
  + Tabulky se poté naplňují vlastním obsahem - konkrétními daty.
* Kolekce více tabulek, jejich funkčních vztahů, indexů a dalších součástí tvoří relační databázi.
* Relační model přináší celou řadu výhod, zejména mnohdy přirozenou reprezentaci zpracovávaných dat, možnost snadného definování a zpracování vazeb apod.
* Relační model klade velký důraz na zachování integrity dat.
* Zavádí pojmy **referenční integrita, cizí klíč, primární klíč, normální tvar**…
* Základním konstruktorem relačních databází jsou relace (databázové tabulky), což jsou dvourozměrné struktury tvořené záhlavím a tělem.
  + **Jejich sloupce = atributy, řádky tabulky jsou pak záznamy.**
* Atributy mají určen svůj konkrétní datový typ a doménu, což je množina přípustných hodnot daného atributu.
* Řádek je řezem přes sloupce tabulky a slouží k vlastnímu uložení dat.

**Referenční integrita**

* omezení vztahu mezi 2 entitami (závislou a nezávislou tabulkou)
* k definování referenční integrity slouží **primární klíč** nezávislé entity a **cizí klíč** závislé entity
* hlídá operace INSERT (vložení záznamu), UPDATE (aktualizace záznamu), DELETE (vymazání záznamu)

**MS Access**

* (tabulky, dotazy, formuláře)
* = nástroj na správu relačních databází od společnosti Microsoft
* **Tabulky**
  + slouží k ukládání dat o určitém předmětu (zaměstnanci, výrobky).
  + Tabulka obsahuje záznamy.
  + V Accessu má tabulka dva typy zobrazení.
    - Jedním je **návrhové zobrazení**, ve kterém vytváříme návrh tabulky (rozhodujeme, jaký formát a datové typy budou mít jednotlivá pole v tabulce).
    - Druhým typem zobrazení je **datový list**. Tam už do tabulky přidáváme samotné záznamy.
* **Dotazy**
  + umožňují vyhledávat data v tabulce, filtrovat je nebo s nimi provádět různé výpočty.
  + Dotaz vytvoříme na kartě vytvoření a poté zvolíme návrh dotazu.
* **Formuláře**
  + objekty, pomocí kterých můžete vy nebo jiní uživatelé přidat, upravit nebo zobrazit data uložená v databázi Accessu.
  + vzhled formuláře je velmi důležitý – efektivita a přesnost zadávání dat
* **Další databázové systémy**
  + Oracle, MySQL, FireBird