9. Řízení vnější paměti

- K čemu slouží vnější paměť
- Charakteristika HDD
- Metody přidělování místa na disku
    - Spojité
    - Spojitý seznam
    - Indexová alokace
- Plánovací metody přístupu na disk
    - FCFS
    - SSTF
    - SCAN
    - C-SCAN
    - LOOK
    - C-LOOK
- HDD vs. SSD
    - Defragmentace

1. K čemu slouží vnější paměť

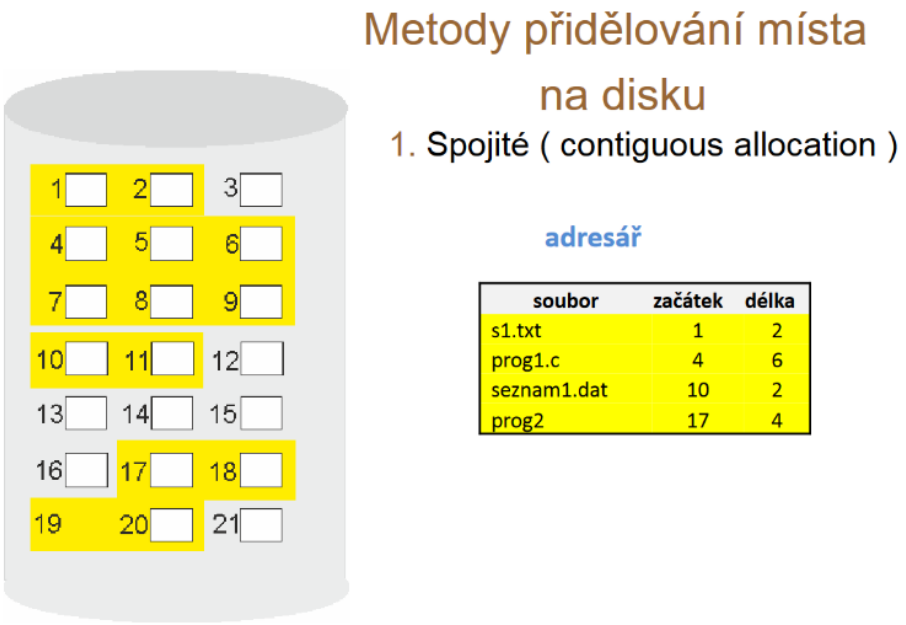
- vnější paměť slouží k trvalému ukládání informací (programy a data)
- Obsah ve vnější paměti se po vypnutí počítače neztratí (jako u RAM paměti)
- Procesor nemá přímý přístup k disku
- OS používá k přístupu do vnější paměti ovladače zařízení
- Data jsou organizována do souboru na základě souborového systému, který je použit
- Výhody
    - Nízké náklady
    - Energetická nezávislost
    - Nedestruktivní čtení
        - přečtení informace žádným negativním způsobem tuto informaci neovlivní
- Stálá paměť: HDD, SSD
- Výměnná paměť: disketa, CD, DVD, USB flash disk

2. Charakteristika HDD

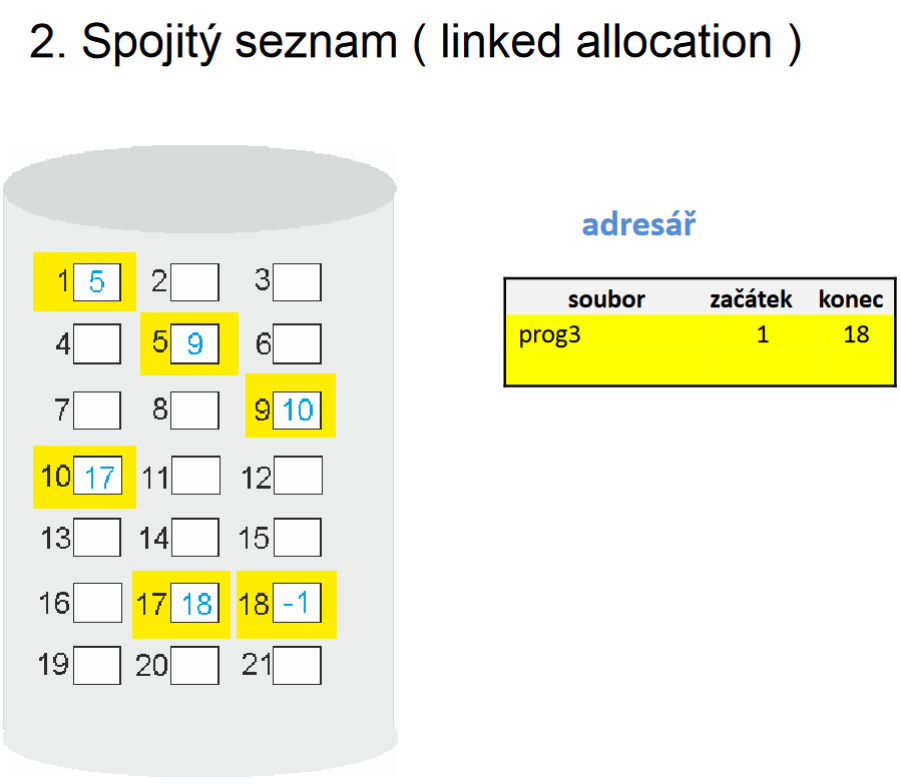
- Hard Disk Drive
- Pevný disk je zařízení, které slouží k trvalému nebo dočasnému ukládání dat
- Data se uchovávají pomocí magnetické indukce
- Výhodou je nízká cena, velká kapacita a delší životnost informací na disku než u SSD
- Nevýhodami jsou pouze sekvenční přístup -> pomalejší než SSD a mechanické řešení -> snadné poškození
- Samotný disk je vyroben z nemagnetického materiálu
- Jeho povrch je pokryt vrstvou feromagnetického materiálu (oxid železa)

3. Metody přidělování místa na disku

- Spojité
  - Nejjednodušší způsob přidělování místa na disku
  - Souvislá alokace, kde každý soubor zabírá množinu sousedních bloků na disku
    - Přesně vím, kde soubor začíná a kolik místa v paměti zabere
  - Pokud je nový soubor moc velký, je nutná defragmentace, aby vznikl dostatečně velký prostor pro uložení souboru.
  - Pokud se soubor v průběhu zvětší, musí se přeuspořádat soubory na disku
  - Přístup na disk je přímý i sekvenční
    - Sekvenční přístup
      - před zpřístupněním informace z paměti je nutné přečíst všechny předcházející informace
      - **postupně prochází od začátku místo v paměti**
    - Přímý přístup
      - je možné zpřístupnit přímo požadovanou informaci
      - **vím, kde soubor začíná a kolik zabere -> uloží rovnou na dané místo v paměti**
  - Výhodou je malý pohyb hlaviček (vystavovacího mechanismu), neboť na sebe bloky navazují
  - Problém vzniká při vzniku nových souborů, protože dopředu nevíme, kolik zaberou místa
  - Při přidělování volného místa se řídí alokační strategií (algoritmy)
    - FIRST FIT
      - Obsadí 1. volný blok, do kterého se proces vejde
      - Nejčastější a nejjednodušší na implementaci
    - BEST FIT
      - Obsadí nejvhodnější blok -> zůstane málo volného prostoru
    - LAST FIT
      - Obsadí poslední volný blok
    - WORST FIT
      - Neřeší nic, umístí se do největšího vyhovujícího volného místa

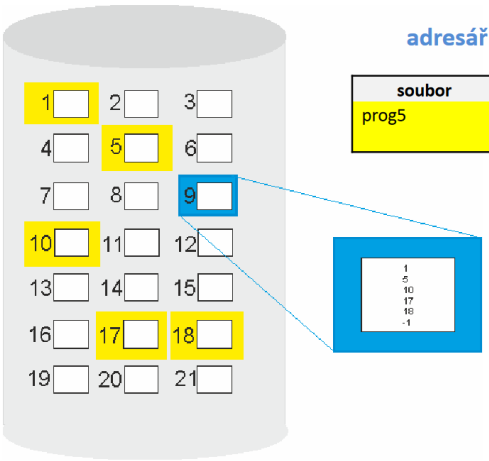


- Spojitý seznam
  - Odstraňuje nutnost souvislého prostoru na disku
    - Stačí nám znát začátek souboru a konec (EOF)
  - Soubor je uložen do volných bloků tak, aby byly jednotlivé části souboru co nejblíže u sebe při minimalizaci pohybu hlaviček
    - Prázdné bloky se zbytečně nepřidělují (nic dopředu)
    - Ovšem bude vždy pomalejší čtení, jelikož málokdy bude celý soubor uložený do bloků hned vedle sebe.
  - Přístup je pouze sekvenční, neboť při ukládání není nutné znát jeho velikost -> potlačení externí fragmentace
    - Po přidání FAT (File Alocation Table, je uložena na začátku disku) je umožněn i přímý přístup, jelikož nám tabulka ukazuje, kam každý blok odkazuje. Nevýhodou je, že FAT zabírá místo na disku a v případě jejího poškození ztratíme data.
  - Velikost alokačního bloku je dána v závislosti na použitém systému a kapacitě disku
    - Alokační blok je cluster
  - Příliš velké alokační bloky způsobí, že část posledního bloku bude nevyužitá a tím vzniká vnitřní fragmentace
  - Využití v MS DOS a WIN95/96



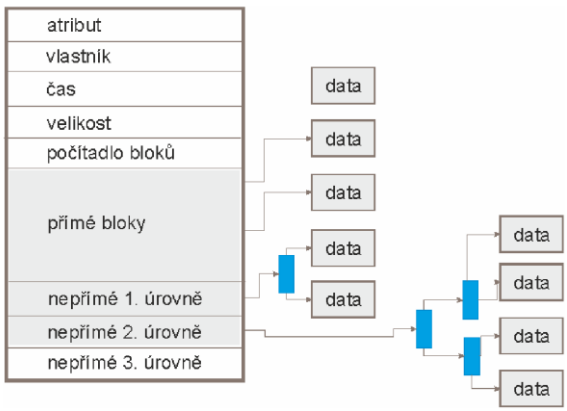
- Indexová alokace

3. Index bloky (index block)



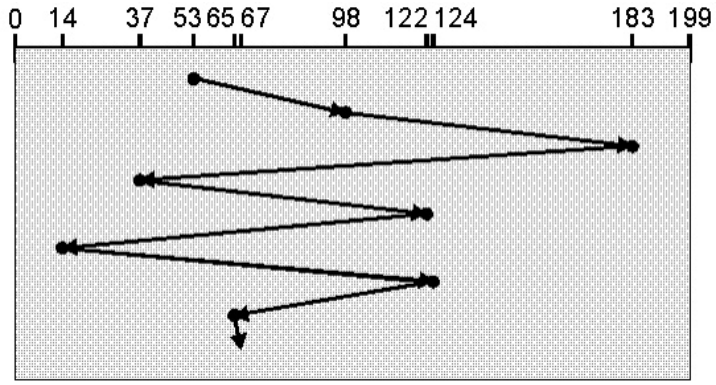
  - Indexy všech bloků souborů jsou umístěny pohromadě v indexovém bloku
    - Každý soubor má svůj indexový blok
    - Při poškození ztratíme pouze daný soubor, a ne všechny data jako u FAT.
  - Vhodné pro přímý i sekvenční přístup
  - Při práci se souborem je indexový blok nahrán do operační paměti
  - Je složitější na realizaci oproti ostatním, ale rychlejší
  - Využití v UNIXu
  - Vzniká vnitřní fragmentace (blok není využitý celý)
  - Potlačená externí fragmentace
  - Snaha mít soubory co nejblíž k sobě z důvodu minimalizace pohybu hlaviček
  - Snaha o co nejmenší indexový blok
    - Metadata souboru
      - Atribut
        - Read
        - Write
        - Execute
        - Directory (zda se jedná o adresář)
        - Znakový speciální soubor
        - Blokový speciální soubor
      - Vlastník
        - Uživatel – vlastník souboru
        - Skupina – možnost nastavení práv pro jednotlivé skupiny abychom je mohli použít při přenosu a nemuseli je nastavovat u každého jednoho uživatele
      - Čas
        - Časové razítka
        - Datum vytvoření, poslední změny a posledního přístupu
      - Velikost
        - Aby se nemusela pokaždé počítat
      - Počítadlo bloků
        - Kontrolní součet konečných dat, abychom věděli, jestli nějaké nechybí
        - Z pravidla 12 přímých bloků po 4kB, v případě, že se nevleze, použijeme nepřímé bloky, kterých je také 12, kde každý odkazuje na 12 přímých bloků. Popř. jsme schopni přidat další vrstvy nepřímých bloků.

4. Index bloky v UNIX



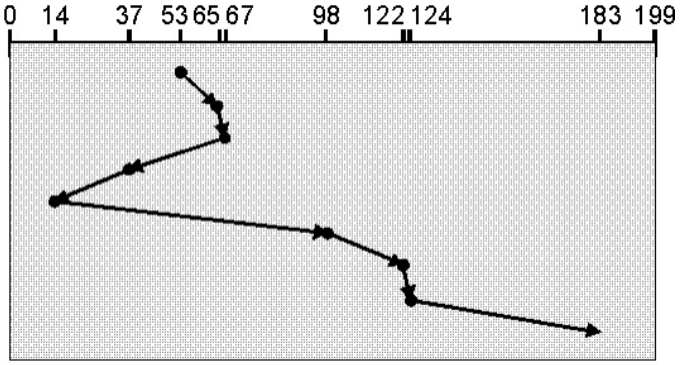
4. Plánovací metody přístupu na disk

- FCFS
    - First Came, first serve
    - První, který přijde, bude nejdříve obsloužený
    - Nejpomalejší
    - Jednoduchý na programování
    - Vhodný pro lehčí zátěž



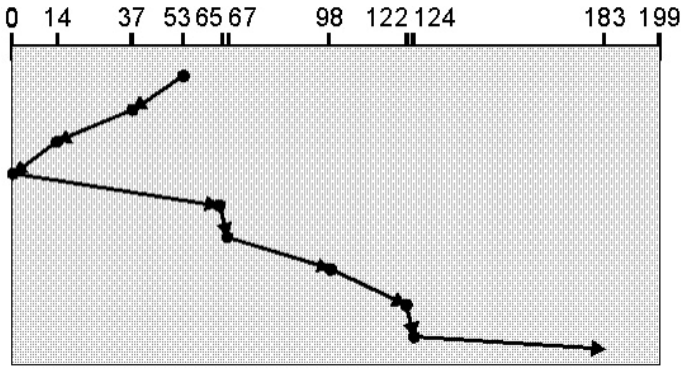
Pořadí	Pozice hl.	cesta
1	53	0
2	98	45
3	189	85
4	37	146
5	122	85
6	14	108
7	124	110
8	65	59
9	67	2

- SSTF
    - Shortest Seek Time First
    - Požadavek, který je nejblíže k hlavičkám má přednost a bude obsloužen jako první
    - Hrozí hladovění požadavků, které budou daleko od hlavy
    - Není ideální
    - Je to rychlá metoda pro krátké vzdálenosti



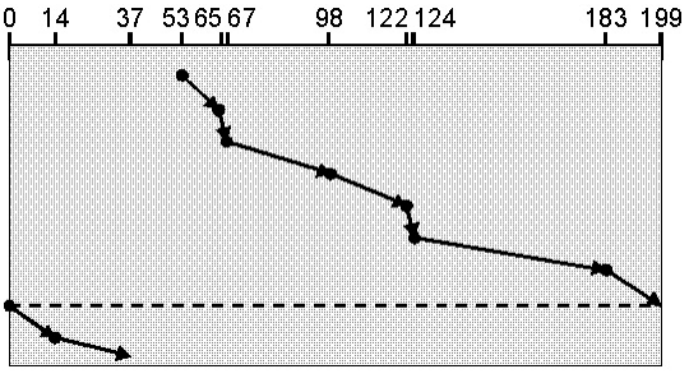
Pořadí	Pozice hl.	cesta
1	53	0
2	65	12
3	67	2
4	37	30
5	14	23
6	98	84
7	122	24
8	124	2
9	183	59

- SCAN
  - Hlava jezdí ze strany na stranu (od kraje, ke kraji) a postupně obsluhuje požadavky
  - Hlavička disku začíná na začátku a přesunuje se na konec, zatímco zpracovává požadavky, které jsou po cestě, pak se vrací zpět



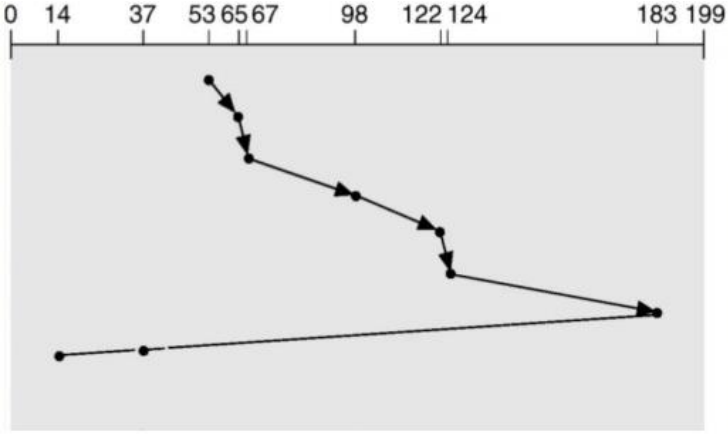
Pořadí	Pozice hl.	cesta
1	53	0
2	37	16
3	14	23
4	65	14+65
5	67	2
6	98	31
7	122	24
8	124	2
9	183	59

- C-SCAN
  - Varianta SCAN, která se liší v tom, že když hlavička dojede na konec, nevrací se zpět, ale přesune se opět na začátek (tam kde začal) a pokračuje ve zpracovávání



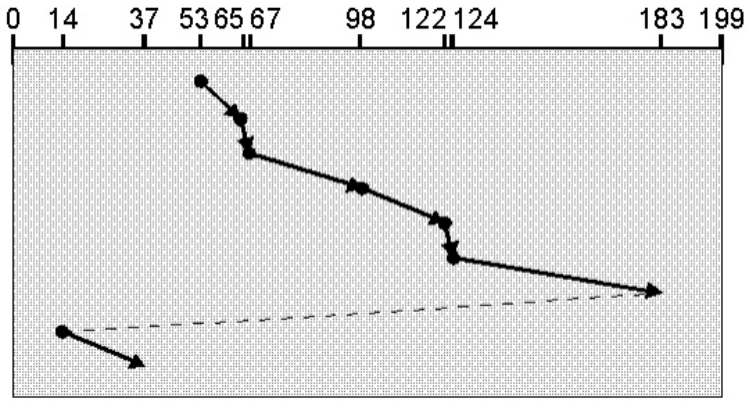
Pořadí	Pozice hl.	cesta
1	53	0
2	65	12
3	67	2
4	98	31
5	122	24
6	124	2
7	183	59
8	14	30(229)
9	37	23

- LOOK
  - Upravená metoda SCAN
  - Hlava začíná na jedné straně disku (na začátku) u prvního požadavku a jde směrem k poslednímu požadavku na druhý konec a zpracovává požadavky, které jsou po cestě
  - Jakmile obslouží poslední požadavek, otočí se a jede zpět a vrací se tam, kde začal



Pořadí	Pozice hl.	cesta
1	53	0
2	65	12
3	67	2
4	98	31
5	122	24
6	124	2
7	183	59
8	37	146
9	14	23

- C-LOOK
    - Vylepšená metoda LOOK
    - Hlava začíná na jedné straně u prvního požadavku a pohybuje se na konec k druhému požadavku, zatímco postupně zpracovává požadavky, které jsou po cestě
    - Jakmile dorazí na poslední požadavek, přesune se opět na začátek disku (na další první požadavek) a proces se opakuje



Pořadí	Pozice hl.	cesta
1	53	0
2	65	12
3	67	2
4	98	31
5	122	24
6	124	2
7	183	59
8	14	169
9	37	23

5. HDD vs. SSD

- **HDD (Hard Disk Drive):**
    - Větší přístupový čas -> pomalejší
    - Větší spotřeba elektřiny kvůli pohyblivým součástkám
    - Dochází k fragmentaci dat -> zhoršení výkonu
    - Hrozí poruchy z důvodu mechanických součástí disku – plotny, hlavy, vystavovací mechanismus
    - Magnetická vrstva, která je na plotnách disku se může při vibracích nebo nárazech poškodit
    - Hlučnější
- **SSD (Solid State Drive):**
    - Kratší přístupový čas -> rychlejší
    - Žádné pohyblivé mechanické součástky -> odolnější proti poškození
    - Nedochází k fragmentaci dat
    - Jedná se o integrovaný obvod
    - Je dražší

- Defragmentace
    - Defragmentace je proces optimalizace dat na disku tím, že fyzicky uspořádá fragmentované soubory tak, aby byly uloženy na disku kontinuálně. To pomáhá zvýšit rychlost přístupu k datům a celkový výkon disku. U SSD není defragmentace tak kritická jako u HDD, protože SSD nemá mechanické části a přístupové časy jsou konzistentní bez ohledu na to, kde jsou data uložena.
    - Defragmentační programy:
        - V OS je to defragmentace
        - O&O Defrag
        - Diskeeper