

Operační systémy 2: Zápočtový úkol

ZS 2023

Cílem zápočtových úkolů bude vytvořit kód podobný tomu, jaký se v operačních systémech stará o uložení dat. Přesto, že pro systémové programování se obvykle používá jazyk C, lze tuto úlohu řešit v libovolném vhodném jazyce. Proto abyste měli možnost zvolit si jazyk, který vám nejlépe vyhovuje, je zadání formulováno obecně a volnou formou a je na vás, abyste je přeformulovali do jazyka funkcí a metod.

1 Blokové zařízení

Vytvořte vrstvu, která bude emulovat blokové zařízení typu pevný disk. Toto blokové zařízení bude data ukládat do zadaného souboru a bude pracovat s bloky o velikosti 512 B. Vámi vytvořená vrstva bude poskytovat následující operace.

- `drv_manufacture(file, size)` – vyrobí nový „disk“, který bude reprezentován souborem `file` a bude mít velikost `size` bloků;
- `drv_open(file)` -> `drive` – zahájí práci s diskem a vrátí objekt reprezentující tento disk;
- `drv_stat(drive)` -> `size` – pro disk `drive` otevřený operací `open` vrátí jeho velikost;
- `drv_read(drive, blockId)` -> `data` – z disku `drive` přečte blok s pořadovým číslem `blockId` a vrátí jej;
- `drv_write(drive, blockId, data)` – na disk `drive` zapíše `data` jako blok s pořadovým číslem `blockId`;
- `drv_close(drive)` – ukončí práci s diskem a uvolní z paměti všechny prostředky s ním spojené.

1.1 Požadavky na implementaci

- Kód vytvořte tak, aby bylo možné v jeden okamžik pracovat s více disky současně.
- Můžete si zvolit jedno z následujících metod implementace. Pro práci se soubory budete používat, buď

(a) mapování souborů do paměti, nebo¹

¹Čárka před *nebo* značí *vylučné nebo*, tj. XOR.

(b) běžné operace typu `read`, `write` poskytované standardní knihovnou jazyka. V takovém případě bude nutné naprogramovat nad blokovým zařízením ještě vrstvu, která se bude chovat jako *write-through cache*. Tato vrstva bude fungovat tak, že:

- při jejím vytvoření uvedete s jakým „diskem“ se má pracovat a kolik bloků se má maximálně držet v paměti;
- při zápisu bloku dat budou data nejdříve zapsaná do cache a následně na disk;
- operace čtení zkontroluje obsah cache, a pokud je daný blok v cache, je vrácen okamžitě, v opačném případě je blok načten z disku a uložen do cache;
- pokud dojde k naplnění cache a bude potřeba uvolnit místo pro nový blok, použijte algoritmus na bázi algoritmu druhé šance, abyste určili, který blok se má z cache uvolnit.

2 Souborový systém

Vytvořte souborový systém FAT8, který bude kopírovat architekturu souborových systémů rodiny FAT.

Tento souborový systém bude mít následující strukturu.

- 1. blok disku nese informace o vlastnostech souborového systému, minimálně je to velikost souborového systému a počet bloků (sektorů) připadajících na jeden cluster;
- 2. blok disku obsahuje FAT, obsahující záznamy o velikosti 8 bitů,
- 3. a následující bloky obsahují clusterly s daty.

Jak budete implementovat práci s adresáři a strukturu souborů s adresáři je na vás. V minimální variantě můžete předpokládat, že souborový systém obsahuje jen kořenový adresář, který je umístěn v jednom clusteru. Můžete ale implementovat i pokročilejší řešení. Rozsah metadat, která budou uložena v adresářích společně se soubory je na vás.

Se souborovým systémem se bude pracovat pomocí následujících operací:

- `fs_format(drive)` – na disku `drive` vytvoří prázdný souborový systém, velikost clusteru je zvolena tak, aby byl disk využit maximálním možným způsobem;
- `fs_open(drive) -> fs` – otevře souborový systém na disku `drive` pro další práci a vrátí objekt umožňující pracovat s daným souborovým systémem;
- `fs_close(fs)` – ukončí práci s daným souborovým systémem a uvolní z paměti všechny zdroje s ním spojené;
- `file_open(fs, fileName) -> fd` – na souborovém systému `fs` otevře soubor `fileName` v režimu pro čtení a zápis a vrátí objekt umožňující s tímto souborem pracovat. Ukazatel v souboru je umístěn na začátek. Pokud soubor neexistuje, je vytvořen.
- `file_truncate(fd, size)` – změní velikost souboru na požadovanou velikost; pokud je soubor větší než `size`, je zkrácen a příslušné clusterly uvolněny, pokud je soubor menší, jsou na volné místo vloženy nulové byty;

- `file_seek(fd, position)` – přesune ukazatel souboru na pozici `position` od začátku souboru;²
- `file_stat(fd)` -> `size` – vrací velikost souboru `fd`;
- `file_tell(fd)` -> `position` – vrací pozici ukazatele v souboru;
- `file_read(fd, size)` -> `data` – ze souboru `fd` přečte nejvýše `size` bytů a vrátí tato data. Ukazatel v souboru se posune o přečtený počet bytů.
- `file_write(fd, data)` – od aktuální pozice v souboru zapíše předaná data; v případě zápisu za hranice souboru, je soubor adekvátně rozšířen;
- `file_close(fd)` – ukončí práci se souborem a uvolní všechny prostředky v paměti se souborem spojené;
- `file_readdir(fs[, fileName])` -> `List<fileName>` – vrátí seznam názvů souborů v adresáři. Parametr `fileName` bude použit jen v případě, že je implementována podpora podadresářů.
- `file_delete(fs, fileName)` – odstraní soubor `fileName` a uvolní všechny jeho clustery.

3 Ukázková aplikace

Vytvořte demonstrační aplikaci, která bude využívat funkce z předchozího úkolu a ukáže všechny jejich možnosti. Součástí aplikace by měly být i instrukce, jak s aplikací pracovat.

Odevzdávání úkolů

- Zápočet bude udělen za zpracování zápočtového úkolu a na základě osobní diskuze nad řešením.
- Na konci semestru budou na webové stránce předmětu oznámeny termíny a časy, kdy je možné přijít zkontrolovat zápočtové úkoly. Tyto termíny budou vypsány převážně na zápočtový týden.
- **Důležité:** Nejméně 24 hodin před termínem odešlete vaše řešení s předmětem "OS2: zapocet" na email `petr.krajca (zavinac) upol.cz` a napište, na který termín byste se chtěl(a) přihlásit.
- Řešení odevzdávejte zabalené jako jeden archiv typu zip nebo tar+gz. Název soubor by měl být ve tvaru: `<prijmeni><prvni-znak-jmena>.[zip|tgz]`.
- Kód by měl být řádně a srozumitelně komentován.
- Pokud budou odhalena dvě shodná řešení, nebude zápočet uznán za žádné z takových řešení.
- Na odevzdávaná řešení se vztahují stejná pravidla jako na všechny závěrečné práce. Jmenovitě, pokud nějakou část řešení odněkud přebíráte, je to nutné vyznačit a uvést původní zdroj. To se vztahuje i na služby typu ChatGPT, GitHub Copilot apod.
- Pokud se ukáže, že svému řešení nerozumíte, bude to důvodem pro neudělení zápočtu.

²Pozice ukazatele nesmí být větší než velikost souboru.