

Systemy operacyjne 2016

Projekt nr 2

Termin oddawania 2 luty 2017

Posługując się szablonem zadania dostępnym na stronie zajęć zaprogramuj sterownik obsługi systemu plików FAT16 z użyciem interfejsu **FUSE**¹. Należy zaimplementować zestaw poniższych operacji, których semantykę opisano szczegółowo w rozdziale §4.9 dokumentacji FUSE:

lookup	wyszukiwanie plików w danym katalogu
getattr	pobranie metadanych pliku
opendir	utworzenie uchwytu do katalogu
readdir	czytanie wpisów katalogu
releasedir	zwolnienie uchwytu do katalogu
open	utworzenie uchwytu do pliku
read	przeczytanie wybranego przedziału bajtów z pliku
release	zwolnienie uchwytu do pliku
statfs	pobranie metadanych systemu plików

Zadbaj o zwracanie odpowiednich kodów błędu w razie niepowodzenia powyższych operacji.

Obraz systemu plików należy traktować jako dyskowe urządzenie blokowe z sektorami o wielkości 512 bajtów, tj. dane między pamięcią, a dyskiem można transferować wyłącznie całymi sektorami. W tym celu przygotowano poniższy interfejs dostępu do zawartości obrazu, gdzie `blkoff` to numer sektora względem początku obrazu, a `blkcnt` to liczba sektorów w danym transferze.

```
1 int blk_open(const char *path);
2 void blk_read(int fd, void *buf, size_t blkoff, size_t blkcnt);
3 void blk_write(int fd, const void *buf, size_t blkoff, size_t blkcnt);
4 void blk_close(int fd);
```

Na stronie zajęć podano odnośniki do samouczków objaśniających dyskową reprezentację poszczególnych obiektów systemu plików FAT16 i FAT32. Ich odpowiedniki zdefiniowano jako struktury języka C w pliku nagłówkowym `fat16.h` dołączonym do szablonu zadania.

Przygotowanie

Zainstaluj pakiety `dosfstools` i `libfuse-dev` dla swojej ulubionej dystrybucji Linuksa². Pełna dokumentacja FUSE jest umieszczona na stronie zajęć. Przykłady wykorzystania biblioteki FUSE znajdują się w katalogu `/usr/share/doc/libfuse-dev/examples`. Dobrze jest też poczytać źródła innych sterowników FUSE i zaglądać do plików nagłówkowych biblioteki.

Do testowania sterownika należy utworzyć obraz systemu plików. By ułatwić to zadanie do szablonu zadania dodano skrypt powłoki `make-fat16.sh`, który utworzy 100MiB plik z obrazem systemu FAT16 zawierającym nietrywialną strukturę katalogów wraz plikami.

¹<https://github.com/libfuse/libfuse/#about>

²w tym przypadku nazwy pakietów i ścieżki plików odnoszą się do dystrybucji wywodzących się z Debian-a

Testowanie

Prezentacja rozwiązania będzie polegała na zamontowaniu obrazu systemu plików wywołując:

```
# ./fat16 fs-image.raw mountpoint/
```

... i wykonywaniu poleceń `stat`, `ls`, `find`, `cat` i `df`.

Bonusy

Punkty bonusowe będą przyznawane za:

- obsługę FAT32
- wsparcie dla długich nazw LFN (ang. *long filename*)
- obsługę modyfikacji metadanych pliku (operacja `setattr`)
- operacje na katalogach (operacje `mkdir`, `rmdir`, `create`, `unlink` i `rename`)
- obsługę zapisu i dopisywania danych do pliku (operacje `write` i `fallocate`)
- buforowanie danych (operacje `fsync`, `fsyncdir`, `forget` i `flush`)